SAFR® Documentation

SAFR Facial Recognition

Documentation Version = 3.024

Publish Date = July 10, 2021

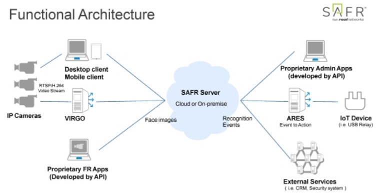
The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

Contents

1 SAFR Overview

SAFR is an exceptionally accurate AI-powered facial recognition that provides a new level of visibility and situational awareness for security professionals. You can easily integrate access control peripherals such as cameras, door locks, or alert systems in order to manage access to a location based on people's identities. SAFR runs on a variety of operating systems, including Windows, macOS, Linux, iOS, and Android.

1.1 SAFR Components



SAFR consists of the following components:

- **SAFR Server**: Available for Windows, macOS, and Linux. SAFR Server consists of a recognition engine, an event server and several databases. The databases contain stored face images of enrolled people, the identity information for the stored faces, and recognition events that have been generated by the SAFR system.
  SAFR Server runs as several background services that automatically start on system reboot and are kept active by the operating system. They must be running at all times for the SAFR system to be operational. In order to be functional, all other SAFR components must maintain a connection to a SAFR Server. Note that if you're doing a cloud deployment you'll be connecting to a SAFR Server in the cloud that RealNetworks maintains.
- **Desktop Client**: Available for Windows and macOS. The Desktop Client is one of the primary ways that administrators and operators can interact with the SAFR system. As such, the client can be used to enable camera connectivity, monitor video camera feeds, register users, view recognition events, and more.
- **Mobile Client**: Available for Android and iOS. The Mobile Client converts a mobile device into a registration kiosk or a recognition panel. Registration kiosks allow people to self-register their face into the Identity Database so they can be approved for access or granted other privileges. Recognition panels enable the mobile device to scan the faces of people that walk by and compare those faces against faces in the Identity Database. Mobile devices set up as recognition panels can also provide visual or audio feedback to the person viewing the mobile device based on actions that a SAFR administrator has configured.
- **Video Recognition Gateway (VIRGO)**: Available as a standalone download for macOS and Linux. It's also available as part of the SAFR Desktop and SAFR Platform download packages. VIRGO is a daemon system which receives video feeds from one or more cameras and recognizes and tracks faces in those video streams in real time. It generates tracking events and sends those events to an event server. The VIRGO video feeds can be controlled either by the command line tool or by the Video Feeds window in the Desktop Client or the Web Console.
- **Web Console**: Available on all platforms. The Web Console provides administrators and operators web-based access to the SAFR system. As such, the Web Console can be used to generate analytical reports, monitor video camera feeds, register users, view recognition events, and more.
- **Actions Relay Event Service (ARES)**: Available as a standalone download for all platforms. ARES is a cross-platform Java application that acts as the event listener that dispatches configured actions in response to events. ARES can provide replies on any event handled by the client that originates an event and is normally installed as a service when either SAFR Platform or SAFR Desktop are installed. It is constantly active and is automatically started by the operating system on power-up.
- **SAFR Actions**: SAFR Actions is a GUI that facilitates configuring SAFRActions.config. SAFRActions.config is the file that defines all the defined actions for your SAFR System, as well as a couple fields that are used to connect ARES (and SAFR Actions) to your primary SAFR Server, whether that

server is local or in the cloud. See Actions for more information about actions in SAFR.

In addition to the SAFR components listed above, SAFR also relies on a couple additional non-SAFR components:

- **IP Cameras**: As you might expect, Internet Protocol (IP) cameras are absolutely integral to SAFR. Both the Desktop Client and VIRGO automatically detect integrated, USB, and Open Network Video Interface Forum (ONVIF) IP cameras. If an IP camera does not support ONVIF or doesn't have ONVIF enabled, you can still manually add it to the SAFR system as described here.
- **Physical access control devices**: Door locks, electronic gates, etc. can all be used by SAFR to grant or deny access to people, depending on whether or not they're identified as having the proper authorization.
- **Notification systems**: Email can be used to discretely notify specified people of various events, while general alarms can be used to alert everybody in the vicinity when unauthorized people attempt to force entry.
- **Additional external peripherals**: Any device that can be controlled by a computer language or protocol can be incorporated into the SAFR system.

1.2 Available Download Packages

The following download packages are available on the SAFR Download Portal:

- **SAFR Platform:** Available on Windows, macOS, and Linux. The SAFR Platform installs everything you need to set up a local deployment of SAFR. This downlaod package enables a locally deployed system to be easily deployed on a single computer and afterwards expanded to additional computers as needed. See Getting Started with SAFR Platform on Windows or macOS and Getting Started with SAFR Platform on Linux for more information.
- **SAFR Desktop:** Available on Windows and macOS. Installs the Desktop Client, SAFR Actions, and one of the VMS extensions. Windows has an additional download variant called SAFR Desktop Lite which has fewer features and lower system requirements. See Getting Started with SAFR Desktop with a Cloud Account for more information.
- **SAFR Mobile:** Available on Android and iOS. Installs the Mobile Client. When you download SAFR Mobile for Android, you're also offered the SAFR Beam download. SAFR Beam allows you to enable the more secure Lock Task Mode on your Android device. If you don't install SAFR Beam, then Android devices can only enable the less secure Screen Pinning Mode. See Configure Devices into Locked Mode for more information.
- **Actions Relay Event Service (ARES):** Available on all platforms. Installs ARES.
- **Video Recognition Gateway (VIRGO):** Available on Linux and macOS. Installs VIRGO.

1.3 Deployment Types

There are two types of SAFR deployment: cloud and on-premise. Each deployment type requires its own account type; a cloud deployment requires a SAFR Cloud Account, while an on-premise deployment requires a SAFR Local Account. Contact your SAFR Account Manager to obtain either type of account.

1.3.1 Cloud Deployments

When SAFR is deployed as a cloud deployment, all your SAFR components are deployed locally except for the SAFR Server. Your components will connect to a SAFR Server located in the cloud which is operated by RealNetworks, Inc. Using the cloud SAFR Server greatly simplifies deployment and maintenance, but it requires a network connection to the cloud at all times in order to be operational.

A single installation of the Desktop Client can handle about 16 connected cameras, assuming the hosting machine meets the recommended system requirements listed here. Expanding your SAFR system beyond this limitation is fairly easy; simply install additional Desktop Clients onto additional machines.

1.3.2 On-Premise Deployments

When SAFR is deployed as an on-premise deployment, all of the SAFR components (including SAFR Server) are installed locally. During installation your SAFR Server will attempt to connect to a SAFR License Server in the cloud to obtain a licence, but after a license has been obtained on-premise deployments do not require a connection to the cloud. Please note that it's possible to obtain a SAFR license without ever connecting your SAFR Server to the cloud; see On-Premise Licensing for details.

A single installation of the SAFR Server can handle about 25 viewed faces at one time, assuming the hosting machine meets the recommended system requirements listed here. Note that for the purposes of server capacity, "25 viewed faces" can mean "25 cameras with 1 face in each camera view" or "1 camera with 25 faces in its camera view", or anything in between. If you want to expand your SAFR system beyond this limitation please see SAFR Server Clusters.

2 SAFR System Requirements

2.1 Windows Requirements

Product	Description	Minimum Requirements	Recommended Requirements
Desktop Client	One of the adminstration consoles for SAFR. Use the Desktop Client to connect to a camera, video feed, detect faces, and submit images to the SAFR Server for recognition.	• Windows Server 2016 or Windows 8.1 • .NET Framework 4.6.2 or later • Intel Core i5-8259U or AMD Ryzen 7 2700X • NVIDIA GT 1030 2GB or Quadro P1000 • SAFR versions earlier than 3.1 are only compatible with NVidia driver versions 418.96 to 431.86 • 16GB RAM • 1.5GB available storage • Supports 2-3 4K cameras[1] • Supports 4+ 1080p cameras[1]	• Windows Server 2016 or Windows 10 • .NET Framework 4.6.2 or later • Intel Core i9-7980XE or AMD Ryzen 7 2700X • NVIDIA GTX 2070 Ti or Quadro P4000 • SAFR versions earlier than 3.1 are only compatible with NVidia driver versions 418.96 to 431.86 • NVIDIA driver 418.96+ for GPU-enhanced performance • 16GB RAM • 1.5GB available storage • Supports up to eight 4K cameras[1] • Supports 9+ 1080p cameras[1]
Desktop Lite	A version of the Desktop Client with fewer features and lower system requirements.	• Windows Server 2016 or Windows 8.1 • .NET Framework 4.6.2 or later • Intel Core i5-7260U • 16GB RAM • 0.5GB available storage	• Windows Server 2016 or Windows 10 • .NET Framework 4.6.2 or later • Intel Core i7-8750H • 16GB RAM • 0.5GB available storage

Product	Description	Minimum Requirements	Recommended Requirements
SAFR Actions	Actions allow you to create and manage responses to event triggers; deploy them to unlock a door, turn on a light, send an alert, record data for reporting, or any security response to fit the use case.	• Windows Server 2016 or Windows 8.1 • Intel Core i3-4340 or AMD Ryzen 7 1750 • 1GB RAM • 1GB available storage	• Windows Server 2016 or Windows 10 • Intel Core i5-726OU or AMD Ryzen 7 1950 • 1GB RAM • 1GB available storage
SAFR Server[2]	The trusted engine of SAFR solutions, the SAFR Server includes: the facial recognition server, identity database, recognition event server, event archive, report server, and remote video feed administration servers.	• Windows Server 2016 or Windows 10 • .NET Framework 4.6.2 or later • Intel Core i9-7980XE or AMD Ryzen TR 1950X • NVIDIA Quadro P2000 • SAFR versions earlier than 3.1 are only compatible with NVidia driver versions 418.96 to 431.86 • NVIDIA driver 418.96+ for GPU-enhanced performance • 16GB RAM • 1TB available storage	• Windows Server 2016 or Windows 10 • .NET Framework 4.6.2 or later • Intel Core i9-7980XE or AMD Ryzen TR 3700 • Quadro RTX 5000 or Tesla T4 • SAFR versions earlier than 3.1 are only compatible with NVidia driver versions 418.96 to 431.86 • NVIDIA driver 418.96+ for GPU-enhanced performance • 32GB RAM • 1TB available storage

1 = Number of cameras is based on an average of five visible faces in a 4K resolution camera view, running at 15 frames per second. Using fewer faces per camera and lower resolution will enable support for more cameras.

2 = Installed as part of the SAFR Platform installer.

2.2 macOS Requirements

Product	Description	Minimum Requirements	Recommended Requirements
Desktop Client	One of the adminstration consoles for SAFR. Use the Desktop Client to connect to a camera, video feed, detect faces, and submit images to the SAFR Server or Cloud for recognition.	• macOS 10.12 • Dual Core i7 • 1GB RAM per connected camera • 0.5GB available storage • Supports one 4K camera[1] • Supports 2+ 1080p cameras[1]	• **iMac Pro:** • macOS 10.12 • 8-core Intel Xeon • 1GB RAM per connected camera • 0.5GB available storage • Supports up to eight 4K cameras[1] • **iMac, MacBook Pro, Mac mini:** • macOS 10.12 • 6-core Intel i7 • 1GB RAM per connected camera • 0.5GB available storage • Supports up to three 4K cameras[1] • Supports 4+ 1080p cameras[1]
SAFR Actions	Actions allow you to create and manage responses to event triggers; deploy them to unlock a door, turn on a light, send an alert, record data for reporting, or any security response to fit the use case.	• macOS 10.12 • Dual Core i5 • 1GB RAM • 0.5GB available storage	• macOS 10.12 • Quad Core i5 • 1GB RAM • 0.5GB available storage

Product	Description	Minimum Requirements	Recommended Requirements
SAFR Server[2]	The trusted engine of SAFR solutions, the SAFR Server includes: the facial recognition server, identity database, recognition event server, event archive, report server, and remote video feed administration servers.	• macOS 10.12 • Quad Core i7 2.6GHz • 16GB RAM • 1TB available storage • Supports one 4K camera[1]	• **iMac Pro:** • macOS 10.12 • 10-core Intel Xeon • 32GB RAM • 1TB available storage • Supports up to six 4K cameras[1] • Supports 7+ 1080p cameras[1] • **Mac mini:** • macOS 10.12 • 6-core Intel i7 3GHz • 32GB RAM • 1TB available storage • Supports up to four 4K cameras[1] • Supports 5+ 1080p cameras[1]

1 = Number of cameras is based on an average of five visible faces in a 4K resolution camera view, running at 15 frames per second. Using fewer faces per camera and lower resolution will enable support for more cameras.

2 = Installed as part of the SAFR Platform installer.

2.3 Linux Requirements

Product	Description	Minimum Requirements	Recommended Requirements
Desktop Client	Not available on Linux.	N/A	N/A
SAFR Actions	Not available on Linux.	N/A	N/A

Product	Description	Minimum Requirements	Recommended Requirements
SAFR Server[1]	The trusted engine of SAFR solutions, SAFR Server includes: the facial recognition server, identity database, recognition event server, event archive, report server, and remote video feed administration servers.	• Linux Ubuntu 16.04(.5+), Ubuntu 18.04(.2+), CentOS 7.x, or Amazon Linux 2018.03 • Intel Core i5-8259U or AMD Ryzen 7 2700X • Quadro P2000 • SAFR versions earlier than 3.1 are only compatible with NVidia driver versions 418.96 to 431.86 • 16GB RAM • 1TB available storage **Install the following additional software components to allow VIRGO to run successfully:** • libcurl3 if Linux Ubuntu 16.* is being used • libcurl4 if Linux Ubuntu 18.* is being used • libgomp1 • libatomic1 • libbsd0 • libv4l-0	• Linux Ubuntu 16.04(.5+), 18.04(.2+), CentOS 7.x, or Amazon Linux 2018.03 • Intel Core i9-7980XE or AMD Ryzen TR 3700 • Quadro RTX 5000 or Tesla T4 • SAFR versions earlier than 3.1 are only compatible with NVidia driver versions 418.96 to 431.86 • 32GB RAM • 1TB available storage **Install the following additional software components to allow VIRGO to run successfully:** • libcurl3 if Linux Ubuntu 16.* is being used • libcurl4 if Linux Ubuntu 18.* is being used • libgomp1 • libatomic1 • libbsd0 • libv4l-0

1 = Installed as part of the SAFR Platform installer.

2.4 Jetson Requirements

Product	Description	Minimum Requirements	Recommended Requirements
Desktop Client	Not available on Jetson.	N/A	N/A
SAFR Actions	Not available on Jetson.	N/A	N/A

Product	Description	Minimum Requirements	Recommended Requirements
SAFR Server[1]	The trusted engine of SAFR solutions, SAFR Server includes: the facial recognition server, identity database, recognition event server, event archive, report server, and remote video feed administration servers.	• Linux Ubuntu 18.04(.2+) • 6GB RAM • 5.5GB available storage • Jetson TX2 • Jetson Xavier **Install the following additional software components to allow VIRGO to run successfully:** • libcurl4 • libgomp1 • libatomic1 • libbsd0 • libv4l-0	• Linux Ubuntu 18.04(.2+) • 6GB RAM • 5.5GB available storage • Jetson TX2 • Jetson Xavier **Install the following additional software components to allow VIRGO to run successfully:** • libcurl4 • libgomp1 • libatomic1 • libbsd0 • libv4l-0

1 = Installed as part of the SAFR Platform installer.

2.5 Mobile Requirements

Product	Description	Minimum Requirements	Recommended Requirements
Mobile Client for iOS	Set up a registration kiosk, perform facial recognition, and add users — all from a mobile device.	• iOS 11.0 • iPad Pro or iPhone 6/7/8/X	• iOS 11.0 • iPad Pro or iPhone 6/7/8/X
Mobile Client for Android	Set up a registration kiosk, perform facial recognition, and add users — all from a mobile device.	• Android 5.0 with Google Play Services 13.2.74 or later • Quad-core Snapdragon 802 2.5GHz • 2GB RAM • 13MB available storage	• Android 6.0 • Quad-core Snapdragon 802 2.5GHz • Samsung Galaxy Tab S4 • Samsung Galaxy S8 • Google Pixel 2 XL • 2GB RAM • 13MB available storage

Product	Description	Minimum Requirements	Recommended Requirements
SAFR Beam for Android	This SAFR utility allows you to configure Android mobile devices for secure SAFR operation.	• Android 6.0 • Near-Field Communication (NFC) support required • 1MB RAM • 8MB available storage	• Android 6.0 • Near-Field Communication (NFC) support required • 1MB RAM • 8MB available storage

2.6 SDK Requirements

Product	Description	Minimum Requirements	Recommended Requirements
Windows SAFR SDK, Lite Edition	Create a Windows app that can be used to locate and track faces and/or badges in a video file or live video stream. The Lite Edition lacks GPU acceleration, but it has a smaller footprint.	• Windows 8.1 64-bit • C# 7.0 • 1GB RAM per 4k video stream • 60MB available storage	• Windows 10 64-bit • Microsoft Visual C++ (MSVC) 2017 or newer is strongly recommended • C# 7.0 • 1GB RAM per 4k video stream • 60MB available storage
Windows SAFR SDK, Standard Edition	Create a Windows app that can be used to locate and track faces and/or badges in a video file or live video stream. The Standard Edition has GPU acceleration. Note that the Recommended Requirements are for a single stream. For multiple streams see the Windows Desktop Client requirements.	• Windows 8.1 64-bit • C# 7.0 • 1GB RAM per 4k video stream • 0.5GB available storage • NVIDIA GTX 1030 • NVIDIA driver 418.96 or later	• Windows 10 64-bit • Microsoft Visual C++ (MSVC) 2017 or newer is strongly recommended • C# 7.0 • 1GB RAM per 4k video stream • 0.5GB available storage • NVIDIA GTX 1030 • NVIDIA driver 418.96 or later

Product	Description	Minimum Requirements	Recommended Requirements
Linux SAFR SDK, Lite Edition	Create a Linux app that can be used to locate and track faces and/or badges in a video file or live video stream. The Lite Edition lacks GPU acceleration, but it has a smaller footprint than the Standard Edition.	• Ubuntu 16.04(.5+) or 18.04(.2+) • If Ubuntu 18.* is used, you may need to downgrade the OpenSSL installation to version 3. • 1GB RAM per 4k video stream • 60MB available storage **Install the following additional software components to allow VIRGO to run successfully:** • libcurl3 if Linux Ubuntu 16.* is being used • libcurl4 if Linux Ubuntu 18.* is being used • libgomp1 • libatomic1 • libbsd0 • libv4l-0	• Ubuntu 16.04(.5+) or 18.04(.2+) • If Ubuntu 18.* is used, you may need to downgrade the OpenSSL installation to version 3. • 1GB RAM per 4k video stream • 60MB available storage **Install the following additional software components to allow VIRGO to run successfully:** • libcurl3 if Linux Ubuntu 16.* is being used • libcurl4 if Linux Ubuntu 18.* is being used • libgomp1 • libatomic1 • libbsd0 • libv4l-0

Product	Description	Minimum Requirements	Recommended Requirements
Linux SAFR SDK, Standard Edition	Create a Linux app that can be used to locate and track faces and/or badges in a video file or live video stream. The Standard Edition has GPU acceleration.	• Ubuntu 16.04(.5+) or 18.04(.2+) • If Ubuntu 18.* is used, you may need to downgrade the OpenSSL installation to version 3. • 1GB RAM per 4k video stream • 0.5GB available storage • NVIDIA GTX 1030 • NVIDIA driver 418.96 or later **Install the following additional software components to allow VIRGO to run successfully:** • libcurl3 if Linux Ubuntu 16.* is being used • libcurl4 if Linux Ubuntu 18.* is being used • libgomp1 • libatomic1 • libbsd0 • libv4l-0	• Ubuntu 16.04(.5+) or 18.04(.2+) • If Ubuntu 18.* is used, you may need to downgrade the OpenSSL installation to version 3. • 1GB RAM per 4k video stream • 0.5GB available storage • NVIDIA GTX 1030 • NVIDIA driver 418.96 or later **Install the following additional software components to allow VIRGO to run successfully:** • libcurl3 if Linux Ubuntu 16.* is being used • libcurl4 if Linux Ubuntu 18.* is being used • libgomp1 • libatomic1 • libbsd0 • libv4l-0
macOS SAFR SDK	Create a macOS app that can be used to locate and track faces in a video file or live video stream.	• macOS 10.12 • 1GB RAM per 4K video stream • 215MB available storage	• macOS 10.14 • 1GB RAM per 4K video stream • 215MB available storage
iOS SAFR SDK	Create an iOS app that can be used to locate and track faces in a video file or live video stream.	• iOS 11 or higher • iPhone 6 • Swift 5 • 92MB available storage	• iOS 12 • iPhone X or iPad Pro • Swift 5 • 92MB available storage

Product	Description	Minimum Requirements	Recommended Requirements
Android SAFR SDK	Create an Android app that can be used to locate and track faces in a video file or live video stream.	• Android 6.0 • 1GB RAM • 0.5GB available storage	• Android 6.0 • 1GB RAM • 0.5GB available storage

2.7 Embedded SDK Requirements

Product	Description	Minimum Requirements	Recommended Requirements
Windows x86 SAFR Embedded SDK, Lite Edition	Build a facial recognition app on a Windows device with limited resources (RAM, CPU, or memory). The Lite Edition lacks GPU acceleration, but it has a smaller footprint than the Standard Edition.	• Windows 8.1 64-bit • x86 Architecture • 200MB RAM • 60MB available storage	• Windows 10 64-bit • x86 Architecture • 200MB RAM • 60MB available storage
Windows x86 SAFR Embedded SDK, Standard Edition	Build a facial recognition app on a Windows device with limited resources (RAM, CPU, or memory). The Standard Edition has GPU acceleration.	• Windows 8.1 64-bit • x86 Architecture • 200MB RAM • 0.5GB available storage • NVIDIA GTX 1030 or better • NVIDIA driver 418.96 or later	• Windows 10 64-bit • x86 Architecture • 200MB RAM • 0.5GB available storage • NVIDIA GTX 1080 Ti • NVIDIA driver 418.96 or later
Linux x86 SAFR Embedded SDK, Lite Edition	Build a facial recognition app on a Linux x86 device with limited resources (i.e. RAM, CPU, or memory). The Lite Edition lacks GPU acceleration, but it has a smaller footprint than the Standard Edition.	• Ubuntu 16.04(.5+) or 18.04(.2+) • x86 Architecture • 500 MB RAM	• Ubuntu 16.04(.5+) or 18.04(.2+) • x86 Architecture • 500 MB RAM
Linux x86 SAFR Embedded SDK, Standard Edition	Build a facial recognition app on a Linux x86 device with limited resources (RAM, CPU, or memory). The Standard Edition has GPU acceleration.	• Ubuntu 16.04(.5+) or 18.04(.2+) • x86 Architecture • 1500 MB RAM • Nvidia GPU GTX10xx or later	• Ubuntu 16.04(.5+) or 18.04(.2+) • x86 Architecture • 1500 MB RAM • Nvidia GPU GTX10xx or later

Product	Description	Minimum Requirements	Recommended Requirements
Linux ARM SAFR Embedded SDK	Build a facial recognition app on a Linux ARM device with limited resources (RAM, CPU, or memory).	• Ubuntu 18.04(.02+) • 64bit ARMv8 CPU • 200 MB RAM	• Ubuntu 18.04(.02+) • 64bit ARMv8 CPU • 200 MB RAM
Jetson SAFR Embedded SDK	Build a facial recognition app on a Jetson device with limited resources (RAM, CPU, or memory).	The following Jetson devices are supported: • Nvidia Jetson TX2 • Nvidia Jetson Xavier • Nvidia Jetson Nano	The following Jetson devices are supported: • Nvidia Jetson TX2 • Nvidia Jetson Xavier • Nvidia Jetson Nano
Android ARM SAFR Embedded SDK	Build a facial recognition app on an Android device with limited resources (RAM, CPU, or memory).	• Android 6.0 • ARMv7 or ARVMv8 Architecture • 200MB RAM • 150MB available storage	• Android 6.0 • ARMv7 or ARVMv8 Architecture • 200MB RAM • 150MB available storage

3 Getting Started with SAFR Platform on Windows or macOS

The computer used for the first installation of SAFR Platform acts as the primary server for the entire SAFR system. The primary server acquires a SAFR license that is then restricted to that machine. (See On-Premise Licensing for details.) Any additional instances of SAFR Server you install under the same SAFR Local Account must be configured as secondary servers for the purposes of load balancing or redundancy and are linked to the primary server as described in SAFR Server Clusters.

3.1 SAFR Platform Contents

The Windows SAFR Platform installation includes the following:

- **SAFR Server**: Includes the recognition engine, event server, several databases, and the Web Console. The databases contain stored enrolled face images, the identity information for the stored faces, and recognition events that have been generated by the SAFR system.
- **Desktop Client**: One of the primary ways that administrators and operators can interact with the SAFR system. As such, the client can be used to enable camera connectivity, monitor video camera feeds, register users, view recognition events, and more.
- **Web Console**: Provides web-based access to the SAFR system. As such, the Web Console can be used to generate analytical reports, monitor video camera feeds, register users, view recognition events, and more.
- **SAFR Actions**: SAFR Actions is a GUI that facilitates configuring SAFRActions.config. SAFRActions.config is the file that defines all the defined actions for your SAFR System, as well as a couple fields that are used to connect ARES (and SAFR Actions) to your primary SAFR Server, whether that server is local or in the cloud. See Actions for more information about actions in SAFR.
- **Actions Relay Event Service (ARES)**: ARES is a cross-platform Java application that acts as the event listener that dispatches configured actions in response to events. ARES can provide replies on any event handled by the client that originates an event and is normally installed as a service when either SAFR Platform or SAFR Desktop are installed. It is constantly active and is automatically started by the operating system on power-up.
- **0 or 1 of the VMS extensions**
- **Video Recognition Gateway Administration (VIRGO)**: Receives video feeds from one or more cameras, recognizes and tracks faces in those video streams in real time, generates tracking events, and sends events to an event server.

The macOS SAFR Platform installation includes the following:

- **SAFR Server**: Includes the recognition engine, event server, several databases, and the Web Console. The databases contain stored enrolled face images, the identity information for the stored faces, and recognition events that have been generated by the SAFR system.
- **Desktop Client**: One of the primary ways that administrators and operators can interact with the SAFR system. As such, the client can be used to enable camera connectivity, monitor video camera feeds, register users, view recognition events, and more.
- **Web Console**: Provides web-based access to the SAFR system. As such, the Web Console can be used to generate analytical reports, monitor video camera feeds, register users, view recognition events, and more.
- **SAFR Actions**: SAFR Actions is a GUI that facilitates configuring SAFRActions.config. SAFRActions.config is the file that defines all the defined actions for your SAFR System, as well as a couple fields that are used to connect ARES (and SAFR Actions) to your primary SAFR Server, whether that server is local or in the cloud. See Actions for more information about actions in SAFR.
- **Actions Relay Event Service (ARES)**: ARES is a cross-platform Java application that acts as the event listener that dispatches configured actions in response to events. ARES can provide replies on any event handled by the client that originates an event and is normally installed as a service when either SAFR Platform or SAFR Desktop are installed. It is constantly active and is automatically started by the operating system on power-up.

- **Video Recognition Gateway Administration (VIRGO)**: Receives video feeds from one or more cameras, recognizes and tracks faces in those video streams in real time, generates tracking events, and sends events to an event server.

3.2 Prerequisites

Before you begin the installation, ensure that you have the following prerequisites:

- **SAFR Local Account**: If you're not sure which account type you have, go to the SAFR Download Portal. If SAFR Platform is listed among the downloads, then you have a SAFR Local Account.
- **System requirements**: Ensure that your system meets the minimum system requirements listed here.
- **An up-to-date SAFR On-Premise License**: See On-Premise Licensing for information about SAFR On-Premise Licenses.
- **SSL certificate**: SSL certificates are required if you want your SAFR Server to support HTTPS connections. If you don't care if HTTPS connections are supported, this prerequisite may be skipped. See SSL Certificate Installation for information about how to get an SSL Certificate.
  Note: There are 2 situations where SAFR requires that your server supports HTTPS connections:
  1. **iOS Devices**: The iOS Mobile Client can only connect to the SAFR Server over HTTPS, so you must obtain an SSL certificate if you want to run the Mobile Client on any iOS devices.
  2. **Additional SAFR Servers**: SAFR Servers can only connect to each other over HTTPS, so you must obtain an SSL certificate if you want to install additional SAFR servers. Additional SAFR Servers are used when you want to scale your SAFR system beyond the processing capacity of a single machine. See SAFR Server Clusters for additional information.

3.3 Download and Install SAFR Platform

To download and install SAFR Platform using Windows or macOS, do the following:

1. On the computer where you want to install SAFR Server, open a web browser and go to the SAFR Download Portal.

2. Sign in with your SAFR Local Account's credentials.

3. Once signed in, select your operating system from the menu and download the appropriate SAFR Platform installer.

4. After the download is complete, start the installation.

5. The Platform installer displays a *Choose Components* window where you can choose the features you want to install, such as:

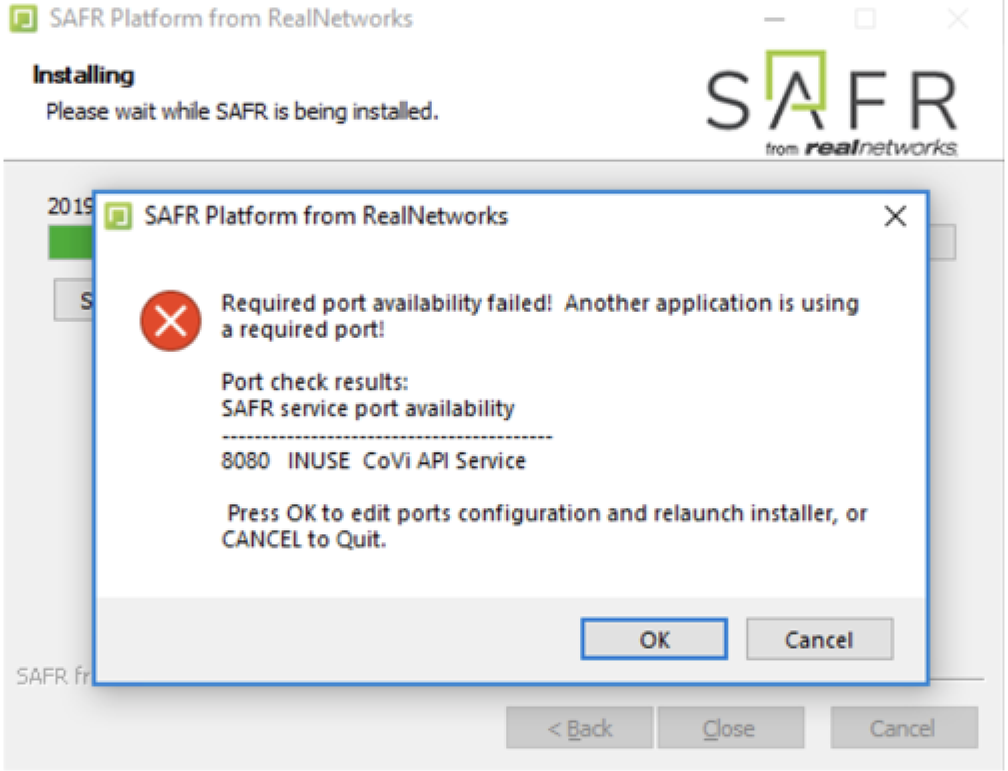
   - SAFR Face Attribute Recognition
     - Age
     - Gender
     - Mask
     - Masked Identity
     - Occlusion
     - Sentiment
     - Optimize GPU models
   - SAFR Peripheral Sub-systems
     - SAFR Actions (If you choose to install SAFR Actions, ARES will automatically also be installed.)
     - SAFR Reports
     - SAFR Logs
     - SAFR Web Console
     - SAFR Video Recognition Gateway (VIRGO)
     - SAFR Video Recognition Gateway Administrator (VIRGA)
   - VMS Extensions

    - Avigilon
    - Digifort
    - Genetec Security Center
    - Genetec Security Center with FaceRec
    - Geutebrueck
    - Milestone XProtect
    - Panasonic Video Insight
  - Camera Extensions
    - Intel RealSense
    - Ximea
  - GPU Support
    - NVIDIA Accelerated Recognition
    - NVIDIA Accelerated Recognition Preprocessing
  - SAFR Application (This refers to the Desktop Client)

  You may uncheck the boxes for any features you do not want to install. We recommend installing all the components except the VMS extensions the first time you install SAFR Platform.
  In addition to whatever components you selected, SAFR Platform also always installs SAFR Server.

6. Follow the installer prompts as they guide you through the rest of the installation process. The final phase of the installation may take a few minutes to complete as it installs dependencies and runs the configuration scripts. Allow it to continue without interruption.

7. On Windows the default SAFR port assignments sometimes conflict with other software port assignments. If a port conflict occurs, the error message shown below will pop up in the middle of your installation.

   
   SAFR Platform from RealNetworks
   Installing
   Please wait while SAFR is being installed.
   SAFR from realnetworks
   2019
   S
   SAFR fr
   SAFR Platform from RealNetworks
   OK
   Cancel
   < Back
   Close
   Cancel

   Required port availability failed! Another application is using a required port!

   Port check results:
   SAFR service port availability

   8080 INUSE CoVi API Service

   Press OK to edit ports configuration and relaunch installer, or CANCEL to Quit.

   If this happens, do the following:

   1. Click **OK** to edit port configurations.
   2. *Notepad* will open, displaying the safrports.conf file.
   3. Edit any conflicting ports to new values. (e.g. CoviHTTP=18080)
   4. Save and exit *Notepad*.

The Platform installer will then restart and the new port values will be used. You can find the modified safrports.conf file at `C:\Program Files\RealNetworks\SAFR\`.

After the installation finishes, three icons will appear on your desktop:

- `SAFR`: Launches the Desktop Client.
- `SAFR Actions`: Launches SAFR Actions.
- `SAFR Admin Console`: Launches the Web Console.

The SAFR Server automatically runs as a collection of background services.

Immediately following installation, the Platform installer opens the Desktop Client and prompts you to log in with your SAFR Local Account. Make sure to log in; it's important in acquiring the SAFR license.

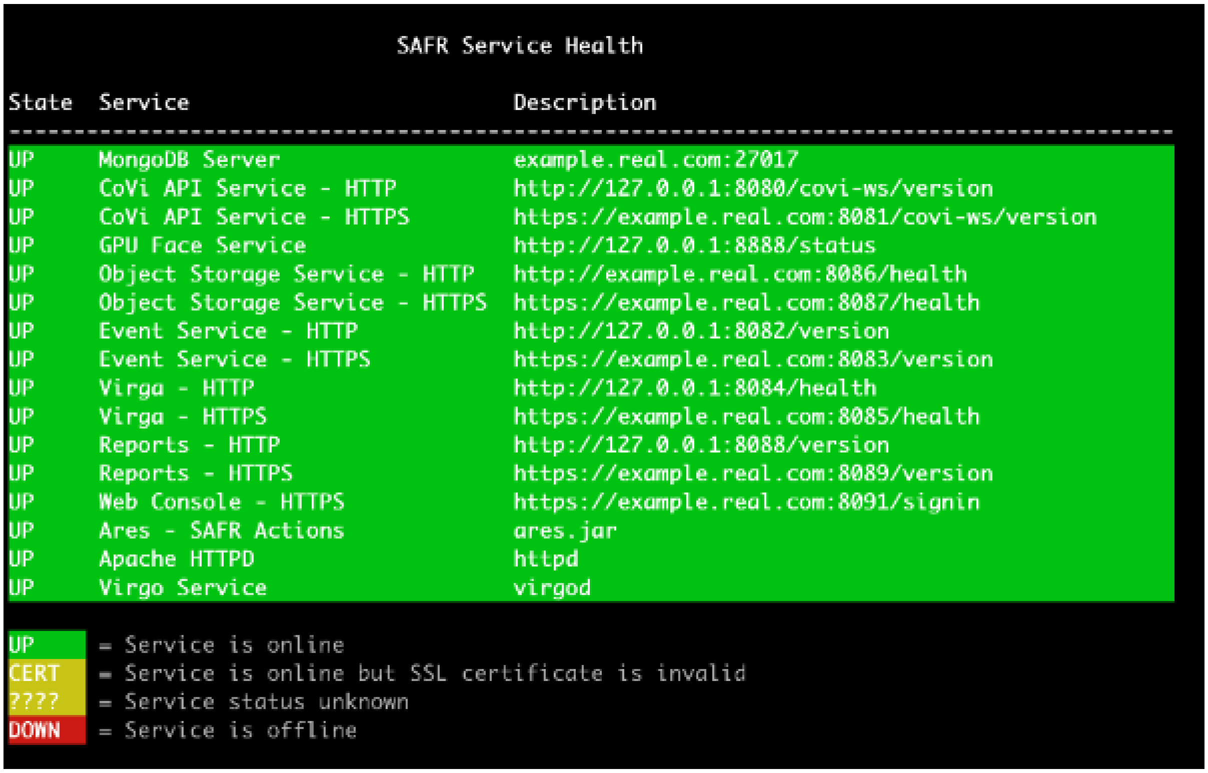
Note: Windows SAFR versions earlier than 3.1 are only compatible with NVidia driver versions 418.96 to 431.86.

3.4 Check Server Status

To check the status of your SAFR Server, use the *check* script. See the table below for the location of the script.

Platform	Script Name	File Location
Windows	check.bat	`C:\Program Files\RealNetworks\SAFR\bin`
macOS	check.sh	`/Library/RealNetworks/SAFR/bin`

The *check* script displays the status of all SAFR services. The following screenshot shows a server installation with healthy statuses for all its services:



```
                              SAFR Service Health

State  Service                          Description
-------------------------------------------------------------------------------------------
UP     MongoDB Server                   example.real.com:27017
UP     CoVi API Service - HTTP          http://127.0.0.1:8080/covi-ws/version
UP     CoVi API Service - HTTPS         https://example.real.com:8081/covi-ws/version
UP     GPU Face Service                 http://127.0.0.1:8888/status
UP     Object Storage Service - HTTP    http://example.real.com:8086/health
UP     Object Storage Service - HTTPS   https://example.real.com:8087/health
UP     Event Service - HTTP             http://127.0.0.1:8082/version
UP     Event Service - HTTPS            https://example.real.com:8083/version
UP     Virga - HTTP                     http://127.0.0.1:8084/health
UP     Virga - HTTPS                    https://example.real.com:8085/health
UP     Reports - HTTP                   http://127.0.0.1:8088/version
UP     Reports - HTTPS                  https://example.real.com:8089/version
UP     Web Console - HTTPS              https://example.real.com:8091/signin
UP     Ares - SAFR Actions              ares.jar
UP     Apache HTTPD                     httpd
UP     Virgo Service                    virgod

UP     = Service is online
CERT   = Service is online but SSL certificate is invalid
????   = Service status unknown
DOWN   = Service is offline
```

4 Connect Desktop Clients

A Desktop Client that is installed on the same machine as the primary server is automatically connected with your primary server; no additional actions need to be taken.

Desktop Clients that are installed on machines other than the primary server, however, need to be configured so they can connect with the primary server. Clients that aren't connected to a server are nearly useless and have very limited functionality.

To connect a remote Desktop Client, do the following:

1. On the remote machine download and install SAFR Desktop for your OS from the SAFR Download Portal.
2. Start the Desktop Client. If prompted, cancel the camera login screen. Also cancel the SAFR Account login if it is displayed.
3. Click **Tools > Preferences**. On the **Account** tab, enter your user identifier and password for your SAFR Local Account.
4. Select *SAFR Custom* from the drop down menu of the *Environment* setting. Do one of the following: **Note**: If you customized ports when installing SAFR Server, use the customized port values instead of the values listed below.
    - If you are running the server without an SSL certificate, enter the following in the associated fields, substituting the server IP Address for **localhost**:
        - CoVi Server: http://localhost:8080
        - Event Server: http://localhost:8082
        - Object Server: http://localhost:8086
        - VIRGA Server: http://localhost:8084
    - If you are running the server with an SSL certificate, enter the following in the associated fields, substituting your server's hostname for **localhost**:
        - CoVi Server: https://localhost:8081
        - Event Server: https://localhost:8083
        - Object Server: https://localhost:8087
        - VIRGA Server: https://localhost:8085
5. Click **OK** to save the preference changes.

5 Getting Started with SAFR Platform on Linux

The computer used for the first installation of SAFR Platform acts as the primary server for the entire SAFR system. The primary server acquires a SAFR license that is then restricted to that machine (see On-Premise Licensing for details). Any additional instances of SAFR Server you install under the same SAFR account must be configured as secondary servers for the purposes of load balancing or redundancy and are linked to the primary server as described in SAFR Server Clusters.

5.1 SAFR Platform Contents

The Linux SAFR Platform installation includes the following:

- **SAFR Server**: Includes the recognition engine, event server, and several databases. The databases contain stored enrolled face images, the identity information for the stored faces, and recognition events that have been generated by the SAFR system.
- **Web Console**: Provides web-based access to the SAFR system. As such, the Web Console can be used to generate analytical reports, monitor video camera feeds, register users, view recognition events, and more.
- **ARES**: Actions Relay Event Service (ARES) is a cross-platform Java application that acts as the event listener that dispatches configured actions in response to events. ARES can provide replies on any event handled by the client that originates an event and is normally installed as a service when either SAFR Platform or SAFR Desktop are installed. It is constantly active and is automatically started by the operating system on power-up.
- **Video Recognition Gateway Administration (VIRGO)**: Receives video feeds from one or more cameras, recognizes and tracks faces in those video streams in real time, generates tracking events, and sends events to an event server.

5.2 Prerequisites

Before you begin the installation, ensure that you have the following prerequisites:

- **SAFR Local Account**: If you're not sure which account type you have, go to the Download Portal. If SAFR Platform is listed among the downloads, then you have a SAFR Local Account.
- **System requirements**: Ensure that your system meets the minimum system requirements listed here.
- **An up-to-date SAFR On-Premise License**: See On-Premise Licensing for information about SAFR On-Premise Licenses.
- **SSL certificate**: SSL certificates are required if you want your SAFR Server to support HTTPS connections. If you don't care if HTTPS connections are supported, this prerequisite may be skipped. See SSL Certificate Installation for information about how to get an SSL Certificate.
  Note: There are 2 situations where SAFR requires that your server support HTTPS connections:
  1. **iOS Devices**: The iOS Mobile Client can only connect to the SAFR Server over HTTPS, so you must obtain an SSL certificate if you want to run the Mobile Client on any iOS devices.
  2. **Additional SAFR Servers**: SAFR Servers can only connect to each other over HTTPS, so you must obtain an SSL certificate if you want to install additional SAFR servers. Additional SAFR Servers are used when you want to scale your SAFR system beyond the procesing capacity of a single machine. See SAFR Server Clusters for additional information.

5.3 Download and Install the SAFR Platform

To download and install SAFR Platform on Linux, do the following:

1. Go to the SAFR Download Portal and enter your SAFR Local Account credentials.

2. On the download page, go to SAFR Platform and select *Linux* from the drop-down menu to the right.
   Note: If you want to install SAFR Platform on NVIDIA Jetson system, you should instead select *Jetson* from the drop-down menu.

3. Right-click the **Download** button for your preferred Linux distribution and select **Copy Link Address**.
4. Download the file to your local machine. The following is an example cURL request which will accomplish this: `curl -L -o safrinst.sh '<your copied link address>'`
5. After the SAFR Platform installer is downloaded use chmod to make the downloaded file executable, if necessary.
6. Run the installer program.
7. The default SAFR port assignments sometimes conflict with other software port assignments. If a port conflict occurs, you'll see this error message:

   ```
   Updating SAFR service port configuration
   Enter new ports, or press enter to accept default.
   ```

8. You will be prompted to reconfigure your conflicted port values, one by one, until all conflicts are resolved.

   ```
   CoviHTTP (8081):
   ```

   The number in parenthesis is the current (i.e. conflicted) port number assignment.

   - If you enter an invalid value, (e.g. FRED) you will receive the error message

     ```
     Invalid response: FRED - Enter integer value between 1024 and 65535.
     ```

     You'll then be prompted to enter a different port number.

   - If you enter a port number that's also conflicted, you'll receive the error message

     ```
     Port 1234 is already in use by CoviHTTP
     ```

     You'll then be prompted to enter a different port number.

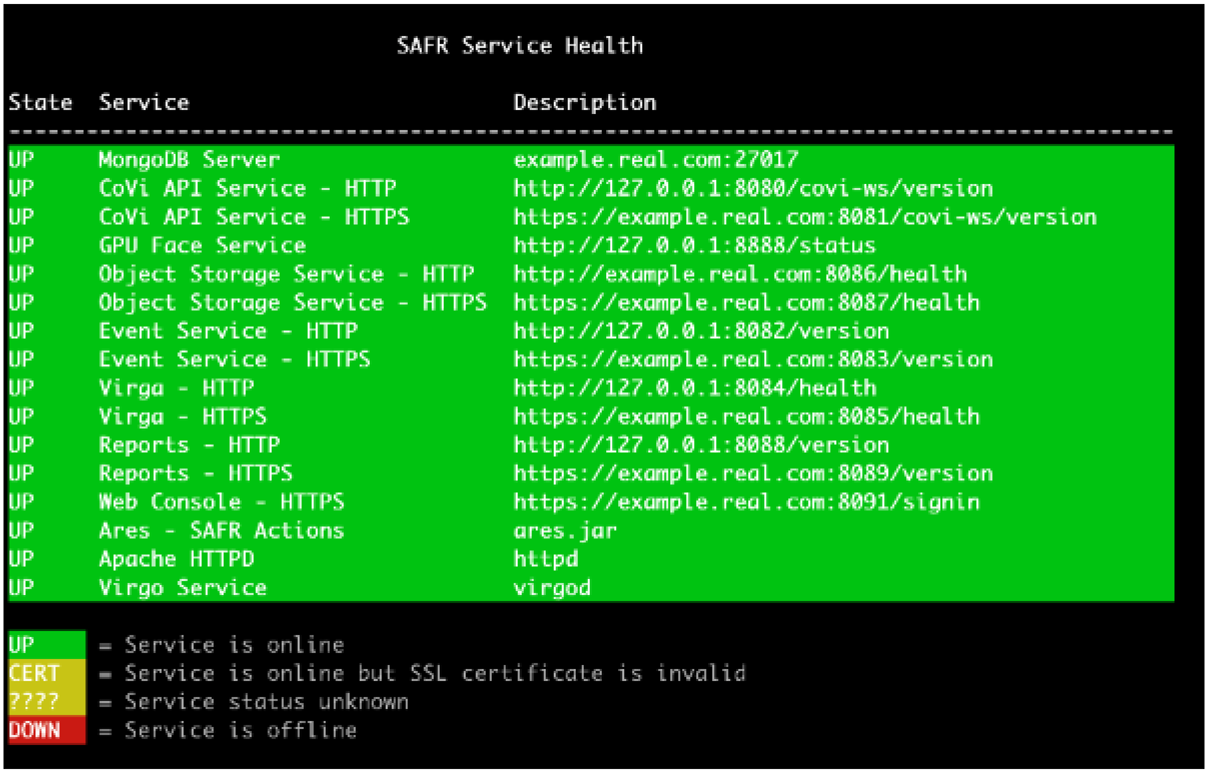
9. The Platform installer will then restart and the new port values will be used. You can find the modified safrports.conf file at `/opt/RealNetworks/SAFR/`.

After it finishes, the installer exits. Your SAFR Server is now running as a collection of background services and is ready for use.

Note: Linux SAFR versions earlier than 3.1 are only compatible with NVidia driver versions 418.96 to 431.86.

5.4 Check Server Status

To check the status of your SAFR Server, run the *check* script by executing the following command: `/opt/RealNetworks/SAFR/bin/check`. The script displays the status of all SAFR services. The following screenshot shows a server installation with healthy statuses for all its services:



```
                              SAFR Service Health

State  Service                        Description
-------------------------------------------------------------------------------------
UP     MongoDB Server                 example.real.com:27017
UP     CoVi API Service - HTTP        http://127.0.0.1:8080/covi-ws/version
UP     CoVi API Service - HTTPS       https://example.real.com:8081/covi-ws/version
UP     GPU Face Service               http://127.0.0.1:8888/status
UP     Object Storage Service - HTTP  http://example.real.com:8086/health
UP     Object Storage Service - HTTPS https://example.real.com:8087/health
UP     Event Service - HTTP           http://127.0.0.1:8082/version
UP     Event Service - HTTPS          https://example.real.com:8083/version
UP     Virga - HTTP                   http://127.0.0.1:8084/health
UP     Virga - HTTPS                  https://example.real.com:8085/health
UP     Reports - HTTP                 http://127.0.0.1:8088/version
UP     Reports - HTTPS                https://example.real.com:8089/version
UP     Web Console - HTTPS            https://example.real.com:8091/signin
UP     Ares - SAFR Actions            ares.jar
UP     Apache HTTPD                   httpd
UP     Virgo Service                  virgod

UP     = Service is online
CERT   = Service is online but SSL certificate is invalid
????   = Service status unknown
DOWN   = Service is offline
```

5.5 Connect Remote Desktop Clients

Desktop Clients that are installed on Windows or macOS machines need to be configured to connect with the primary server. Clients that aren't connected to a server are nearly useless and have very limited functionality.

To connect a remote Desktop Client, do the following:

1. On the remote machine download and install SAFR Desktop for your OS from the Download Portal.
2. Start the Desktop Client. If prompted, cancel the camera login screen. Also cancel the SAFR Account login if it is displayed.
3. Click **Tools > Preferences**. On the **Account** tab, enter your user identifier and password for your SAFR Local Account.
4. Select *SAFR Custom* from the drop down menu of the *Environment* setting. Do one of the following: **Note**: If you customized ports when installing SAFR Server, use the customized port values instead of the values listed below.
   - If you are running the server without an SSL certificate, enter the following in the associated fields, substituting the server URL for **localhost**:
     - CoVi Server: http://localhost:8080
     - Event Server: http://localhost:8082
     - Object Server: http://localhost:8086
     - VIRGA Server: http://localhost:8084
   - If you are running the server with an SSL CERT, enter the following in the associated fields, substituting your server's hostname for **localhost**:
     - CoVi Server: https://localhost:8081
     - Event Server: https://localhost:8083
     - Object Server: https://localhost:8087
     - VIRGA Server: https://localhost:8085
5. Click **OK** to save the preference changes.

6 Getting Started with SAFR Desktop with a Cloud Account

SAFR Desktop installs the Desktop Client as well as SAFR Actions, a programmable interface to create and manage responses to event triggers.

6.1 SAFR Desktop Contents

The Windows SAFR Desktop installation includes the following:

- **Desktop Client**: One of the primary ways that administrators and operators can interact with the SAFR system. As such, the client can be used to enable camera connectivity, monitor video camera feeds, register users, view recognition events, and more.
- **SAFR Actions**: SAFR Actions is a GUI that facilitates configuring SAFRActions.config. SAFRActions.config is the file that defines all the defined actions for your SAFR System, as well as a couple fields that are used to connect ARES (and SAFR Actions) to your primary SAFR Server, whether that server is local or in the cloud. See Actions for more information about actions in SAFR.
- **Actions Relay Event Service (ARES)**: ARES is a cross-platform Java application that acts as the event listener that dispatches configured actions in response to events. ARES can provide replies on any event handled by the client that originates an event and is normally installed as a service when either SAFR Platform or SAFR Desktop are installed. It is constantly active and is automatically started by the operating system on power-up.
- **0 or 1 of the VMS extensions**
- **Video Recognition Gateway Administration (VIRGO)**: Receives video feeds from one or more cameras, recognizes and tracks faces in those video streams in real time, generates tracking events, and sends events to an event server.

The macOS SAFR Desktop installation includes the following:

- **Desktop Client**: One of the primary ways that administrators and operators can interact with the SAFR system. As such, the client can be used to enable camera connectivity, monitor video camera feeds, register users, view recognition events, and more.
- **SAFR Actions**: SAFR Actions is a GUI that facilitates configuring SAFRActions.config. SAFRActions.config is the file that defines all the defined actions for your SAFR System, as well as a couple fields that are used to connect ARES (and SAFR Actions) to your primary SAFR Server, whether that server is local or in the cloud. See Actions for more information about actions in SAFR.
- **Actions Relay Event Service (ARES)**: ARES is a cross-platform Java application that acts as the event listener that dispatches configured actions in response to events. ARES can provide replies on any event handled by the client that originates an event and is normally installed as a service when either SAFR Platform or SAFR Desktop are installed. It is constantly active and is automatically started by the operating system on power-up.
- **Video Recognition Gateway Administration (VIRGO)**: Receives video feeds from one or more cameras, recognizes and tracks faces in those video streams in real time, generates tracking events, and sends events to an event server.

6.2 Prerequisites

Before you begin the installation, ensure that you have the following prerequisites:

- **SAFR Cloud Account**: If you're not sure which account type you have, go to the Download Portal. If SAFR Cloud is listed among the downloads, then you have a SAFR Cloud Account.
- **System requirements**: Ensure that your system meets the minimum system requirements listed here.
- **An up-to-date SAFR License**: See Cloud Licensing for information about SAFR Cloud Licenses.
- **An Internet connection**: Cloud deployments must maintain a network connection with the SAFR Server maintained by RealNetworks in the cloud at all times. Any components that lose their connection to the cloud will immediately lose almost all their functionality.

6.3 Download and Install SAFR Desktop

To download and install SAFR Desktop, do the following:

1. On the computer where you want to install SAFR Desktop, open a web browser and go to the Download Portal.
2. Sign in with your SAFR Cloud Account's credentials.
3. Once signed in, select your operating system from the menu and download the appropriate SAFR Desktop installer.
4. After the download is complete, start the installation.
5. The SAFR Desktop installer displays a *Choose Components* window where you can choose the features you want to install, such as:
    - SAFR Actions (If you choose to install SAFR Actions, ARES will automatically also be installed.)
    - SAFR Video Recognition Gateway (VIRGO)
    - VMS Extensions
        - Avigilon
        - Digifort
        - Genetec Security Center
        - Genetec Security Center with FaceRec
        - Geutebrueck
        - Milestone XProtect
        - Panasonic Video Insight
    - Camera Extensions
        - Intel RealSense
        - Ximea

    You may uncheck the boxes for any features you do not want to install. We recommend installing all the components except the VMS extensions the first time you install SAFR Desktop.
6. Follow the installer prompts as they guide you through the rest of the installation process. The final phase of the installation may take a few minutes to complete as it installs dependencies and runs the configuration scripts. Allow it to continue without interruption.

After the installation finishes, two icons will appear on your desktop: one labeled "SAFRActions" and another labeled "SAFR". *SAFRActions* launches SAFR Actions, while *SAFR* launches the Desktop Client.

Note: Windows SAFR versions earlier than 3.1 are only compatible with NVidia driver versions 418.96 to 431.86.

7 Attendance Log

This solution guide describes how to set up SAFR so that it can be used to record attendance to an event via a self-registration kiosk. New arrivals will go to the kiosk, click the screen to wake it up if necessary, and wait for their name to be displayed. If recognition occurs, then a welcome message is displayed. If recognition does not occur, then the user is prompted to click on the purple oval around their face to register themselves.

For simplicity's sake, this guide assumes that you're using a SAFR Cloud License. If you're using a SAFR On-Premise License, you'll also need to install and manage the SAFR Server.

7.1 Required Hardware and Software

7.1.1 Required Hardware

- One Android tablet for every entrance you want monitored. (iOS tablets could also be used, although some of the details will be slightly different.) See the Mobile tab of SAFR System Requirements for recommended specifications for tablets.

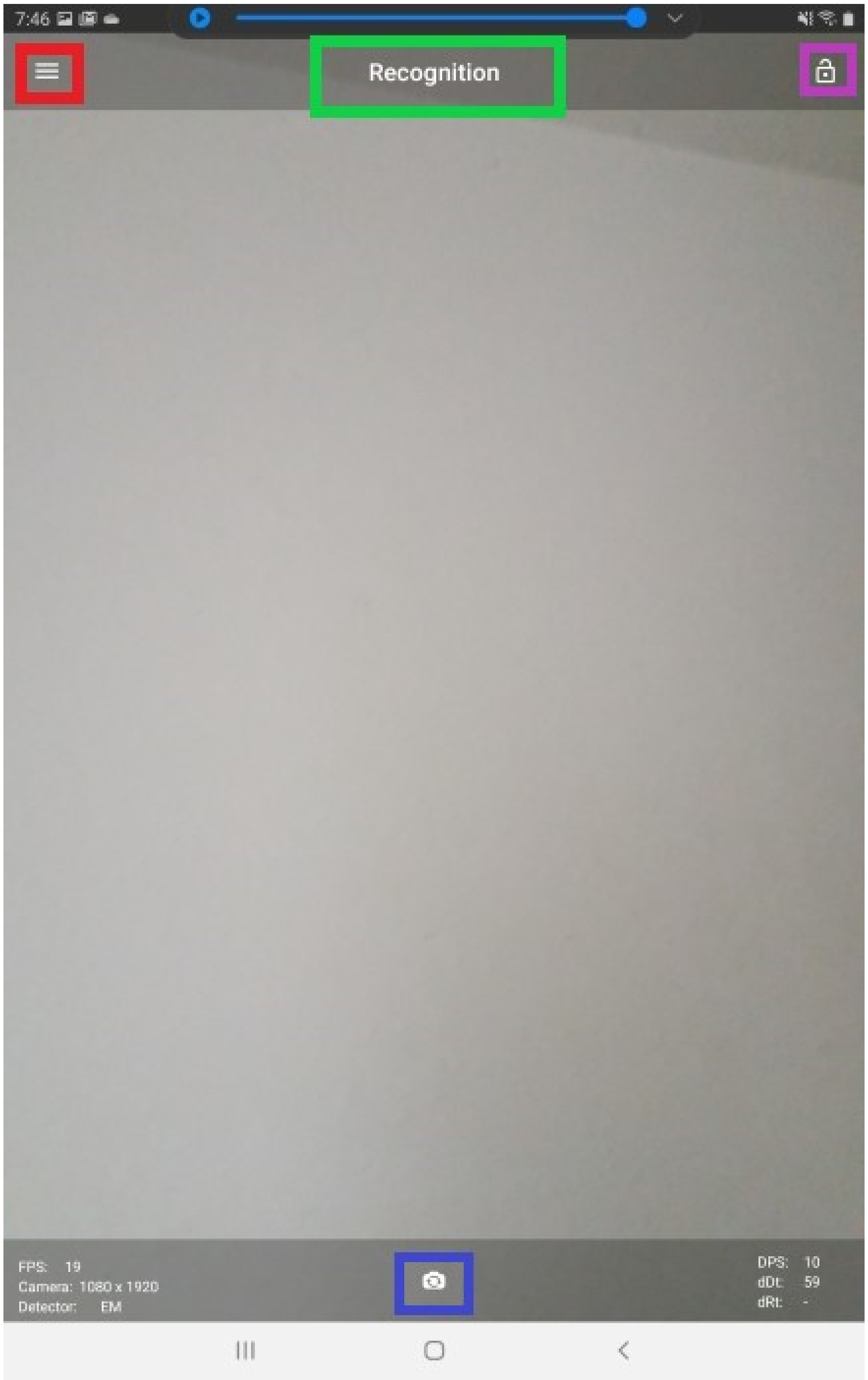
7.1.2 Required Software

- Install SAFR Mobile Client on every Android tablet you're going to use.

7.2 Create Registration Kiosks

For every Android tablet that you're using, do the following:

1. Download and install the SAFR Mobile installer for Android from the SAFR Download Portal.
2. Allow SAFR to take pictures and record video when prompted by the dialog.
3. Enter your SAFR credentials when prompted.



FPS: 19
Camera: 1080 x 1920
Detector: EM

DPS: 10
dDt: 59
dRt: -

4. Tap on the visual processing mode (the text enclosed in the green rectangle) and select **Registration Kiosk** from the pop up menu.
5. Choose whether to use your tablet's default camera (i.e. the camera pointed at the tablet's operator) or its front-facing camera by tapping on the camera selection icon (outlined by the blue rectangle). Because we want users to see and interact with the tablet's screen, select your tablet's default camera.
6. Configure your client's preference settings by tapping on the hamburger icon (outlined by the red rectangle), and then selecting **Settings**. You'll be taken to the preference menus; you can change which preference menu you're configuring by selecting the appropriate icon along the bottom.
   Configure the following settings:
   - **Account Menu**:

        - **User Source** - This setting usually represents the name of the tablet. Enter an appropriate name for each tablet.
        - **User Site** - This setting usually represents the name of your entire facility. This can be useful if, for example, you have multiple stores scattered across the country.
    - **Events Menu**:
        - **Report events** - Enable.
        - **Include Unrecognizable Events** - Disable.
        - **Include Stranger Events** - Disable.
        - **Include Speculated Identity Events** - Disable.
7. To prevent unauthorized access to the tablet, lock the screen by tapping on the icon outlined by the purple rectangle.
8. Your tablet can now function as a registration kiosk.

7.3 Test Your Registration Kiosks

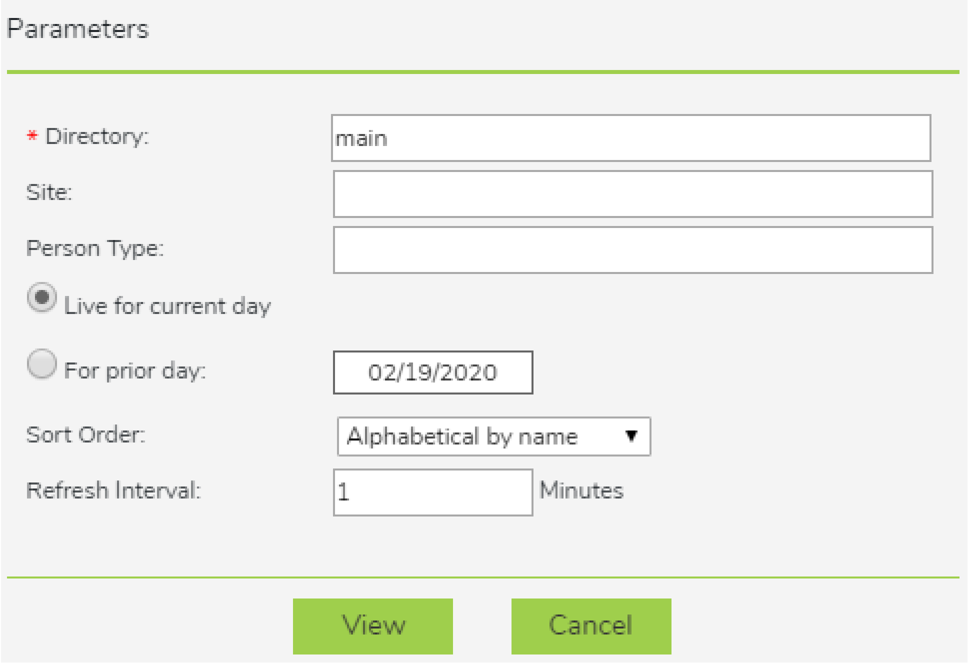
To make sure your registration kiosks are functioning correctly, do the following:

1. Show your face to a registration kiosk and wait for a purple oval to appear around your face. You should also see the text “Tap face to register” below your face.
2. Tap your face. You’ll be prompted to enter your name.
3. Enter your name and tap **Register**.
4. Hide your face from all registration kiosks.
5. How your face to a registration kiosk and wait for a green oval to appear around your face. Your name should appear at the bottom of the oval.
6. You have now verified that your registration kiosks are functioning as expected.

7.4 Reporting

To create a report that summarizes the attendance data for the day, do the following:

1. Go to the SAFR Web Console, located at https://safr.real.com/console/status.
2. Go to the Reports Page by clicking on **Reports** near the top of the page.
3. Click on Attendance Dashboard.
4. You’ll be prompted to enter the parameters for the report. You can accept all the default parameters.



Parameters

* Directory: main

Site:

Person Type:

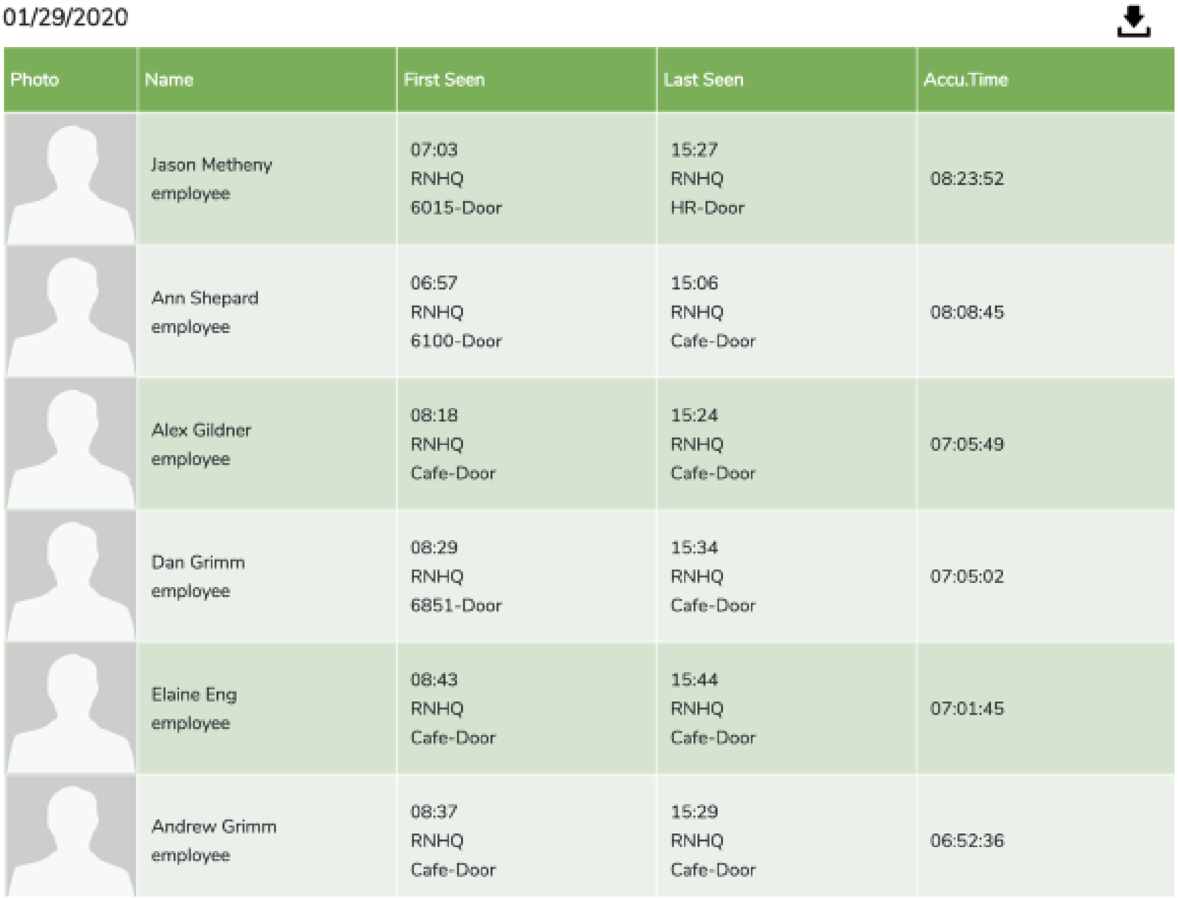
● Live for current day

○ For prior day: 02/19/2020

Sort Order: Alphabetical by name ▼

Refresh Interval: 1 Minutes

5. You'll be presented with a dashboard that is automatically updated every minute with the latest data for the day.



01/29/2020

Photo	Name	First Seen	Last Seen	Accu.Time
	Jason Metheny employee	07:03 RNHQ 6015-Door	15:27 RNHQ HR-Door	08:23:52
	Ann Shepard employee	06:57 RNHQ 6100-Door	15:06 RNHQ Cafe-Door	08:08:45
	Alex Gildner employee	08:18 RNHQ Cafe-Door	15:24 RNHQ Cafe-Door	07:05:49
	Dan Grimm employee	08:29 RNHQ 6851-Door	15:34 RNHQ Cafe-Door	07:05:02
	Elaine Eng employee	08:43 RNHQ Cafe-Door	15:44 RNHQ Cafe-Door	07:01:45
	Andrew Grimm employee	08:37 RNHQ Cafe-Door	15:29 RNHQ Cafe-Door	06:52:36

8 Retail Analytics

This solution guide describes how to set up traffic monitoring for the purpose of understanding the person count in your facility. It uses cameras in your facility to learn and recognize individuals and analyze and record their age, gender, and sentiment. Because people are being learned, SAFR knows when a subject has moved from one camera to another and can report on total individual counts as well as total number of occurrences either at a grouped or individual level.

This guide is written for Windows users. Linux users can achieve the same functionality, but they need to use the Web Console instead of the Desktop Client.

For simplicity's sake, this guide assumes that you're using a SAFR Cloud License. If you're using a SAFR On-Premise License, you'll also need to install and manage the SAFR Server.

8.1 Required Hardware and Software

8.1.1 Required Hardware

- At least one camera for every area you want to monitor. See the Camera Selection section below for information about camera requirements.
- Approximately one PC computer for every 8 cameras that you use. See SAFR System Requirements for recommended specifications for the computers.

8.1.2 Required Software

- Install SAFR Desktop Client on each PC computer that you're going to use.

8.2 Install SAFR Desktop Client

Download and install the SAFR Desktop installer for Windows from the SAFR Download Portal. The SAFR Desktop installer includes both the SAFR Desktop Client and SAFR Actions, but there's no need to install SAFR Actions for this use case.

To configure your Desktop Client, do the following:

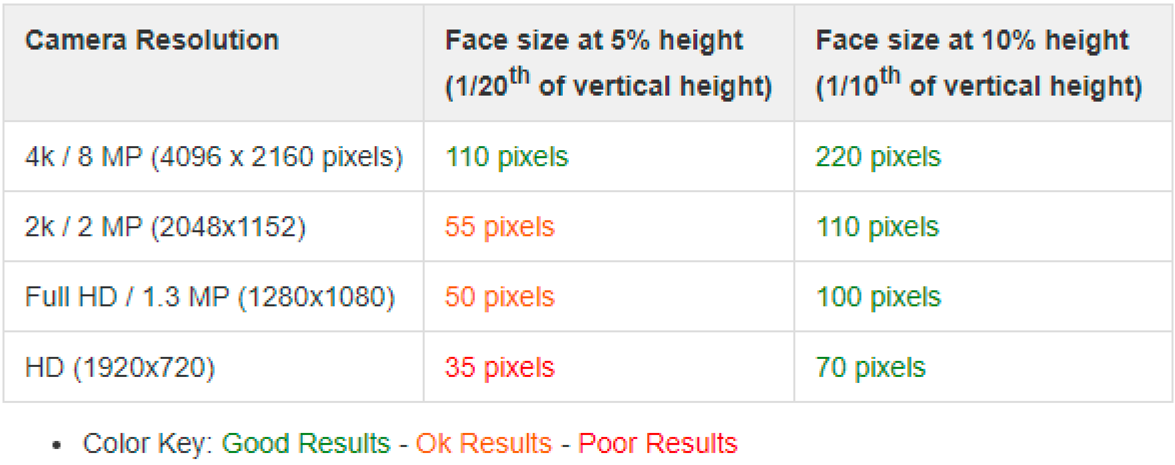
1. Open the SAFR Desktop Client.
2. Select **Preferences...** from the **Tools** drop down menu.
3. Click on the **Recognition** icon at the top of the **Preferences** menu.
4. Set the **For Mode** setting to *Enrolled and Unique Traffic Monitoring.*
5. Scroll down almost to the bottom of the **Recognition menu**, to the **Detect** section.
6. Enable the **Gender** setting. (**Age** and **Sentiment** should already be enabled by default.)

8.3 Camera Selection

Cameras should be located so that subjects' faces aren't backlit; the faces should be well lit from the front with an even and uniform lighting. Direct sunlight on faces should also be avoided in order to reduce sharp contrast, which often hides facial features.

A 1080p USB camera such as the Logitech C920 HD Webcam 1080p Webcam is good enough for this use case. A table top IP camera such as Hikvision DS-2CD2442FWD-IW would also work fine.

Because a wide field of view is typical in these scenarios, it's important to have as high a resolution as possible. Make sure the camera resolution is sufficiently large so that faces are at least 60 pixels tall. (160 pixels is preferable.) The table below provides face sizes for various camera resolutions. Face sizes can be improved by either using larger cameras or by using camera zoom. If you use camera zoom, only use optical zoom. Using digital zoom will negatively impact recognition because it will degrade the quality of image.



Camera Resolution	Face size at 5% height (1/20th of vertical height)	Face size at 10% height (1/10th of vertical height)
4k / 8 MP (4096 x 2160 pixels)	110 pixels	220 pixels
2k / 2 MP (2048x1152)	55 pixels	110 pixels
Full HD / 1.3 MP (1280x1080)	50 pixels	100 pixels
HD (1920x720)	35 pixels	70 pixels

- Color Key: Good Results - Ok Results - Poor Results

Generally a 4k camera or higher is preferable. The larger the face image size the better the result.

8.3.1 Install and Connect your Cameras

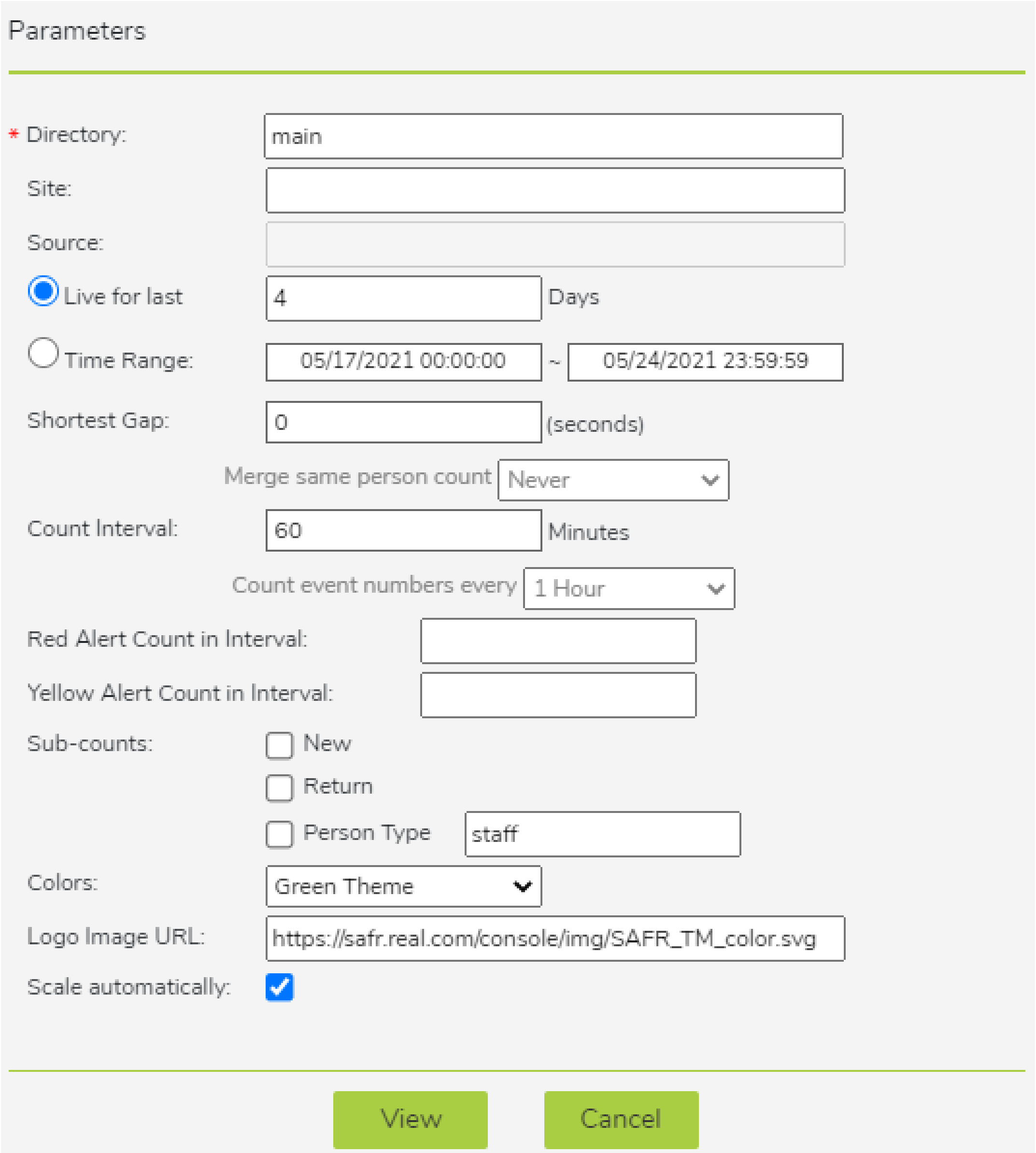
For each camera you install, do the following:

1. Within the SAFR Desktop Client's Camera Feed Analyzer window (i.e. the client's default window), select the camera you want to install.
2. Select the **Enrolled and Unique Traffic Monitoring** video processing mode from the drop-down menu in the upper right corner of the Camera Feed Analyzer window.
3. Optimize the camera location and orientation. Adjust the camera's optical zoom to ensure that subjects' faces are at least 80 pixels high from chin to forehead.
4. Adjust the focus of the camera to the closest position where all subjects' faces are within the camera's field of view. (i.e. ensure that no one will be to the left or right of the camera view)
5. Add the camera's video feed to your Video Recognition Gateway (VIRGO) video feeds by pressing the **Add to Video Feeds** button.
6. Check your system's CPU and GPU load. If you need to install an additional camera, but doing so would increase your CPU or GPU load to 100% or greater, install an additional instance of the SAFR Desktop Client on a different machine and connect the additional camera to the newly installed Desktop Client. Note that you need to configure every Desktop Client as described in the Install SAFR Desktop Client section above.

8.4 Reporting

To create a report that summarizes the traffic data gathered over the previous 4 days, do the following:

1. Go to the SAFR Web Console, located at https://safr.real.com/console/status.
2. Go to the Reports Page by clicking on **Reports** near the top of the page.
3. Click on the first listed report, the Traffic Dashboard.
4. You'll be prompted to enter the parameters for the report. Disable the 3 **Sub-counts** options, but accept all the other default parameter values.



Parameters

* Directory: main

Site:

Source:

● Live for last 4 Days

○ Time Range: 05/17/2021 00:00:00 ~ 05/24/2021 23:59:59

Shortest Gap: 0 (seconds)

Merge same person count Never

Count Interval: 60 Minutes

Count event numbers every 1 Hour

Red Alert Count in Interval:

Yellow Alert Count in Interval:

Sub-counts: □ New

□ Return

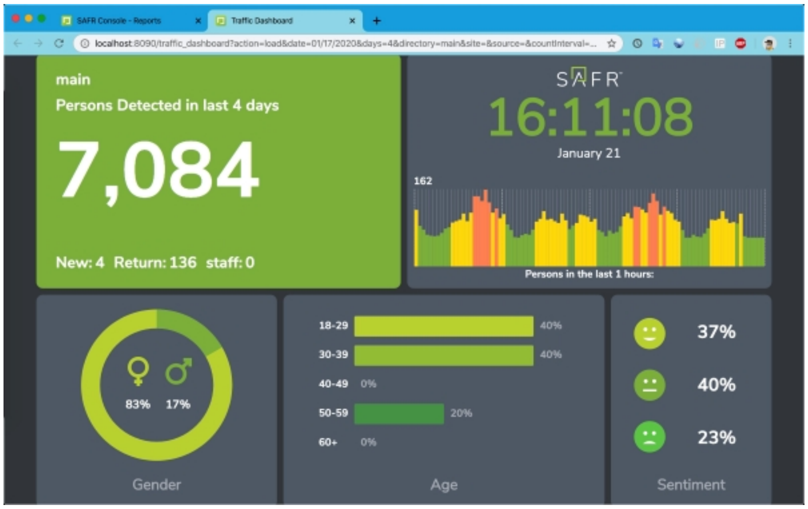
□ Person Type staff

Colors: Green Theme

Logo Image URL: https://safr.real.com/console/img/SAFR_TM_color.svg

Scale automatically: ☑

5. You’ll be presented with a dashboard that is automatically updated with the latest data every couple minutes.


SAFR Console - Reports
Traffic Dashboard
localhost:8090/traffic_dashboard?action=load&date=01/17/2020&days=4&directory=main&site=&source=&countInterval=...
main
Persons Detected in last 4 days
7,084
New: 4 Return: 136 staff: 0
SAFR
16:11:08
January 21
162
Persons in the last 1 hours:
83%
17%
Gender
18-29
40%
30-39
40%
40-49
0%
50-59
20%
60+
0%
Age
37%
40%
23%
Sentiment

8.5 Test Your Retail Analytics System

9 Threat Detection

This solution guide describes how to use SAFR to create an alert if a known threat appears at an entrance monitored by a video camera.

This guide is written for Windows users. Linux users can achieve the same functionality, but they need to use the Web Console instead of the Desktop Client & SAFR Actions GUI.

For simplicity's sake, this guide assumes that you're using a SAFR Cloud License. If you're using a SAFR On-Premise License, you'll also need to install and manage the SAFR Server.

9.1 Required Hardware and Software

9.1.1 Required Hardware

- At least one camera for every entrance you want to monitor. See the Camera Selection section below for information about camera requirements.
- Approximately one PC computer for every 8 cameras that you use. See SAFR System Requirements for recommended specifications for the computers.
- You may want doors that can be locked electronically, in case you want to automatically lock all entrances to your facility if a known threat is detected.

9.1.2 Required Software

- Install SAFR Desktop Client on each PC computer you're going to use.
- Install SAFR Actions on one (and only one) of the SAFR Desktop Client machines.

9.2 Install SAFR Desktop Client and SAFR Actions

Download and install the SAFR Desktop installer for Windows from the SAFR Download Portal. The SAFR Desktop installer includes both the SAFR Desktop Client and SAFR Actions. Note that when you install SAFR Actions, both of the following are installed:

- Actions Relay Event Service (ARES) - A Java application that acts as an event listener that dispatches configured actions in response to events.
- SAFR Actions GUI - A GUI that helps users manage ARES.

If you have many cameras, you may need to install multiple Desktop Clients on different machines. Typically each machine can handle up to 8 cameras. If you do install multiple Desktop Clients, don't install ARES (or SAFR Actions) on the additional clients; you should only install one ARES instance.

9.3 Camera Selection and Positioning

Because the goal of this solution guide is to detect when a known threat attempts to enter the facility, every entrance to the facility should have at least one camera pointed at it. Camera placement should follow these guidelines:

- In most cases, cameras should be about 10-12 feet off the ground and about 20-30 feet from the entrance.
- Subjects should be facing nearly straight on.
- Camera resolutions should be such that faces are at least 120 pixels high when at the entrance. (Larger than 120 pixels would be even better.)
- Camera lenses should have sun/rain shades.

9.3.1 Lighting Considerations

Success with recognition largely depends on lighting conditions.

- At nighttime and indoors there should be a light source behind the camera so that subjects' faces are clearly illuminated.

- Avoid a direct line between the sun and your cameras' lenses.

Backlighting, which is when light shines from behind a subject, can cause significant difficulties in recognition and should be avoided whenever possible. However, because backlighting often can't be avoided, the cameras you use should have the following features which help alleviate backlighting difficulties:

- Shutter priority mode
- Exposer compensation
- Manual mode with full shutter speed, iris (aperture), and gain adjustment

The Sony 772R is one example of a camera with all of the features above. In addition to these features the 772R also offers various digital tools to enhance the image:

- Visibility enhancer
- Backlight compensation
- Highlight compensation
- Noise reduction

The basic steps in handling a backlight situation are as follows:

1. Angle the camera slightly toward the floor to eliminate as much direct light into the camera's sensor as possible.
2. Because people will be moving through the field of view (FOV), maintaining high shutter speed is important to obtain blur-free images. To help accomplish this, place a bound on the slowest shutter speed allowed.
3. Turn on backlight compensation if available (it's available on the 772R), and adjust the level.
4. If faces are still dark, adjust the exposure compensation to brighten the faces. The background will probably become overexposed, but that's OK for facial recognition.
5. If there are specular reflections or if faces are still too dark, turn on highlight compensation.

The previous approach is appropriate for the situation where varying outdoor conditions also vary the amount of light reflected from the face. Light intensity is simply boosted above what the cameras would choose automatically and enhancing the image to reduce exposer variance.

In cases where outdoor conditions generate backlight conditions and there is minimal variation in lighting from inside (e.g. there are few windows, so indoor illumination doesn't affect the lighting on subjects' faces), it's more appropriate to place the camera in fully manual mode and set the shutter speed, iris, and gain values manually to properly expose the face while allowing the background to be overexposed. When in manual mode, the camera makes no auto-adjustments and is not thrown off by momentary bursts of light due to door opening or other momentary reflections. When setting your camera in manual mode, do the following:

1. Set the shutter speed to 1/90 or higher.
2. Open the iris, increasing the f-stop for the iris (aperture) until subjects' faces are bright enough.
3. Focus the camera on the sweet spot of the recognition where people are most likely to face towards the camera.

Increasing the iris reduces the distance during which subjects' faces are in focus. (This is called the depth of field.) Increasing the iris increases the quality of the image but reduces the amount of time the image is in focus and viability for recognition. In either case, focus the camera on the sweet spot of the recognition where people are most likely to face toward the camera.

9.3.2 Install and Connect your Cameras

For each camera you install, do the following:

1. Within the SAFR Desktop Client's Camera Feed Analyzer window (i.e. the client's default window), select the camera you want to install.
2. Select the **Secure Access** video processing mode from the drop-down menu in the upper right corner of the Camera Feed Analyzer window.

3. Optimize the camera location and orientation. Adjust the camera's optical zoom to ensure that subjects' faces are at least 80 pixels high from chin to forehead.
4. Adjust the focus of the camera to the closest position where all subjects' faces are within the camera's field of view. (i.e. ensure that no one will be to the left or right of the camera view)
5. Add the camera's video feed to your Video Recognition Gateway (VIRGO) video feeds by pressing the **Add to Video Feeds** button.
6. Check your system's CPU and GPU load. If you need to install an additional camera, but doing so would increase your CPU or GPU load to 100% or greater, install an additional instance of the SAFR Desktop Client on a different machine and install the additional camera on the newly installed Desktop Client.

9.3.3 Additional Camera Resources

- Anyone setting up cameras for facial recognition should be fully familiar with digital photography concepts described in the video Aperture, Shutter Speed, ISO, & Light Explained-Understanding Exposure & Camera Settings. (The video is 15 minutes long.)
- Specifics on backlight compensation can be found here: https://www.dpmag.com/how-to/tip-of-the-week/combat-backlighting-with-exposure-compensation/
- The SONY 772R User Guide

9.4 Register Threats

9.4.1 Register a Person Into SAFR Using a Photo

To register a person from a photo, do the following:

1. Open the SAFR Desktop Client.
2. Navigate to the People Window by selecting **People...** from the **Tools** drop-down menu.
3. Click the **Add face** button near the top of the People Window.
4. Select the photo image located on your hard drive.
5. Any photo that has sufficient quality for recognition will show a purple oval around the face with the option to click and add a name. Type a name if desired.

Note: It's very helpful to use as high quality a photo as possible.

9.4.2 Register a Person Into SAFR Using a Video

You can also register people from a saved video file by doing the following:

1. Open the SAFR Desktop Client.
2. Select the **Import** video processing mode from the drop-down menu in the upper right corner of the Camera Feed Analyzer window.
3. Choose **File** > **Open**.
4. Open a video file.
5. Any face within the video that has sufficient image quality for recognition will be added to the Identity Database.
6. If you want, you can add names to the newly registered people by doing the following:
   1. Navigate to the People Window by selecting **People...** from the **Tools** drop-down menu.
   2. Sort by *Enrollment Date*, and set sort order to *Descending* in order to see the newly added entries.
   3. Double click the users whose names you want to set, and enter their names.

9.4.3 Configure a Registered Person as a Threat

To configure a registered person as a threat, do the following:

1. Open the SAFR Desktop Client.
2. Navigate to the People Window by selecting **People...** from the **Tools** drop-down menu.

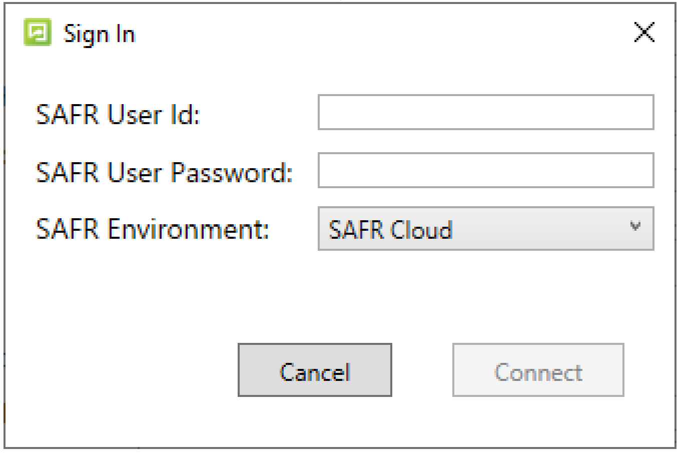
3. If you want to see most recently added people, sort by *Enrollment Date*, and set sort order to *Descending* in order to see the newly added entries.
4. You can remove people already marked as *Threat* or *Concern* by filtering on ID Class for *No Concern*.
5. For each user you want to configure as a thread:
    1. Double-click the user.
    2. Choose *Threat* or *Concern* from the **ID Class** menu list.
    3. In the Desktop Client, that person will now be marked with a red oval overlay to indicate a Threat or an amber oval overlay to indicate a Concern.

9.5 Configure SAFR to Send Notifications

9.5.1 Configure SAFR to Send Emails

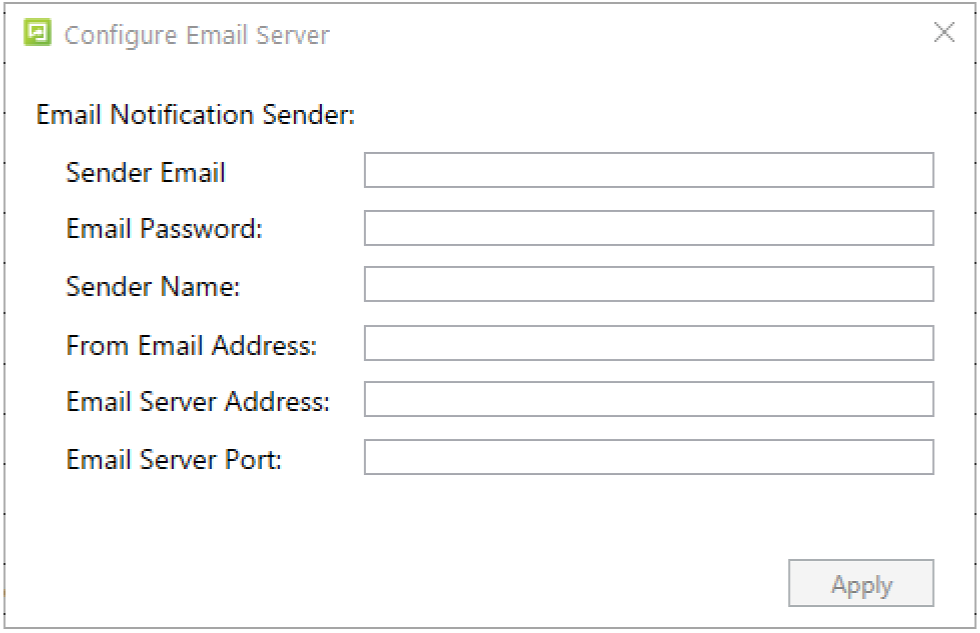
Configure SAFR so that it can use an email server to send email messages.

1. Get an SMTP Server account you can use for sending emails.
2. Within the SAFR Actions GUI, select **Configure Email Server...** from the **Tools** drop down menu at the top.
3. You'll be prompted to enter your SAFR credentials.



SAFR Environment: SAFR Cloud

4. You'll be presented with the following dialogue. Fill out all the fields of the dialogue.

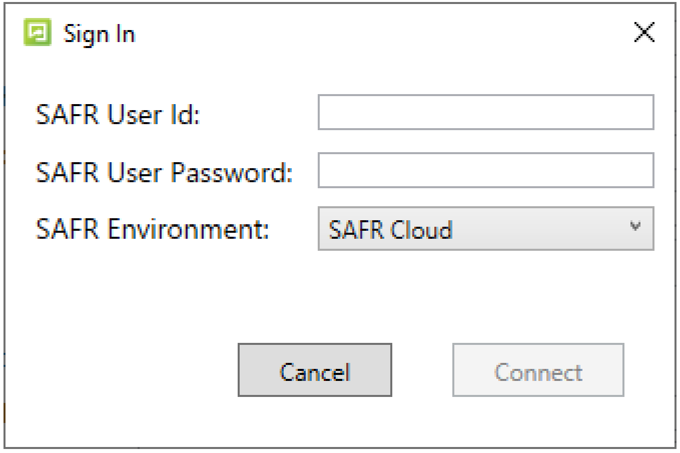


- **Sender Email**: The email username of the SMTP account. (e.g. Susan.Johnson@gmail.com)

   - **Email Password**: The password for the SMTP account.
   - **Sender Name**: The display name on the "From" line. (e.g. Susan Johnson)
   - **From Email Address**: The email address that will appear on the "From" line. This feature isn't supported by all email servers; if this field isn't used then the *Sender Email* value is used for the "From" line.
   - **Email Server Address**: The address of the SMTP email server.
   - **Server Port**: The email server port. The default port for SMTP is 587.
5. Click **Apply**. ### Configure SAFR to Send SMS Messages

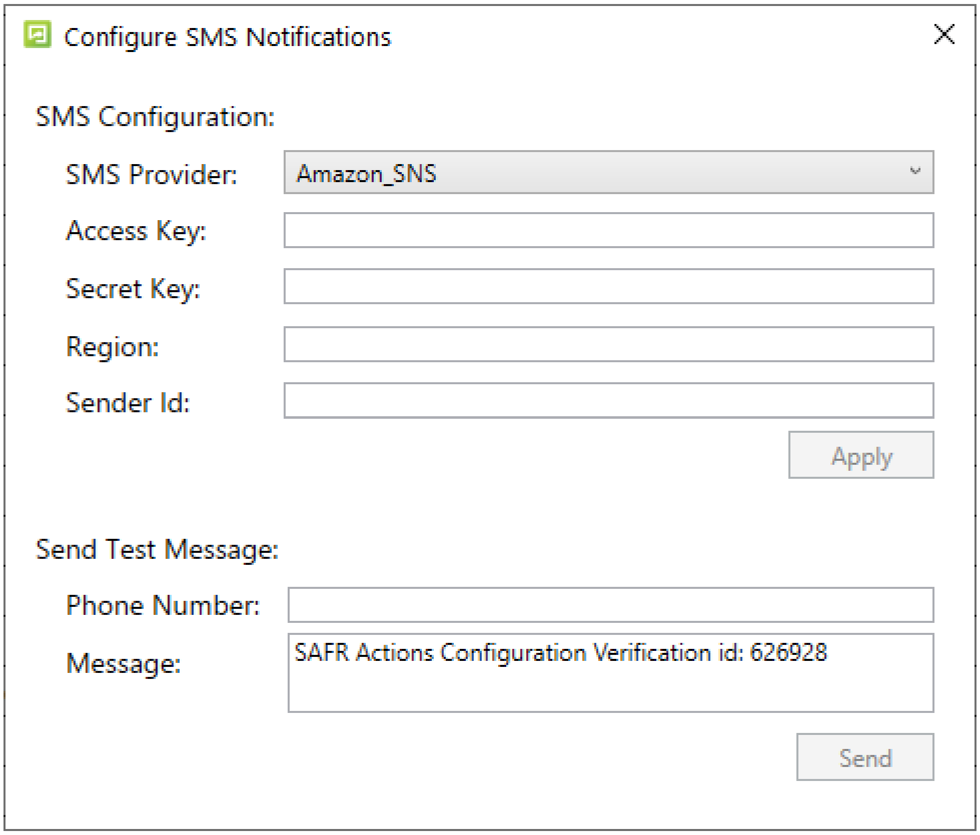
Configure SAFR so that it can use a short message service (SMS) server to send SMS messages.

1. Set up an AWS account. Make sure your AWS account is configured for your region so that it can send SMS messages.
2. Within the SAFR Actions GUI, select **Configure SMS Sender...** from the **Tools** drop down menu at the top.
3. You'll be prompted to enter your SAFR credentials.



SAFR Environment: SAFR Cloud

4. You'll be presented with the following dialogue. Fill out all the fields of the dialogue.



Configure SMS Notifications

SMS Configuration:

SMS Provider: Amazon_SNS

Access Key:

Secret Key:

Region:

Sender Id:

Send Test Message:

Phone Number:

Message: SAFR Actions Configuration Verification id: 626928

- **SMS Provider**: The SMS provider that you're using. This value will always be `Amazon_SNS`.
- **Access Key**: Your Amazon SNS Access Key.
- **Secret Key**: Your Amazon SNS Secret Key.
- **Region**: The region of your Amazon SNS.
- **Sender Id**: The name that will be used to send the SMS messages.
- **Send Test Message**: Configure the test message that will be sent after you finish setting up SMS.
  - **Phone Number**: The phone number to which the test message will be sent. It should be in the E.164 format. (e.g. +2065551313)
  - **Message**: The text message that will be sent to the phone number specified above.

5. Click **Apply**, then click **Send** ## Configure ARES

Note: Make sure the SAFR Actions GUI is not running during steps 1-3 below.

On the machine where ARES is installed:

1. Delete the default SAFRActions.config file located at `C:\ProgramData\RealNetworks\SAFR\ares\config`.
2. Save ThreatSAFRActions.config to `C:\ProgramData\RealNetworks\SAFR\ares\config`.
3. Rename the config file you just saved to *SAFRActions.config*.
4. Start the SAFR Actions GUI.
5. Within the SAFR Actions GUI, set the following fields to the specified values:
   - **userId**: Your username.
   - **userPwd**: Your password.
   - **emailDef -> Item 1 -> recipients -> Item 1**: The email address of whomever you want to receive email notifications. If you want more than 1 person to receive email notifications, do the following:
     1. Hover your mouse over *recipients*.
     2. Press the + button to create additional recipients.
     3. Set the newly created item(s) to the additional email address(es).

- **smsDef -> Item 1 -> recipients -> Item 1**: The phone number of whomever you want to receive SMS notifications. It should be in the E.164 format. (e.g. +2065551313) If you want more than 1 person to receive SMS notifications, do the following:
    1. Hover your mouse over *recipients*.
    2. Press the + button to create additional recipients.
    3. Set the newly created item(s) to the additional phone number(s).

6. Choose **File** > **Save** to save your changes.

Note: If ARES is installed on a different machine from a SAFR Desktop Client, make sure the system clocks on the Desktop Client and ARES machines are within a few seconds of each other. (If the two system clocks differ by more than a few seconds, events may not trigger.)

9.6 Test Your Threat Detection System

To test the system that you have just set up, do the following:

1. Register yourself with SAFR and configure yourself as a threat, as described above.
2. Hide your face from all connected cameras.
3. Show your face to a connected camera and wait for SAFR to recognize you.
4. Upon recognition, you should see the following:
    - A red oval should be drawn around your face in SAFR's video feed, indicating that you're a threat.
    - The message "Threat Detected" should flash on the screen.
    - You should see an email appear in your inbox.
    - Your phone should receive an SMS message.

If you don't see the red oval or the on-screen message, try the following troubleshooting steps:

1. If you see errors with the login, ensure that your account information is correct.
2. If you see a "Too Late" message, the machine running the Desktop Client and the machine running ARES may not be set to the same time. Correct this and try again.
3. Check the ARES log (located at `C:\ProgramData\RealNetworks\SAFR\ares\logs`) to see if there are any clues there.
4. Contact RealNetworks technical support.

10 Visitor Announcement

This solution guide describes how to set up a visitor announcement system which will send a notification message to a receptionist or sales assistant when a registered person arrives at the entrance of your office.

This guide is written for Windows users. Linux users can achieve the same functionality, but they need to use the Web Console instead of the Desktop Client & SAFR Actions GUI.

For simplicity's sake, this guide assumes that you're using a SAFR Cloud License. If you're using a SAFR On-Premise License, you'll also need to install and manage the SAFR Server.

10.1 Required Hardware and Software

10.1.1 Required Hardware

- One camera for every entrance you want the visitor announcement system to cover. See the Camera Selection section below for information about camera requirements.
- Approximately one PC computer for every 8 cameras that you use. See SAFR System Requirements for recommended specifications for the computers.

10.1.2 Required Software

- Install SAFR Desktop Client on each PC computer you're going to use.
- Install SAFR Actions on one (and only one) of the SAFR Desktop Client machines.

10.2 Install SAFR Desktop Client and SAFR Actions

Download and install the SAFR Desktop installer for Windows from the SAFR Download Portal. The SAFR Desktop installer includes both the SAFR Desktop Client and SAFR Actions. Note that when you install SAFR Actions, both of the following are installed:

- Actions Relay Event Service (ARES) - A Java application that acts as an event listener that dispatches configured actions in response to events.
- SAFR Actions GUI - A GUI that helps users manage ARES.

If you have many cameras, you may need to install multiple Desktop Clients on different machines. Typically each machine can handle up to 8 cameras. If you do install multiple Desktop Clients, don't install ARES (or SAFR Actions) on the additional clients; you should only install one ARES instance.

10.3 Camera Selection

Cameras should be located so that subjects' faces aren't backlit; the faces should be well lit from the front with an even and uniform lighting. Direct sunlight on faces should also be avoided in order to reduce sharp contrast, which often hides facial features.

A 1080p USB camera such as the Logitech C920 HD Webcam 1080p Webcam is good enough for this use case. A table top IP camera such as Hikvision DS-2CD2442FWD-IW would also work fine.

10.3.1 Install and Connect your Cameras

For each camera you install, do the following:

1. Within the SAFR Desktop Client's Camera Feed Analyzer window (i.e. the client's default window), select the camera you want to install.
2. Select the **Enrolled and Stranger Monitoring** video processing mode from the drop-down menu in the upper right corner of the Camera Feed Analyzer window.
3. Optimize the camera location and orientation. Adjust the camera's optical zoom to ensure that subjects' faces are at least 80 pixels high from chin to forehead.

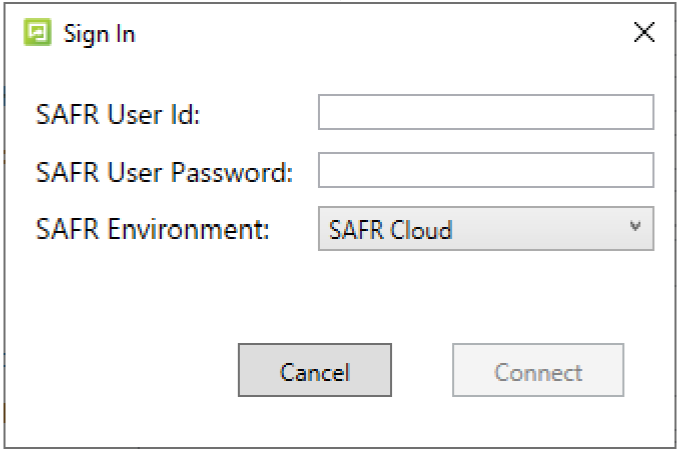
4. Adjust the focus of the camera to the closest position where all subjects' faces are within the camera's field of view. (i.e. ensure that no one will be to the left or right of the camera view)
5. Add the camera's video feed to your Video Recognition Gateway (VIRGO) video feeds by pressing the **Add to Video Feeds** button.
6. Check your system's CPU and GPU load. If you need to install an additional camera, but doing so would increase your CPU or GPU load to 100% or greater, install an additional instance of the SAFR Desktop Client on a different machine and install the additional camera on the newly installed Desktop Client.

10.4 Configure SAFR to Send Notifications

10.4.1 Configure SAFR to Send Emails

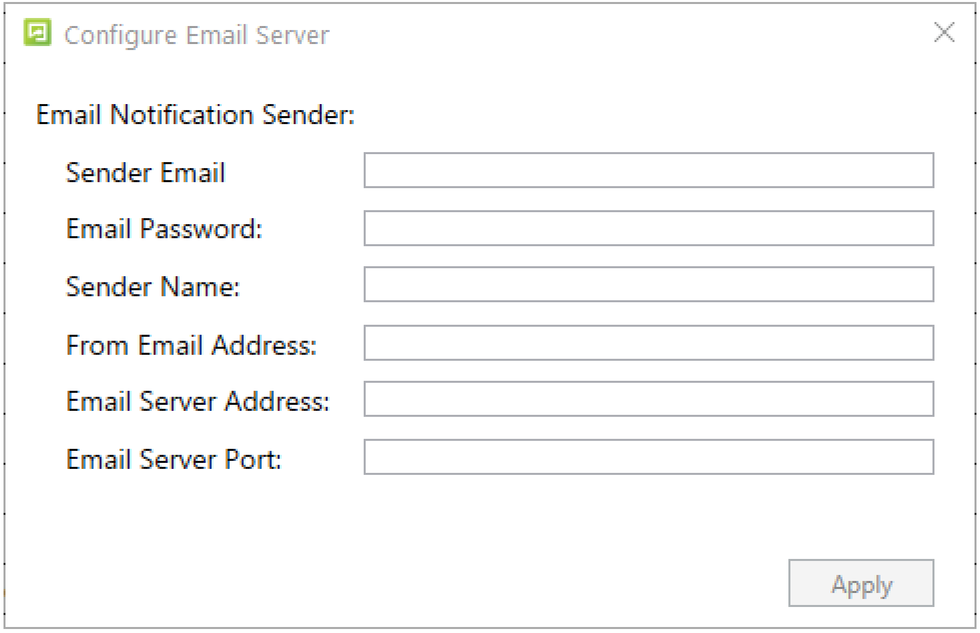
Configure SAFR so that it can use an email server to send email messages.

1. Get an SMTP Server account you can use for sending emails.
2. Within the SAFR Actions GUI, select **Configure Email Server. . .** from the **Tools** drop down menu at the top.
3. You'll be prompted to enter your SAFR credentials.



SAFR Environment: SAFR Cloud

4. You'll be presented with the following dialogue. Fill out all the fields of the dialogue.

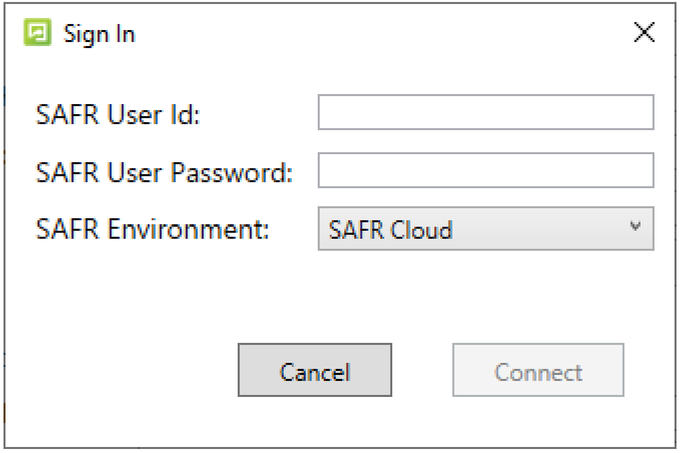


- **Sender Email**: The email username of the SMTP account. (e.g. Susan.Johnson@gmail.com)

   - **Email Password**: The password for the SMTP account.
   - **Sender Name**: The display name on the “From” line. (e.g. Susan Johnson)
   - **From Email Address**: The email address that will appear on the “From” line. This feature isn’t supported by all email servers; if this field isn’t used then the *Sender Email* value is used for the “From” line.
   - **Email Server Address**: The address of the SMTP email server.
   - **Server Port**: The email server port. The default port for SMTP is 587.
5. Click **Apply**. ### Configure SAFR to Send SMS Messages

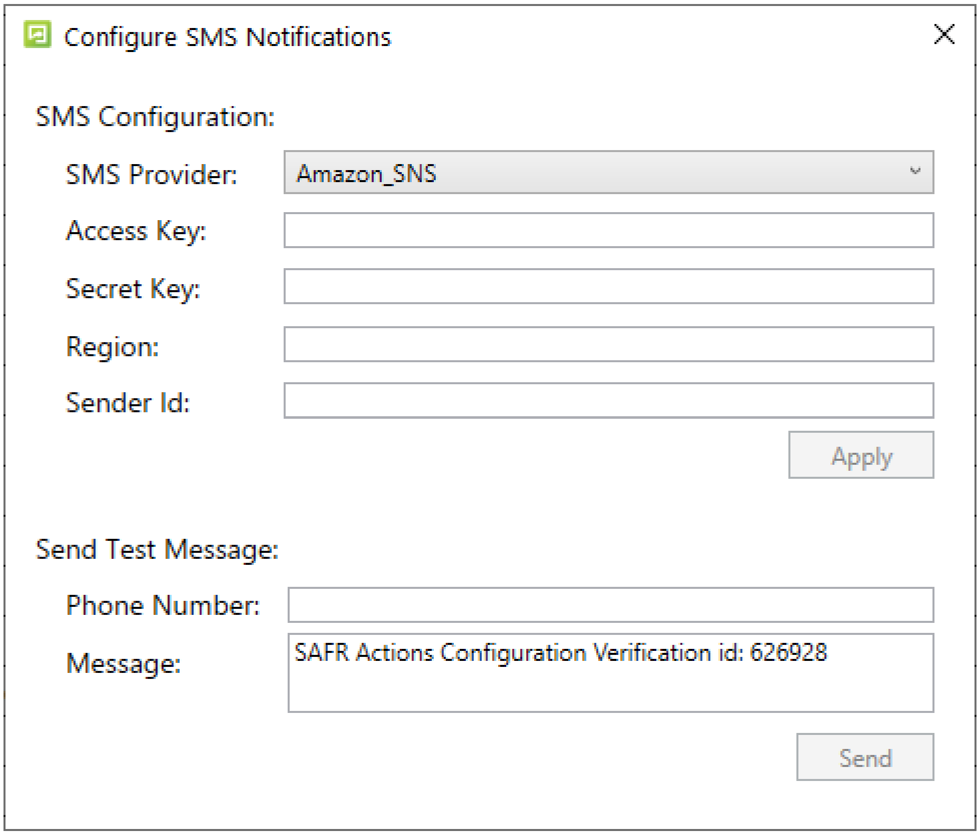
Configure SAFR so that it can use a short message service (SMS) server to send SMS messages.

1. Set up an AWS account. Make sure your AWS account is configured for your region so that it can send SMS messages.
2. Within the SAFR Actions GUI, select **Configure SMS Sender…** from the **Tools** drop down menu at the top.
3. You’ll be prompted to enter your SAFR credentials.



SAFR Environment: SAFR Cloud

4. You’ll be presented with the following dialogue. Fill out all the fields of the dialogue.


Configure SMS Notifications

SMS Configuration:

SMS Provider: Amazon_SNS

Access Key:

Secret Key:

Region:

Sender Id:

Send Test Message:

Phone Number:

Message: SAFR Actions Configuration Verification id: 626928

- **SMS Provider**: The SMS provider that you're using. This value will always be `Amazon_SNS`.
- **Access Key**: Your Amazon SNS Access Key.
- **Secret Key**: Your Amazon SNS Secret Key.
- **Region**: The region of your Amazon SNS.
- **Sender Id**: The name that will be used to send the SMS messages.
- **Send Test Message**: Configure the test message that will be sent after you finish setting up SMS.
  - **Phone Number**: The phone number to which the test message will be sent. It should be in the E.164 format. (e.g. +2065551313)
  - **Message**: The text message that will be sent to the phone number specified above.

5. Click **Apply**, then click **Send** ## Configure ARES

Note: Make sure the SAFR Actions GUI is not running during steps 1-3 below.

On the machine where ARES is installed:

1. Delete the default SAFRActions.config file located at `C:\ProgramData\RealNetworks\SAFR\ares\config`.
2. Save VisitorSAFRActions.config to `C:\ProgramData\RealNetworks\SAFR\ares\config`.
3. Rename the config file you just saved to *SAFRActions.config*.
4. Start the SAFR Actions GUI.
5. Within the SAFR Actions GUI, set the following fields to the specified values:
   - **userId**: Your username.
   - **userPwd**: Your password.
   - **emailDef -> Item 1 -> recipients -> Item 1**: The email address of whomever you want to receive email notifications. If you want more than 1 person to receive email notifications, do the following:
     1. Hover your mouse over *recipients*.
     2. Press the + button to create additional recipients.
     3. Set the newly created item(s) to the additional email address(es).

- **emailDef** -> **Item 2** -> **recipients** -> **Item 1**: The email address of whomever you want to receive email notifications. If you want more than 1 person to receive email notifications, do the following:
  1. Hover your mouse over *recipients*.
  2. Press the + button to create additional recipients.
  3. Set the newly created item(s) to the additional email address(es).
- **smsDef** -> **Item 1** -> **recipients** -> **Item 1**: The phone number of whomever you want to receive SMS notifications. It should be in the E.164 format. (e.g. +2065551313) If you want more than 1 person to receive SMS notifications, do the following:
  1. Hover your mouse over *recipients*.
  2. Press the + button to create additional recipients.
  3. Set the newly created item(s) to the additional phone number(s).
- **smsDef** -> **Item 2** -> **recipients** -> **Item 1**: The phone number of whomever you want to receive SMS notifications. It should be in the E.164 format. (e.g. +2065551313) If you want more than 1 person to receive SMS notifications, do the following:
  1. Hover your mouse over *recipients*.
  2. Press the + button to create additional recipients.
  3. Set the newly created item(s) to the additional phone number(s).

6. Choose **File** > **Save** to save your changes.

Note: If ARES is installed on a different machine from a SAFR Desktop Client, make sure the system clocks on the Desktop Client and ARES machines are within a few seconds of each other. (If the two system clocks differ by more than a few seconds, events may not trigger.)

10.5 Test Your Visitor Announcement System

To test the “StrangerDetected” notifications of the system that you have just set up, do the following:

1. Open the SAFR Actions GUI and make sure that your email address and phone number are listed as recipients of the “StrangerDetected” email and SMS notifications, as described in the Configure ARES section above.
2. Hide your face from all connected cameras.
3. Show your face to a connected camera and wait for SAFR to recognize you.
4. Upon recognition, you should see the following:
   - You should receive a “StrangerDetected” email notification in your inbox.
   - Your phone should receive a “StrangerDetected” SMS message.

To test the “RegisteredDetected” notifications, do the following:

1. Register yourself into SAFR.
2. Open the SAFR Actions GUI and make sure that your email address and phone number are listed as recipients of the “RegisteredDetected” email and SMS notifications, as described in the Configure ARES section above.
3. Hide your face from all connected cameras.
4. Show your face to a connected camera and wait for SAFR to recognize you.
5. Upon recognition, you should see the following:
   - You should receive a “RegisteredDetected” email notification in your inbox.
   - Your phone should receive a “RegisteredDetected” SMS message.

11 Camera Best Practices

Although it's sometimes possible to make facial recognition work on existing cameras, you should be methodical about attempting to re-use existing cameras for facial recognition. A thorough survey of locations you want to monitor should be performed to determine the goals at each location and the requirements needed to achieve those goals. Only then should you take stock of existing cameras to see if they meet the requirements to meet your goals. If not, new cameras should be employed that allow you to meet your goals.

11.1 Where to Set Up Your Cameras

The best results with facial recognition generally happen when you set up your facial recognition cameras at choke points such as narrow passages (e.g. doorways) or concentrated standing areas (e.g. bus stops).

Below are some considerations when evaluating for choke points:

- Look for places where people are traveling slower or are stationary.
- Find places where people are facing a consistent direction.
- The narrower the choke point, the more pixels that can be devoted to the face.
    - A door that's 6m wide yields half the pixels as a door that's 3m wide.
- Lighting is critical. The lighting conditions section below goes into more detail about desired lighting.

Example choke points:

Scenario	Optimal Location	Challenges	Potential Mitigation Strategies
Doorway, elevator exit, or gateway	• Exit side of door/elevator, 5-10m away, and 3-4m high. • If a wall or post is 3-4m away, then you can place the camera 2.5m high.	• Subjects turn left/right as they pass thru doorway/exit elevator. • Subjects look left/right as they pass thru doorway/exit elevator. • Strong backlight (not applicable for elevator). • Automatic doors cause sudden changes in lighting.	• If possible, avoid backlight conditions. If not possible, add more light to subject's faces to counter the backlight. • Use a camera with good Wide Dynamic Range (WDR) performance.

Scenario	Optimal Location	Challenges	Potential Mitigation Strategies
Hallway	• At end of hall where subjects turn left or right 2.5m high and 2-4m back. • Hung from ceiling 3-4m high and 5-10m back.	• Tall ceilings.	• If mounted on a wall, consider mounting on wall but target subjects more than 10m away use the camera's optical zoom. • If poor lighting, use SAFR's Contrast Enhancement feature.
Stairway/escalator	• 2-4m from top of stair/elevator pointing down, parallel to stairs. • Subjects tend to look up when going up so a higher camera position is OK.	• Poor lighting	• If poor lighting, use SAFR's Contrast Enhancement feature.
Front queue	• Exit side of queue 5-10m away and 3-4m high in line with queue. • Queues where subjects stand and wait.	• Subjects turn left/right as they exit queue. • Moving stations (e.g. airports).	• Add object of interest such as a TV monitor to draw eyes towards camera.
Near artwork or other objects of interest	• Centered above an object of interest.	• Distance from object. • Wide field of view.	• Use a high resolution camera. • Target a narrow field of view.

11.2 Camera Facial Recognition Factors

Below are the key factors that affect the success of camera facial recognition.

- Face Image Size – The number of pixels that are present in a facial image.
  - Video Resolution – The width and height of a video, measured in pixels.
  - Angle of View – Determined by the angle of the camera lens.
  - Distance to Subject – The distance from the camera to the subject of interest.
- Sharpness – The degree to which edges remain crisp and pixels are not blurred together.
  - Focus – The degree to which camera image is sharp.
  - Depth of Field – The distance between the nearest and the furthest objects that can be in focus for a camera at the same time.
  - Video Compression – The process of encoding video files such that they consume less space and are easier to transmit over the network. Video compression can have the effect of blurring video

  images, however.

- Lighting Conditions – Adequate lighting conditions are critical for successful facial recognition. There are two aspects of lighting that are particularly important:
  - Backlight – Bright lighting behind the subject of interest.
  - Low Light – Nighttime or dim indoor environments.
- Center Pose Quality – A value from 0 to 1 that specifies how directly a face is looking at the camera.
  - Yaw Angle (horizontal) – The horizontal angle between the subject's gaze and the direct line to the camera.
  - Depression Angle (vertical) – The vertical angle between the subject and the camera.
- Data Rate – Data processing factors that affect performance and quality.
  - Frame Rate – Number of video frames delivered by the camera per second.
  - Video Bitrate – The amount of data allocated to the digitized video, measured in bits per pixel (bps).

11.2.1 Face Image Size

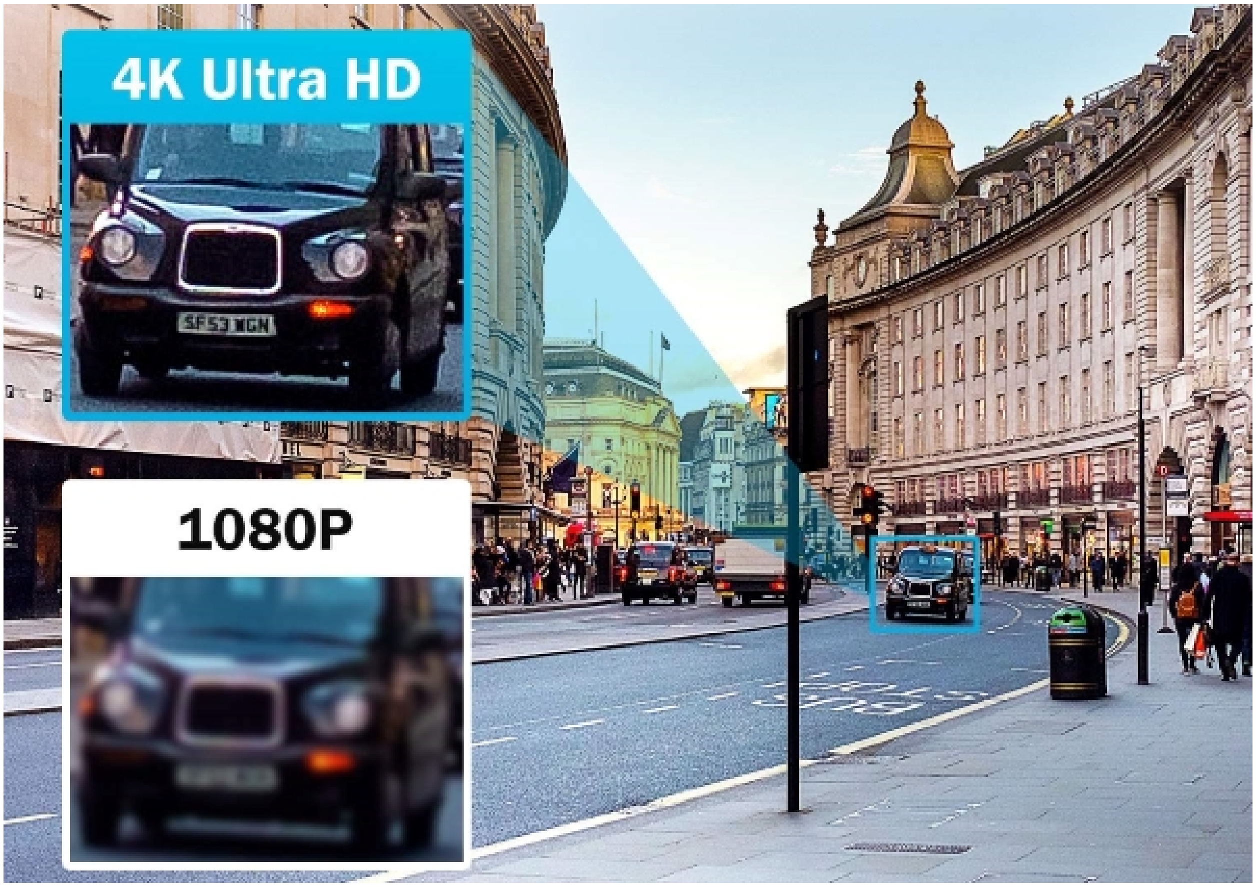
The number of pixels a camera allocates to a face is determined by three main variables, listed in order of impact:

- Video resolution
- Angle of view
- Distance to subject

11.2.1.1 Video Resolution Video resolution describes the number of pixels in each video frame. Video resolution is measured as width x height (in that order). For convenience, some people only cite the height measurement when talking about video resolution. Thus, cameras with resolutions of 1920x1080 might be said to have 1080p resolution.

Obviously, the higher the camera's video resolution, the better. The minimum video resolution that we recommend for successful facial recognition is about 2500p. Below you can see the effect of different video resolutions.

Note: "4K Ultra HD" has a resolution of approximately 2500p.



As you can see, the license plate is much more legible in the higher resolution.

11.2.1.2 Angle of View Angle of View (AoV) is another significant factor impacting face image size; it can increase the face image size by orders of magnitude. A camera with a wide AoV will spread its limited number of pixels over a wide area, a problem which increases dramatically as subjects get further from the camera. Conversely, a small AoV will retain the number of pixels it can use for face image size even as the distance increases.

Wide-angled lenses tend to be bad for facial recognition. They introduce significant perspective distortion, as well as requiring closer distances for accurate results.

Cameras' AoVs are usually reported in their camera specifications sheets. You can also get this information from tools such as IPVM.com calculator.

11.2.1.3 Distance to Subject This is the distance from the subject to the camera lens. Obviously, you want the camera to be as close to subjects as possible.

Cameras' zoom functionality can mitigate distance from the subject. Be aware that there are fundamentally two different types of zoom:

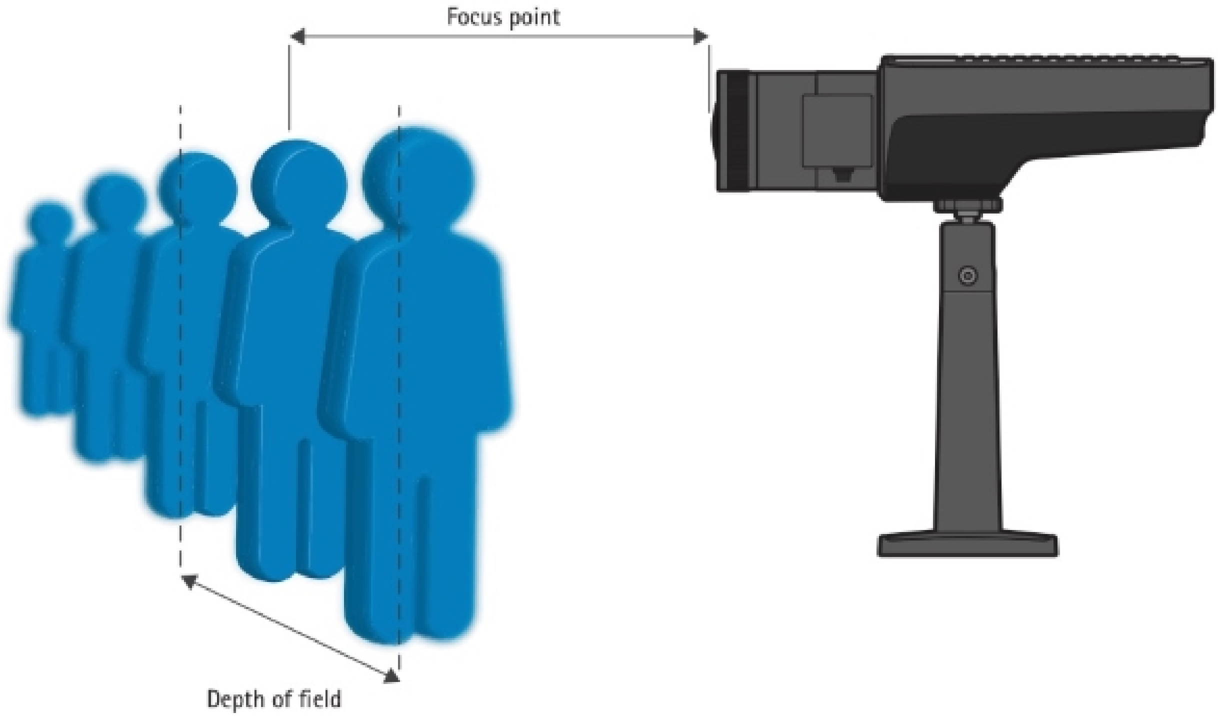
- Optical zoom – Zooms achieved by using the camera's lens. The lens is used to bend the light onto the full region of the sensor, usually resulting in negligible image loss. Optical zooms are very helpful when performing facial recognition.
- Digital zoom - Zooms achieved Scaling video in software is known as digital zoom. Digital zoom takes a smaller region from the already digitized image, cut the existing pixels into smaller ones and stretch those. This process often creates significant degradation of the image. It should be avoided always.

11.2.2 Sharpness

11.2.2.1 Focus Focus is critical for successful facial recognition. If a camera model provides a focus control, you should set the camera's focus to where you expect to capture subjects' facial images, as best you can.

Calibrating the camera using manual focusing and a good focus chart will almost always produce better results than using the camer's auto-focus, even if camera manufacturers claim otherwise. Auto-focus may just focus on a door or something at the extreme back end of where you want to focus.

11.2.2.2 Depth of Field When considering different cameras, a larger depth of field is desired because it means that the camera will be able to maintain a sharp and clear focus for a greater near and far distance.



Subjects are only in focus for a specific distance from the camera. Subjects both further or closer will be out of focus.

Auto-focus is typically not used with facial recognition because multiple people at different distances will sometimes need to be recognized, and auto-focus typically only focuses on individual objects rather than a group of objects. Auto-focus usually prioritizes focusing on closer objects, which will cause objects further back to lose focus. Furthermore, that closer object might not even be a person's face.

11.2.2.3 Video Compression Video compression is the process of encoding video files such that they consume less space and are easier to transmit over the network. Compression is often provided as a setting for the number of bits per second (aka the bitrate) delivered by a video stream. To receive the highest quality video you will need to perform analysis with each specific model of camera that you intend to use. As an initial guide, select a bitrate between 4K (4096) and 8K (8192) with VBR (variable bitrate, as opposed to CBR, constant bitrate), on the h.264 encoder is usually good to start.

11.2.3 Lighting Conditions

It is critical that subjects' faces are illuminated well enough that facial details are clearly visible by human eyes. The color of the light should be white; colored light can alter or "flatten" people's skin tones.

11.2.3.1 Backlight If the environment behind people is brighter than the light illuminating people's faces, the people will appear dark and with reduced details because the camera's sensor will be overwhelmed by the brightness behind the people. In such situations, a bright white light located near the camera illuminate people's faces. Such a light has the added benefit of causing most people to look directly at the camera as they seek the source of the bright light shining in their faces, which helps their Center Pose Quality. (See the Center Pose Quality section below for more information.)

11.2.3.2 Low Light A good low light camera will produce a video image that maintains image detail both within dark areas as well as within bright areas. A bad low light camera produces banding and noisy/grainy video when in low light. These functional differences are often the result of which sensor type the camera is using. Good low light cameras often use CCD sensors, while bad low light cameras often use less expensive CMOS sensors instead.

11.2.4 Center Pose Quality

A value from 0 to 1 that specifies how directly a face is looking at the camera. If a face is looking directly at the camera, this value is 1. The more that the face turns away from the camera, the lower this value becomes.

11.2.4.1 Yaw Angle (horizontal) Yaw is the horizontal angle between the direction a subject is looking and the camera line of sight. The ideal angle for facial recognition is 0Âř. (i.e. The subject is looking directly at the camera.) Facial recognition works well for angles up to 30Âř. Between 30Âř and 60Âř recognition still occurs but only if motion is relatively low or the lighting is good. At angles above 60Âř up to 90Âř facial recognition is very challenging but still possible.

11.2.4.2 Depression Angle (vertical) Depression angle is the vertical angle from the subject's face up (or down) to the camera. A value of 15Âř or less is best though up to 30Âř is acceptable. Values greater than 45Âř will present a challenge to the face recognition software.

11.2.5 Data Rate

11.2.5.1 Frame Rate Frame rate refers to the number of video frames delivered by the camera per second. In general, 15 frames per second is considered the minimum for real-time surveillance. When selecting which camera(s) to use for facial recognition, check to see if the frame rate changes significantly with resolution. If it does, that's an indication that after-capture software is scaling the video, which is bad for facial recognition.

11.2.5.2 Video Bitrate The video bitrate should be selected to ensure highest quality possible within the network limitations. The table below provides the recommended video bitrate for common resolutions.

Resolution	Bitrate (Kbps)	30 fps	20fps	15fps	10 fps
3000p	Max	27000	20500	16400	12300
	Avg	11000	8200	6600	5200
2160p	Max	20000	14300	11300	9200
	Avg	8000	6100	5100	4200
2048p	Max	14000	9200	7700	6100
	Avg	6000	4200	3700	2900
1920p	Max	11000	8200	6700	5100
	Avg	5000	3700	3200	2600
1440p	Max	8000	5100	4400	3600
	Avg	4000	2600	2200	2200
1080p	Max	5000	3100	2200	1500
	Avg	2500	1900	1600	1200

- When using constant bitrate (CBR), the Max value shown above is recommended.
- Select the best compression technology available for the camera (h.264, h.264+, or h.265). Some cameras offer custom technologies that reduce bandwidth usage even further. For example, ZipStream by Axis supports dynamic frame rate, dynamic GOP, and region of motion encoding which greatly reduce bandwidth usage while still maintaining compatibility with all standard decoders.

12 Set up ONVIF IP Cameras

A camera must be correctly configured for authentication via the ONVIF protocol to work.

12.1 Enable ONVIF

Make sure that ONVIF is enabled in the camera settings. The precise procedure for how this is done depends on the make and model of the camera.

12.2 Configure the Date and Time

The canmera's configured date & time must not differ by more than +/- 5 seconds from the machine you're connecting the camera to. Follow these steps to ensure that the camera date & time are configured correctly:

1. Set the camera Time Zone to the local time zone. (e.g. GMT-8 if you're in Seattle)
2. Disable daylight savings time (DST) adjustments. The Network Time Protocol (NTP) will take care of this automatically.
3. Set the NTP server to `time.google.com` and port `123`.
4. Synchronize the camera time to the time on your computer. The web interface usually has a button that allows you to do this.
5. Enable the NTP service.

The end result should be that the camera's date and time are up-to-date and that the NTP service is enabled to keep it up-to-date.

Some camera web UIs will show an incorrect/strange/nonsensical time after you've set the time zone to your local time zone. Do not change the time zone away from your local time zone! It must be set to the local time zone for ONVIF authentication to work.

12.3 Configure the Camera's ONVIF User

Many cameras maintain two sets of users: one set of web users and a second (and independent) set of ONVIF users. These cameras with 2 sets of users do not automatically create an ONVIF user even when a new web user is created.

Be sure that your camera has at least one ONVIF user with administration privileges. If there aren't any ONVIF with administration privileges, ONVIF authentication will not work.

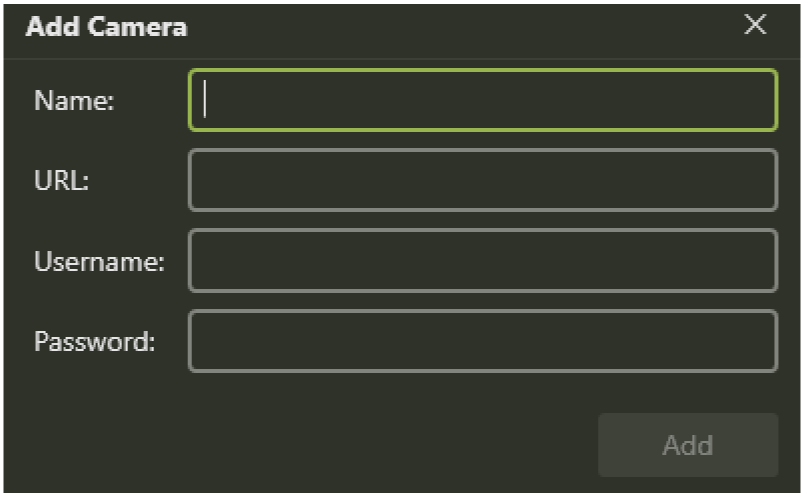
13 Connect Cameras to SAFR

SAFR supports USB and integrated cameras, which are always auto-detected. SAFR also supports the standard Open Network Video Interface Forum (ONVIF) camera auto-discovery protocol used by most IP cameras. When ONVIF discovery is enabled, IP cameras are also auto-detected. To connect an auto-detected camera, simply click on the **Select Camera** menu on *Camera* window.

In some cases, however, ONVIF discovery is disabled by default. (ONVIF is sometimes disabled by default for security reasons.) This makes the camera effectively invisible to SAFR and SAFR is unable to discover or communicate with the camera automatically. See the Manually Add and Configure IP Cameras section to see how to manually add such cameras.

13.1 Manually Add and Configure Cameras

1. On the Desktop Client, click **Tools > Preferences**, then select the **Camera** tab.
2. In the lower left corner, click +.
3. In the **Name** field, enter a descriptive name for the camera.
4. In the **URL** field, enter the RTSP (Real Time Streaming Protocol) URL to the live video feed of the camera or an RTSP server. For information on how to find the RTSP URL for your camera, see the **Determine the IP Address and Streaming URL for the Camera** section below.



After a camera is successfully connected, you can configure it. See camera preferences for more information on available options and how to use them.

13.2 Determine the IP Address and Streaming URL for the Camera

The IP addresses of many security cameras can't be auto-discovered because their ONVIF has been disabled for security reasons. This makes the camera effectively invisible to SAFR and thus SAFR is unable to find and communicate with the camera automatically. For SAFR to discover and connect to a camera automatically, you must enable ONVIF.

If you already know the IP address for the camera, do the following:

1. Connect to the camera from a web browser by typing the URL of the camera's IP address (e.g. *http://10.124.13.34*).
2. Find its streaming URL (starting with rtsp://). You need the streaming URL to enter when you add the camera to SAFR. The rtsp:// URL you enter into SAFR most likely includes the camera user ID and password.

The streaming URLs, while different for different camera manufacturers, tend to follow the same format for different camera models of the same manufacturer. The following table includes a few examples of camera streaming URLs for different camera manufacturers. To use them for your camera, match your camera make to an example listed in the table and enter the SAFR-provided streaming URL, replacing <username>, <password>, and <IP address> with actual values configured for your camera. Even if your camera model is not the same as the one listed, there is a good chance the streaming URL provided in the table works if your camera manufacturer is the same.

Camera Make	Camera Model	Example rtsp:// URL
Avigilon	5.0-H3-DO1-IR	rtsp://<username>:<password>@<IP address>/defaultPrimary?streamType=u
Axis	Q6128-E	rtsp://<username>:<password>@<IP address>/axis-media/media.amp
Dahua	HFW5421E-Z	rtsp://<username>:<password>@<IP address>/cam/realmonitor?channel=1&subtype=0
HikVision	DS-2CD4185F-IZ	rtsp://<username>:<password>@<IP address>/h264
Mobotix	M26	rtsp://<username>:<password>@<IP address>/mobotix.h264
Panasonic	WV-SFV781L	rtsp://<username>:<password>@<IP address>/MediaInput/h264/stream_1
Samsung	SND-L6013R	rtsp://<username>:<password>@<IP address>/onvif/profile2/media.smp
Sony	SNC-VM772R	rtsp://<username>:<password>@<IP address>/video1

If your camera make isn't listed above, you can do the following to find your camera's RTSP URL:

1. Execute the following Google search: `<Your Camera Make> RTSP URL iSpy` (e.g. `Ubiquiti RTSP URL iSpy`)
2. Go to the returned ispyconnect.com address for your camera.
3. Scroll down & click on your camera's model to generate the RTSP URL for your camera.

13.2.1 URLs With Unusual Characters

Sometimes when the rtsp:// URL contains unusual characters (e.g. ^), you'll get an error when you try to enter the URL to connect to the camera. To resolve the error, do the following:

1. Go to https://www.urlencoder.org/.
2. Copy the problematic URL into the grey **Encode to URL-encoded format** box at the top of the page.
3. Accept the default **Destination character set**. (i.e. UTF-8)
4. Press the **Encode** button.
5. Use the encoded URL that appears in the grey box below the **Encode** button to connect to the camera.

Here is a video that demonstrates how to do this: URL_Encoding.mp4

13.3 Save Your Camera Configuration

After you complete your manual camera connection, we recommend that you export and save the created camera connection configurations. Although your configurations are automatically stored in the Desktop Client, they cannot be shared with other SAFR components that are installed on different computers unless you export the camera configurations. Exporting your camera connection configurations makes your setup work shareable to other SAFR components and preserves it in case the Desktop Client is re-installed later on different hardware.

To export your camera configurations, do the following:

1. In the Desktop Client menu, click **Tools > Preferences** and select the **Camera** tab.
2. In the lower left corner, click the gear icon.
3. Click **Export Configurations**.
4. Specify the file name and location where the configurations are to be saved.

Note: Your camera connection configurations are saved in a file with an .acc extension. This file may contain your camera access credentials, so save it in a secure location.

To import camera configurations, do the following:

1. In the Desktop Client menu, click **Tools > Preferences** and select the **Camera** tab.
2. In the lower left corner, click the gear icon.
3. Click **Import Configurations**.
4. Specify the file name and location where the configurations are located.

13.4 Interaction of Auto-detected and Manually Configured Cameras

Auto-detected camera configurations and manually entered camera configurations may exist for the same camera. This situation does not cause a conflict. Each configuration can be separately selected; you can choose a method of connection when selecting a camera from the feed window.

Your exported camera connection configurations include only manually configured cameras. Auto-detected camera configurations are dynamic and are discovered by the SAFR client at start and when *Camera Preferences* are opened. Because auto-discovered cameras are dynamically discovered, they may not appear for a few seconds after the application starts, so make sure to wait several seconds to give the discovery process time to complete.

13.5 Delete a Camera Configuration

To delete a camera configuration, do the following:

1. In the Desktop Client menu, click **Tools > Preferences** and select the **Camera** tab.
2. In the left hand field select the camera you want to delete.
3. In the lower left corner, click **-**.

Note: Auto-detected camera configurations cannot be deleted. They are dynamically discovered.

14 Connect to a Video Feed

The Desktop Client requires a video feed window to be open for each camera feed (or video file) it is monitoring. The video feed window not only facilitates SAFR monitoring of the video but can also present additional information overlaid on top of the video to assist staff in interpreting the scene in the video. For more information on the additional overlaid information, see Interpret Video Feed Overlays.

The Desktop Client automatically detects integrated, USB, and Open Network Video Interface Forum (ONVIF)-compatible IP cameras. While you're becoming familiar with SAFR, we recommend that you plug in a single USB camera (aka a web cam). Only after you've spent some time learning SAFR should you attempt to connect to additional IP cameras.

The instructions on this page assume you have at least one camera detectable from the computer where SAFR is installed. See Connect Cameras to SAFR for information about how to connect a camera.

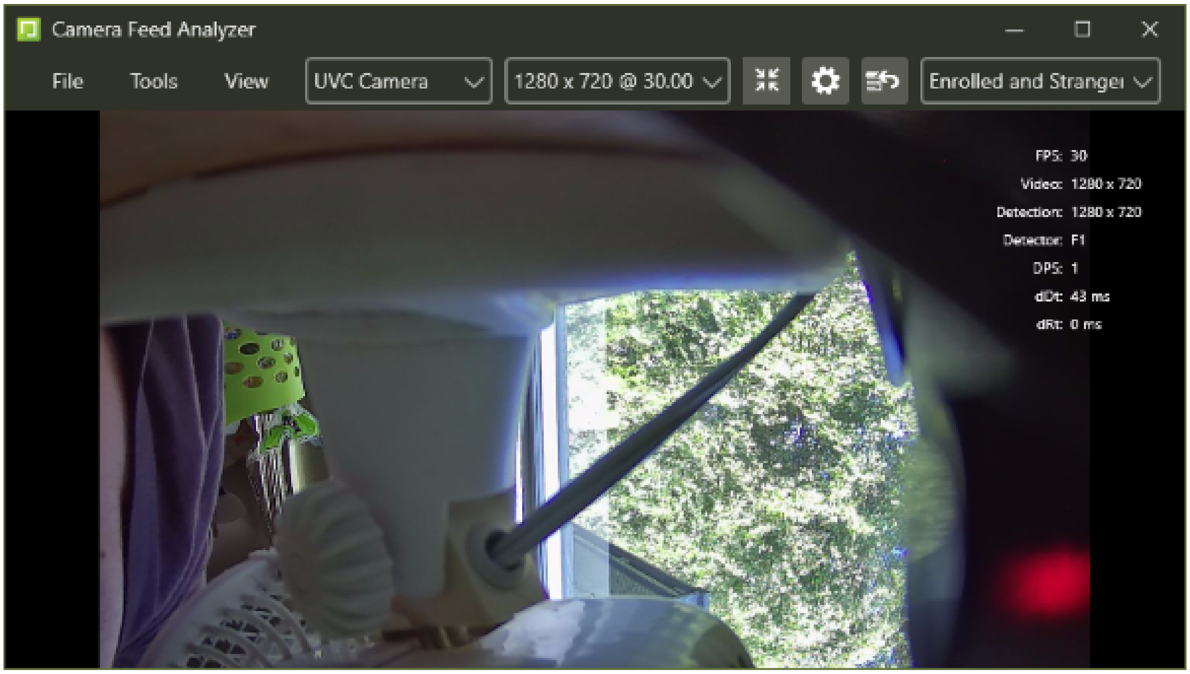
14.1 Connect to a Camera Video Feed with the Desktop Client

If you have an on-premise deployment, you must first install and configure SAFR Platform before connecting to a live video feed. For information on how to do this, see Getting Started with SAFR Platform on Windows or macOS or Getting Started with SAFR Platform on Linux.

If you have a cloud deployment, your Desktop Client should automatically be configured to connect to the SAFR Server in the cloud.

To connect to a video feed, do the following:

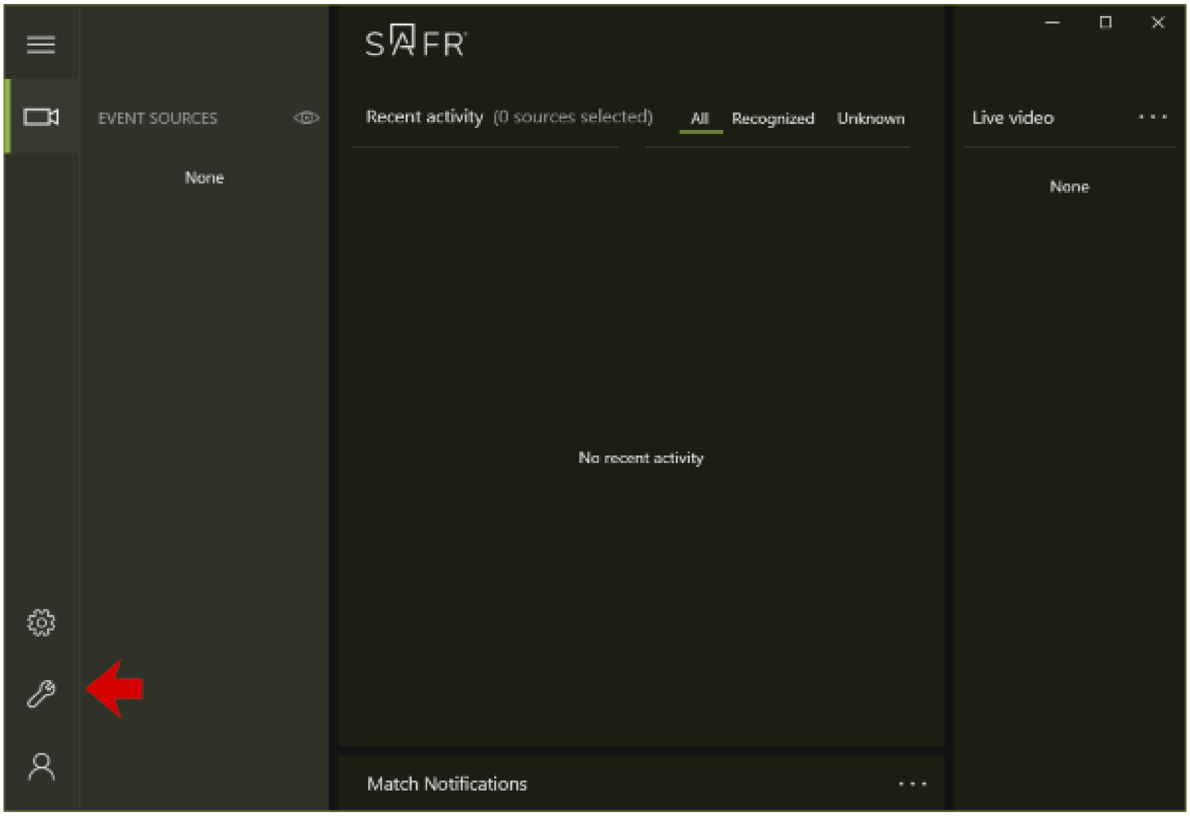
1. Start your Desktop Client. It should default to the *Camera Feed Analyzer* window, shown below.



FPS: 30
Video: 1280 x 720
Detection: 1280 x 720
Detector: F1
DPS: 1
dDt: 43 ms
dRt: 0 ms

If it doesn't default to that window, you can open a new *Camera Feed Analyzer* window by clicking on **Tools > Camera Feed Analyzer. . .** .

Note: In the Operator Console window, the *Tools* menu is the wrench, as shown below.



2. Along the top of the *Camera Feed Analyzer* window, select a camera from the **Camera Selector** drop-down menu. The menu displays all detected cameras.
3. Set the camera frame rate and resolution in the **Video Feed Profile Selector** to the right of the **Camera Selector** menu. Frame rate and resolution selection are only configurable for USB and IP cameras auto-discovered via ONVIF. For IP ONVIF cameras, only configured ONVIF media profiles are able to be selected. The higher the frame rate (i.e. frames per second) and resolution for a video feed, the more processing power is required to monitor and collect data from the video feed.

Once you complete this procedure, SAFR receives, monitors, and processes the video feed from the camera. The Desktop Client *Camera Feed Analyzer* window must remain open for SAFR to continue to monitor the video feed. If the window is closed, SAFR no longer receives the video feed and no longer monitors it. The *Camera Feed Analyzer* window can be minimized without affecting the monitoring of the feed.

14.2 Select a Video Processing Mode

A variety of different video processing modes are supported to accommodate different monitoring and security needs, as described in the table below. Each mode can be customized through the Detection, Tracking, Recognition, Events, and User Interface tabs of the **Preferences Window**.

SAFR Video Processing Mode	Description
Recognition	• This is the default mode typically used for set-up, validation, and experimentation. • Only reports enrolled individuals. • No events are generated or recorded. • Available for both Windows and macOS.

SAFR Video Processing Mode	Description
Import	• Any face that can be clearly seen but is unidentified will be automatically registered. • Faces that are already registered are only recognized and do not create additional entries in the Person Directory. • Additional different face images (e.g. from different expressions) may be added to the existing faces in the Person Directory if they improve the recognition of the person. • No events are recorded. • Available for both Windows and macOS.
Learn and Monitor	• Monitors all person events within view. If a person is not registered in the system, they are added as long as the image meets the specified image quality metric criteria. • Allows for automatic saving of recognized persons to the server. • Available for Windows.
Similar	• Registered faces that are the most similar to the face observed in the video are shown in real time in the *Camera Feed Analyzer* window. • Each similar face displays a % match. Anything less than 100% match is not a certain match. • No event information or images are recorded. • Available for macOS.
Secure Access	• Secure access uses strict criteria for confirming the identity of the face in the view of the camera. • Records events and images for recognized faces. • Listens for event replies and displays them on the screen. • Typically used for door access control. • Events and images are not recorded for unrecognized faces. • Available for both Windows and macOS.

SAFR Video Processing Mode	Description
Secure Access with RGB Liveness	• Secure access uses strict criteria for confirming the identity of the face in the view of the camera. • Records events and images for recognized individuals. • Listens for event replies and displays them on the screen. • Typically used for door access control where you want to use RGB liveness detection to guard against face spoofing. • See RGB Liveness Detection for information about RGB liveness detection. • Only available on Windows machines containing an NVidia card that provides GPU.
Secure Access with Smile	• Uses strict criteria for confirming the identity of the face in the view of the camera. It also looks for transitions in the facial expression (e.g. non-smiling to smiling). • Records events and images for recognized individuals. • Listens for event replies and displays them on the screen. • Typically used for door access control where it is necessary to guard against identity impersonation via photo. • Includes all functionality of *Secure Access* mode and can be used to allow access on first sight recognition for a certain time (e.g. when security staff are on duty) and can change to a higher degree of security (e.g. recognition with smile expression change) at set times (e.g. when security staff is off duty). • Events and images are not recorded for unrecognized faces. • Available for both Windows and macOS.

SAFR Video Processing Mode	Description
Secure Access with Smile and RGB Liveness	• Uses strict criteria for confirming the identity of the face in the view of the camera. It also looks for transitions in the facial expression (e.g. non-smiling to smiling). • Records events and images for recognized individuals. • Listens for event replies and displays them on the screen. • Typically used for door access control where it is necessary to guard against identity impersonation via photo and where you want to use RGB liveness detection to guard against face spoofing. • See RGB Liveness Detection for information about RGB liveness detection. • Only available on Windows machines containing an NVidia card that provides GPU.
Secure Access with Profile Pose	• Secure access uses strict criteria for confirming the identity of the face in the view of the camera. • Records events and images for recognized individuals. • Listens for event replies and displays them on the screen. • Typically used for door access control where you want to use pose liveness detection to guard against face spoofing. • See Pose Liveness Detection for information about pose liveness detection. • Available for both Windows and macOS.
Registration Kiosk	• Allows self-registration into the SAFR system. • Optimized to work from a tablet. We recommend the iPad Pro. • Can be customized in the User Interface preferences tab to display different prompts, gather additional information, or restrict registration with a minimum age. • Records registration and appearance events and associated images of registered persons. • Events and images are not recorded for unrecognized faces unless they initiate the registration process. • Available for macOS.

SAFR Video Processing Mode	Description
Enrolled Monitoring	• Facial recognition, events, and images are only recorded for registered/recognized faces. • Images and events are not recorded for unrecognized faces. • Available for both Windows and macOS.
Anonymous Traffic Monitoring	• When gender and/or age detection is enabled, this metadata is recorded anonymously for faces viewed by cameras. • Images and biometric information are not recorded. • Available for both Windows and macOS.
Enrolled and Anonymous Traffic Monitoring	• Recognition events are recorded for registered individuals. • When gender and/or age detection is enabled, this metadata is recorded anonymously for faces viewed by cameras. • No images are recorded for recognized or unrecognized faces. • Available for both Windows and macOS.
Enrolled and Unique Traffic Monitoring	• Events are recorded for registered and unknown individuals. • Any faces that can be clearly seen but are currently unknown are automatically registered. • Faces that can't be seen clearly enough to attempt recognition are reported as unrecognizable. • Age information is automatically recorded for all clearly seen faces. • Images are recorded for all faces. • Available for both Windows and macOS.
Enrolled and Stranger Monitoring	• Events are recorded for registered and unknown individuals. • Faces that are clearly seen but aren't currently registered are reported as strangers. • Faces that can't be seen clearly enough to attempt recognition are reported as unrecognizable. • Images are recorded for all faces. • Available for both Windows and macOS.

14.3 Recommendations for the Best Video Experience

- Use the highest resolution available (4K) if you need to monitor an area of 5 meters or wider.
- To monitor a narrow area of approximately 2-3 meters, 1080p video is sufficient.
- For up close door access applications, 720p video offers adequate quality.
- For resolutions of 1080p or higher, we recommend 15 frames per second.

- For cameras used in *Secure Access With Smile* mode, we recommend 30 frames per second at 720p resolution.
- Generally, one computer can support one 4K camera (or 2 HD cameras) for every 2 CPU cores depending on the camera make and model.

15 Interpret Video Feed Overlays

Video feed overlays are available in the *Camera Feed Analyzer* window. Depending on the View menu options you choose, various supplemental information is displayed in real time in the form of overlays and readouts to help monitor the video feed.

15.1 Color Codes for the SAFR Recognition Frame

The following colors are used to indicate the level of recognition for a face:

- **Gray**: Unrecognizable. Either the face does not meet minimum quality values to be recognized or the response from the attempted recognition has not yet arrived.
- **Purple**: Stranger. The face met sufficient quality for recognition but was not recognized and did not meet minimum quality to be registered.
- **Cyan**: Identified as a close match to recognized user but not 100% identification.
- **Blue**: Registered person without a name. The face was recognized as matching one already in the Person Directory.
- **Green**: Registered person with a name. The face was recognized as matching one already in the Person Directory.
- **Yellow**: Concern. The registered face has been tagged as a concern.
- **Red**: Threat. The registered face has been tagged as a threat.

15.2 Recognition Indicator Icons

The indicator icons indicate the following:

- Gender
- Age
- Sentiment
- Sentiment score
- Smile

15.3 General Video Information

The following information is provided in the Desktop Client video display:

- **FPS**: The number of frames per second being captured by the camera.
- **Video**: Video resolution. (e.g. 1280x720)
- **Detection**: Face detection resolution.
- **Detector**: The detector CPU capacity level. It's based on the number of CPU processing cores.

15.4 Video Processing Latency Information

- **DPS**: Frame detections per second.
- **dDt**: Detection time. For example, how long it is taking to detect a face.
  - This time should normally be in 20-50ms range.
- **dRt**: Recognition time. For example, how long it is taking to recognize a face.
  - This time should normally be in the 60 - 250 ms range for only identity recognition.
  - If age, gender, and sentiment are also being recognized, the time can be expected to be up to 450 ms.

If your processing times are longer than indicated here, it may be an indication your system is overloaded. You should look to offload some of your video feed processing to other computers.

15.5 Face Detection Information

There are 3 main image quality metrics:

- **Q**: Center pose. Represents how directly the face is looking at the camera. A face looking directly at the camera would receive a score of 1. The more the face looks up, down, left, or right of the camera, the more this metric is reduced.
- **S**: Face sharpness. Represents how clear the image is. A score of 1 represents a perfectly clear image. while 0 represents an extremely blurry image.
- **C**: Face contrast. Represents the color contrast within the image. A score of 1 represents an image with very high contrast, while 0 represents very low contrast.

For guidance on these metrics, see Image Quality Metrics Guidance. These metrics can be configured in the recognition preferences menu.

- **Face size**: Face size is the resolution of the image with a 25% margin. It can be used to ignore background (smaller) faces or to require an up-close high resolution image to be presented before the face is learned by the SAFR system. See detection preferences and recognition preferences for information on how to customize SAFR behavior based on the detected face size.
- **Gender**: Displayed as an icon if gender recognition is enabled.
- **Age**: Age of user if age recognition is enabled.
- **Face recognition image submission**: The thumbnail in the lower right corner is the image submitted for facial recognition. This is only displayed if you select *Recognition Candidates* (for macOS) or *Detection List and Recognition Details* (for Windows) from the View menu.

16 View Video Feeds Status

Video feeds status provides real-time monitoring of the Desktop Client, the Mobile Client, and the VIRGO client within an account. While VIRGO clients support both remote monitoring and configuration, the Desktop and Mobile Clients only support remote monitoring and must be configured locally in their GUI.

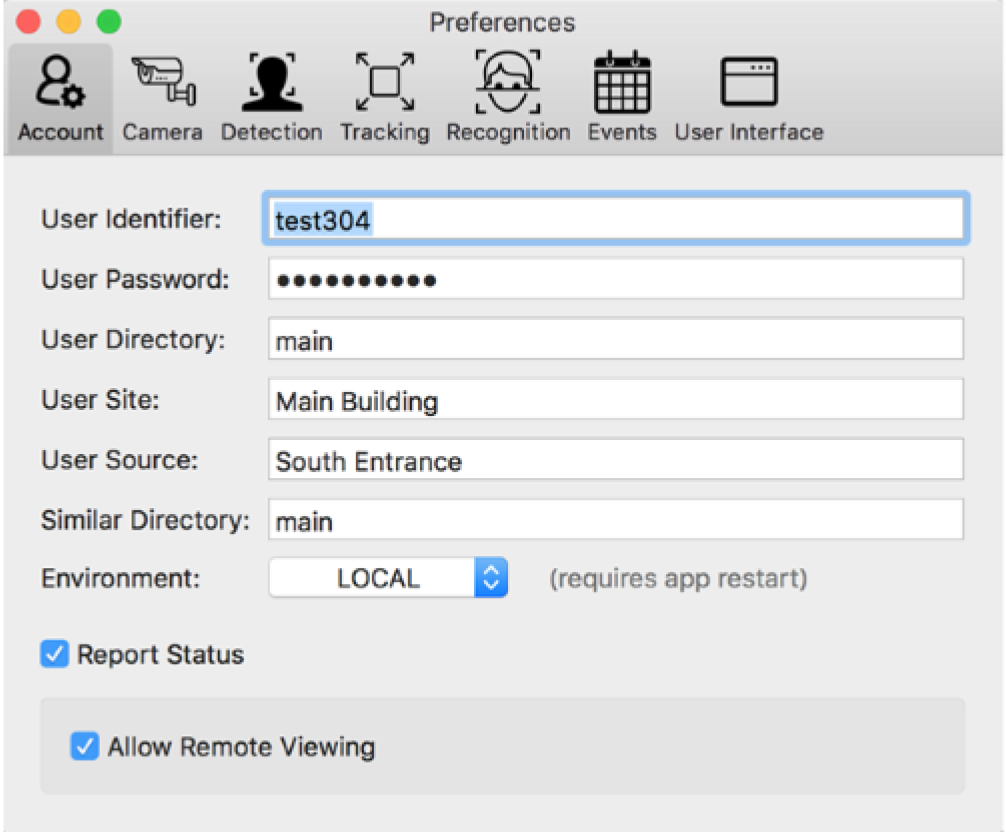
Note: This feature is available with the Web Console or on macOS Desktop Client installations.

16.1 Enable the Desktop or Mobile Clients for Remote Monitoring

Both the Desktop and Mobile Clients can be enabled for remote status monitoring and remote viewing.

To enable those clients for remote monitoring:

- For macOS, click the **SAFR** > **Preferences** > **Account** tab, and select the **Report Status** and **Allow Remote Viewing** check boxes.



User Identifier: test304

User Password: ●●●●●●●●●●●

User Directory: main

User Site: Main Building

User Source: South Entrance

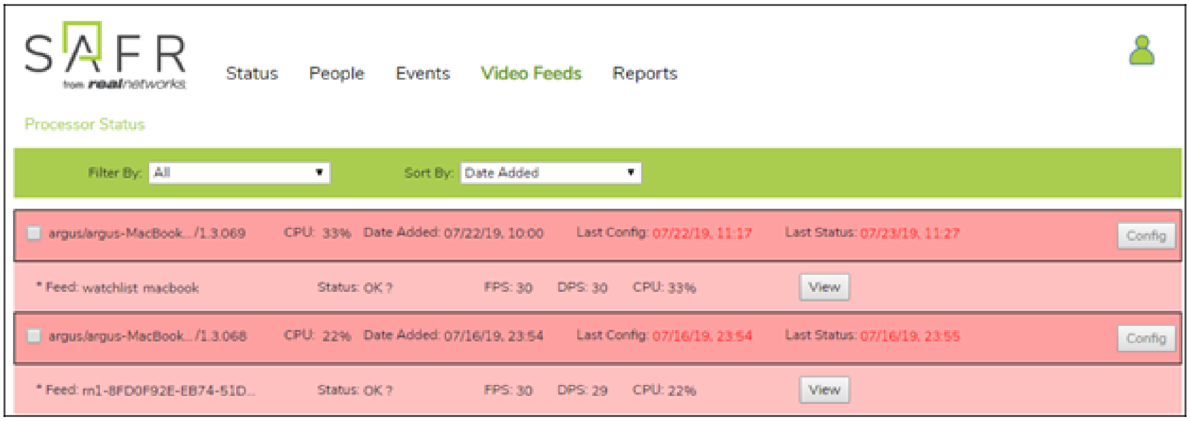
Similar Directory: main

Environment: LOCAL (requires app restart)

☑ Report Status

☑ Allow Remote Viewing

- In the Web Console, click **Video Feeds** > **Processor Status**.



Filter By: All ▾ Sort By: Date Added ▾

□ argus/argus-MacBook.../1.3.069 CPU: 33% Date Added: 07/22/19, 10:00 Last Config: 07/22/19, 11:17 Last Status: 07/23/19, 11:27 Config

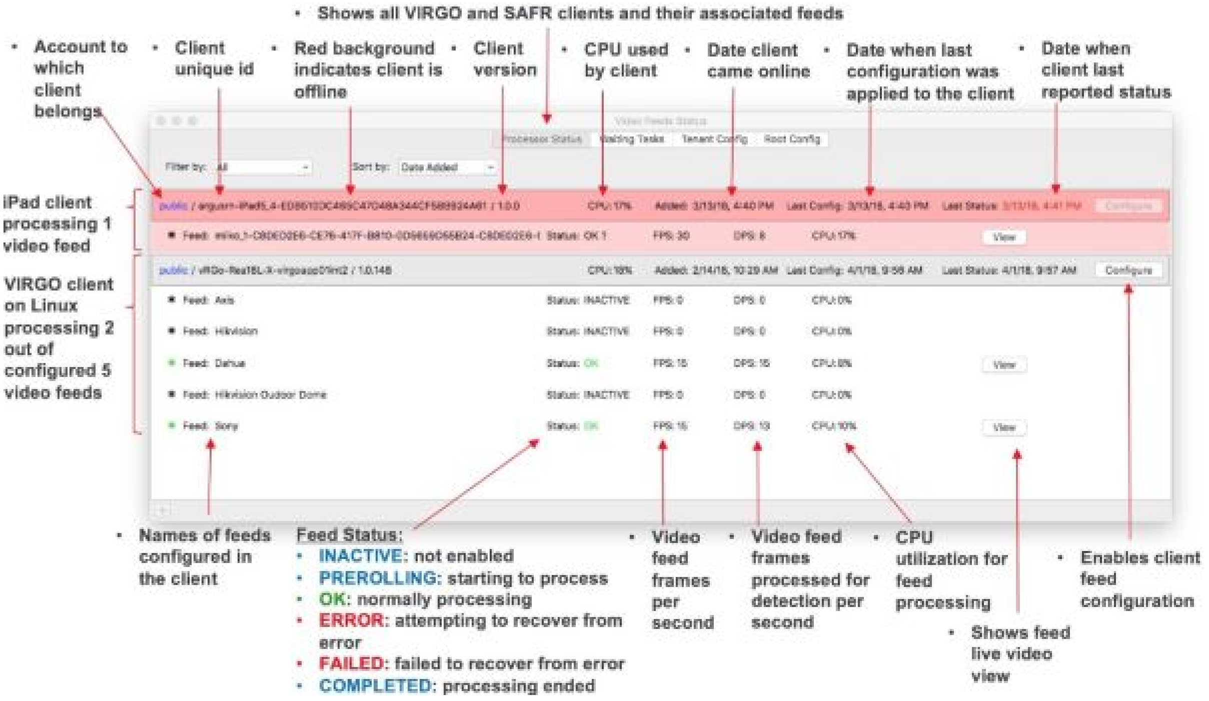
* Feed: watchlist macbook Status: OK ? FPS: 30 DPS: 30 CPU: 33% View

□ argus/argus-MacBook.../1.3.068 CPU: 22% Date Added: 07/16/19, 23:54 Last Config: 07/16/19, 23:54 Last Status: 07/16/19, 23:55 Config

* Feed: m1-8FD0F92E-EB74-51D... Status: OK ? FPS: 30 DPS: 29 CPU: 22% View

16.2 The Video Feeds Status Window (aka Processor Status Window)

The Video Feeds Status (or Processor Status) window can be accessed from the **SAFR** menu of the Desktop Client for macOS and from the Web Console Video Feeds page. It displays real-time status of all VIRGO, Desktop, and Mobile Clients enabled for status reporting.



Filter by: All Sort by: Date Added

public / arguarn-iPad5,4-ED8610DC469C47048A344CF589924A61 / 1.0.0 CPU: 17% Added: 3/13/18, 4:40 PM Last Config: 3/13/18, 4:40 PM Last Status: 3/13/18, 4:41 PM Configure

■ Feed: mlko_1-C8DE02E6-CE76-417F-8810-0D9659065B24-C8DE02E6-1 Status: OK 1 FPS: 30 DPS: 8 CPU: 17% View

public / vIRGo-Rea18L-X-virgoapp01ml2 / 1.0.148 CPU: 18% Added: 2/14/18, 10:29 AM Last Config: 4/1/18, 9:56 AM Last Status: 4/1/18, 9:57 AM Configure

■ Feed: Axis Status: INACTIVE FPS: 0 DPS: 0 CPU: 0%

■ Feed: Hikvision Status: INACTIVE FPS: 0 DPS: 0 CPU: 0%

■ Feed: Dahua Status: OK FPS: 15 DPS: 15 CPU: 8% View

■ Feed: Hikvision Outdoor Dome Status: INACTIVE FPS: 0 DPS: 0 CPU: 0%

■ Feed: Sony Status: OK FPS: 15 DPS: 13 CPU: 10% View

- Shows all VIRGO and SAFR clients and their associated feeds
- Account to which client belongs
- Client unique id
- Red background indicates client is offline
- Client version
- CPU used by client
- Date client came online
- Date when last configuration was applied to the client
- Date when client last reported status

iPad client processing 1 video feed

VIRGO client on Linux processing 2 out of configured 5 video feeds

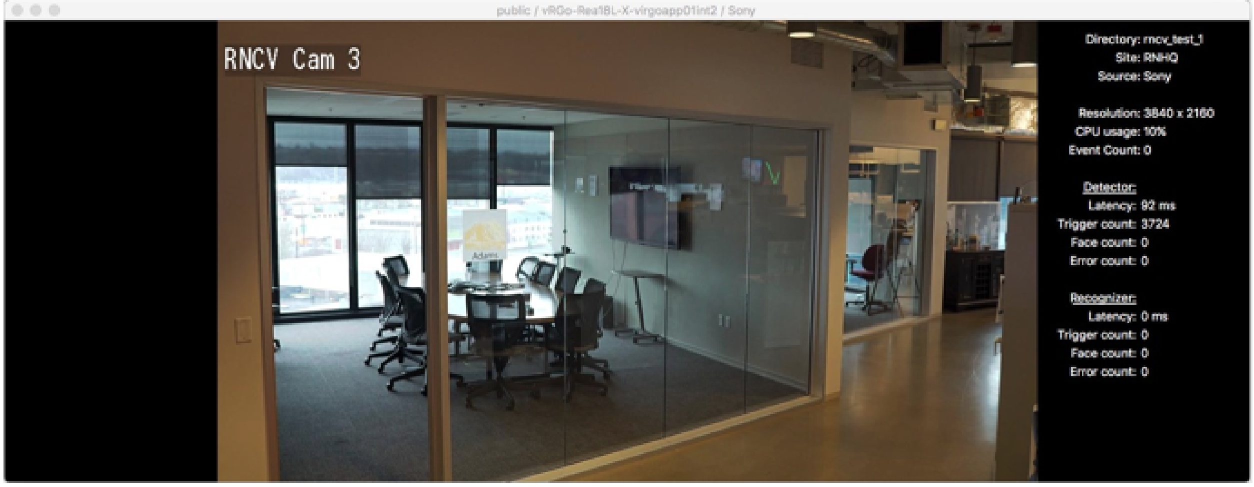
- Names of feeds configured in the client

Feed Status:

- INACTIVE: not enabled
- PREROLLING: starting to process
- OK: normally processing
- ERROR: attempting to recover from error
- FAILED: failed to recover from error
- COMPLETED: processing ended

- Video feed frames per second
- Video feed frames processed for detection per second
- CPU utilization for feed processing
- Enables client feed configuration
- Shows feed live video view

16.3 The Live Status Video View



```
Directory: rncv_test_1
Site: RNHQ
Source: Sony

Resolution: 3840 x 2160
CPU usage: 10%
Event Count: 0

Detector:
Latency: 92 ms
Trigger count: 3724
Face count: 0
Error count: 0

Recognizer:
Latency: 0 ms
Trigger count: 0
Face count: 0
Error count: 0
```

The **Live Status Video** view can be accessed by clicking the **View** button next to an active feed in the **Video Feeds Status** window. You can display the **Live Status Video** view for all active VIRGO clients and for all Desktop and Mobile Clients for which you have enabled **Allow Remote Viewing**.

You can simultaneously view live status video of as many feeds as you like from a single remote viewer. Only one viewer of the status video is allowed at any given time, however.

The **Live Status Video** view is displayed at low frame rates (~1 frame per second) and is intended for cursory inspection or monitoring. Full fidelity live video view, including recognition overlays, is available only in the video feeds window of the Desktop and Mobile Clients to which the camera is connected.

The top of the status video view displays the *Account*, *Client ID*, and the *Name* of the video feed.

On the right-hand side of the view, overlaid text displays additional information about the feed.

- **Directory**: Face Directory used for face recognition in the video feed, as specified in Account Preferences.
- **Source**: Source label for the video feed, as specified in Camera Preferences.
- **Site**: Site label for the video feed.
- **Source**:Source label for the video feed.
- **Resolution**: Video feed resolution.
- **CPU Usage**: CPU percent used for video feed processing.
- **Event Count**: Number of events reported since start of video feed processing.

Detector:

- **Latency**: Face detector latency in the last second of operation.
- **Trigger count**: Number of times face detector was triggered.
- **Face count**: Number of faces detected since the start of the video feed.
- **Error count**: Number of face detector errors since video processing started.

Recognizer:

- **Latency**: Face recognition latency in the last second of operation.
- **Trigger count**: Number of times face recognition was triggered since the start of the video feed.
- **Face count**: Number of faces recognized since video processing started.
- **Error count**: Number of face recognition errors since video processing started.

17 Manage People in the Person Directory

The Person Directory contains a list of all people stored in the user directory location specified under Account Preferences. To open the directory from the Desktop Client:

- For macOS, the **People** menu item is available under the **SAFR** menu.
- For Windows, click **Tools > People**.

By default, the list is displayed in chronological order with the most recently added displayed first. You can also search and filter identities by *Name*, *Person Type*, *ID Class*, and *Home Location.* All 4 of those properties can be changed by clicking the available fields to the right of the identity's picture.

- Metadata applied to identity groups is applied to all identities within the group. Changing these properties for any identity within a group will cause the change to be applied to all identities within that group.
- Groups are alternative identities belonging to a single person. While rare, a person may require such grouping to fully cover all different face modalities by which he or she can be recognized.

Double click the identity entry to view or edit even more information associated with the identity.

- The *Id Class* field is important and can be used to define a person as a *Concern* or *Threat.*
- *Moniker* is an advanced feature used to realize two factor authentication with visual badges.

You can also perform the following actions on identities in the People Directory:

- **Regroup**: Removes selected face from their existing groups (if any) and forms a new group of faces to represent a new identity. Root identity is always the earliest one added to the directory.
- **Delete**: Deletes selected identities and all information associated with the identity from the directory. All information associated with the identity is removed.
- **Export**: Exports a face image into an image (.jpg) file on the local drive.
- **Refresh**: Reloads the people directory page making sure up-to-date information is displayed.

17.1 Add a Person Type or Home Location

In the Person Directory, click **Add Person Type**, and then type the *Person Type* you want to assign (for example, Staff, Guest, or Maintenance). Likewise, you can click **Add Home Location** and type text representing a person's home location.

Best Practice: You can create and customize as many *Person Types* and *Home Locations* as you like, but we recommend keeping the list short (less than a dozen or so) because short lists are easier to maintain. As *Person Types* are entered for a few registered individuals, *Person Types* that are already entered become available for selection once **Add Person Type** is clicked, which makes designation easier for new registration. The same is the case for *Home Location.* The system knows of all previously entered *Home Locations* and offers them in the menu when **Add Home Location** is clicked.

18 Importing and Registering People

There are three main ways to register people to SAFR's Person Directory: cameras, photos, and recorded video. Imported people are registered to the Person Directory and stored in the directory specified in the User Directory setting of your Account Preferences.

18.1 Register People Using a Camera Connected to the Desktop Client

1. Select the connected camera you want to use by clicking on **File > New** to open the Desktop Client's *Camera* window, then select a camera from the **Select Camera** menu.
2. Set your **video feed processing mode** (located in the upper right hand corner) to one of the following modes: *Recognition*, *Import*, or *Learn and Monitor*. *Recognition* is considered the default mode for set-up validation and experimentation.
3. All the faces in view of the camera will initially have a grey overlay, which indicates one of two things:
    1. Your client or console hasn't received a response from its attempted recognition from the SAFR Server yet, or
    2. The face does not meet minimum image quality metric values and recognition cannot be successful. If the grey overlay persists, then the problem is the image quality. Try cleaning your camera lens or adjusting your camera placement to fix the problem.
4. When the overlay turns purple, the face has sufficient image quality for recognition by SAFR but it isn't recognized by SAFR because it hasn't been registered yet. To register the face, double click the face and the *Register* dialog will open.
5. You can choose to enter a name for the face if you want, but it's not required. Click **Register** to complete the registration.
6. The color of the face's overlay will change to either green (if you named the face) or blue (if you didn't name the face). Both of those colors mean that SAFR recognizes the face. For more information on overlays, see Interpret Video Feed Overlays.

18.2 Register People Using the Mobile Client

Another way to register faces is by using a Mobile Client installed on an iOS or Android device. For more information, see the Connect a Registration Kiosk topic in the Mobile Client documentation.

18.3 Register People by Importing Faces from Picture Files

To import faces from picture files, do the following:

1. Open either the Desktop Client (by clicking on the **SAFR** icon on your desktop) or the Web Console. (See Access the Web Console for information on how to do this.)
2. On the Desktop Client, open the People Window by clicking on **Tools** > **People**. On the Web Console, click on the **People** tab.
3. On the Desktop Client, click the **Add face** button at the top of the People Window, then select a facial image. On the Web Console click on the up arrow symbol in the upper right hand corner and select a facial image. **Note**: On the Windows Desktop Client, you're able to select multiple facial images at once, thus allowing you to register multiple people at once.
4. Image files are usually .jpg, .jpeg, or .png files. If the file you selected has multiple faces on it, SAFR will attempt to import all the faces in the image.
5. On Windows, you're able to select multiple image files at the same time to import all of them at one time. We recommend that you select no more than 20 images at a time. If you select more than ~20 images at a time, you're likely to experience a degradation of system performance.
6. When you import facial images, you may be prompted to resolve any duplicate and/or low-quality image conflicts that may have arisen.

18.3.1 Resolve Duplicate Images

Windows only.

To resolve conflicts resulting from people that already exist in the Person Directory, do the following:

1. Click on the **Fix conflicts** link in the notification bar.
2. Duplicates are displayed side by side in the *Report Dialog*.
3. You will be given 4 options:
   - **Create New Person with Imported Face**: A new identity will be created, and the imported face will be used for the identity's reference image.
   - **Replace Existing Face with Imported Face**: The imported face will replace the already-existing face as the reference image for that identity.
   - **Add to Existing Person**: The imported face will be added as an alternate face image for the identity.
   - **Skip Import**: The new face isn't imported; it will be discarded.
4. Repeat the previous step for all duplicates.

Note: At the top of the *Report Dialog*, there is an option to accept the default recommendations. SAFR defaults to accept whichever of the duplicate images is higher quality based on internal image quality metrics.

18.3.2 Resolve Low Quality Images

Windows only.

To resolve conflicts resulting from low-image quality images, do the following:

1. Click on the **Fix conflicts** link in the notification bar.
2. Low-quality images are displayed in the *Report Dialog*. Note that a warning symbol appears next to the image quality metric(s) that are problematic. See Image Quality Metrics Guidance for information and guidance on the metrics.
3. Click on **Import anyway** to import the low-quality image, or click **Skip Import** to not import the image.
4. Repeat the previous step for all low-quality images.

Note: Clicking on **X** in the Notification bar or at the top of the *Report Dialog* cancels the import operation for all remaining low quality image conflicts.

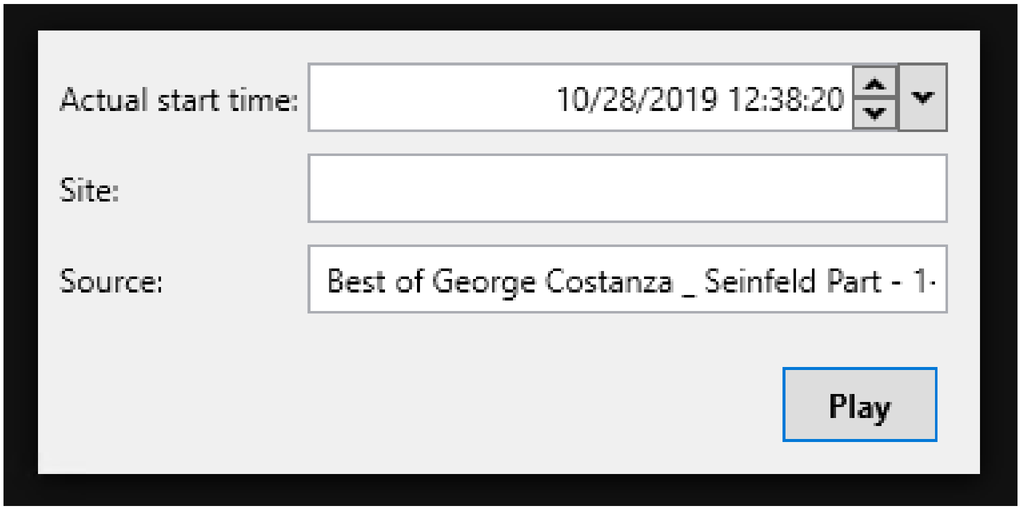
18.4 Register People from a Video File

You can open a saved video file to recognize and extract facial recognition data. To do so, do the following:

1. Open the Desktop Client.

2. Click **File > Open**, and then browse to any saved *.mp4, *.webm, or *.mkv file to open it.

3. If you're on a Windows machine and you have event reporting enabled for the currently selected video processing mode, (located on the Events Preferences tab) the dialog below will open. (If you don't meet both of these conditions, then the video will simply open.)


Actual start time: 10/28/2019 12:38:20

Site:

Source: Best of George Costanza _ Seinfeld Part - 1·

- **Actual start time**: The timestamp that the video will acquire when you press **Play**. (e.g. In the example above, the played video's timestamp would start at 12:38,10/28/2019) The input box starts 'live' and keeps up-to-date with the local time. When you interact with the time or set the focus, the input box stops being live.
  Note: Deleting the timestamp and leaving the field blank is valid, despite the red outline that the field acquires. Of course, if you do leave the field blank, the video won't have a timestamp, as expected.
- **Site**: The **Site** label that will be applied to all events generated by the video. This field is auto-populated with your **User Site** preference located in the Account Preferences.
- **Source**: The **Source** label that will be applied to all events generated by the video. This field is auto-populated with the name of the video.

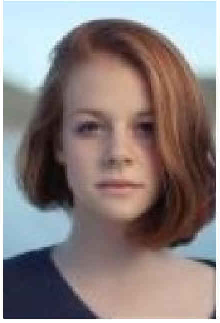
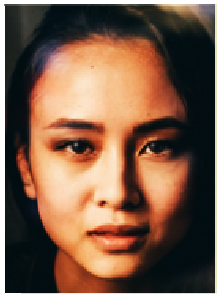
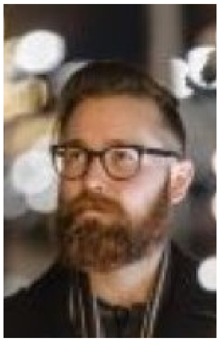
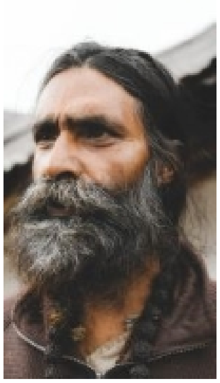
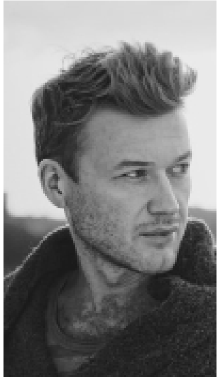
4. Set the video file's video processing mode to *Recognition*.

5. SAFR will proceed to register any unregistered faces that appear in the video.

19 Image Quality Metrics Guidance

Choosing to import images that have been flagged as “low-quality” will cause more false positives to occur as SAFR incorrectly identifies newly scanned faces as identical to the low-quality facial image. Greater discrepancies between the recommended metric value and the actual metric value will result in more false positives. Similarly, having more than one metric value be poor or very poor will also result in more false positives.

19.1 Center Pose

				
Center Pose = .89	Center Pose = .76	Center Pose = .54	Center Pose = .34	Center Pose = .21

Center pose represents how directly the face is looking at the camera. The more the face looks up, down, left, or right of the camera, the more this metric value is reduced from 1. Similarly, if the face is tilted in any way (e.g. the person’s chin is pointing at a corner of the image) this metric value is reduced. The default recommended minimum value for this metric is .59. You can adjust the recommended minimum value by going to **Tools -> Preferences**, clicking on the **Recognition** tab, then adjusting the **For merging** slider in the **Minimum required center pose quality** section.

Quality Label	Metric Range	Description
Excellent	0.7 - 1.0	Full recognition accuracy can be expected under all conditions.
Good	0.6 - 0.7	Very good recognition accuracy can be expected in general but may confuse closely related family members.
Marginal	0.45 - 0.6	Good recognition but may result in occasional failures.
Poor	0.3 - 0.45	Recognitions can be performed to significant extent but may produce false recognitions.
Very Poor	0.0 - 0.3	Recognitions can still be performed but with significant possibility of confusing similar faces.

19.2 Sharpness

Sharpness = .79	Sharpness = .62	Sharpness = .58	Sharpness = .35	Sharpness = .22

Sharpness represents how clear the facial image is. The more blurry the face is, the more this metric value is reduced from 1. The default recommended minimum value for this metric is .45. You can adjust the recommended minimum value by going to **Tools -> Preferences**, clicking on the **Recognition** tab, then adjusting the **For merging** slider in the **Minimum required face sharpness** quality section.

Quality Label	Metric Range	Description
Excellent	0.7 - 1.0	Full recognition accuracy can be expected under all conditions.
Good	0.6 - 0.7	Very good recognition accuracy can be expected in general but may confuse closely related family members.
Marginal	0.45 - 0.6	Good recognition but may result in occasional failures.
Poor	0.3 - 0.45	Recognitions can be performed to significant extent but may produce false recognitions.
Very Poor	0.0 - 0.3	Recognitions can still be performed but with significant possibility of confusing similar faces.

19.3 Contrast

Contrast = 1	Contrast = .87	Contrast = .63	Contrast = .47	Contrast = .40	Contrast = .20

Contrast represents the color contrast within the facial image. The less color contrast a face has, the more this metric value approaches 0. The default recommended minimum value for this metric is .45. You can adjust the recommended minimum value by going to **Tools -> Preferences**, clicking on the **Recognition** tab, then adjusting the **For merging** slider in the **Minimum required face contrast quality** section.

Quality Label	Metric Range	Description
Excellent	0.7 - 1.0	Full recognition accuracy can be expected under all conditions.
Good	0.6 - 0.7	Very good recognition accuracy can be expected in general but may confuse closely related family members.
Marginal	0.45 - 0.6	Good recognition but may result in occasional failures.
Poor	0.3 - 0.45	Recognitions can be performed to significant extent but may produce false recognitions.
Very Poor	0.0 - 0.3	Recognitions can still be performed but with significant possibility of confusing similar faces.

19.4 Face Size

Face size defines the minimum required face size in pixels. The metric also includes a margin around the face. The margin is required when learning a face. The face itself (without the margin) includes the area ranging from the top of the forehead to the bottom of the chin and across the full width of the face excluding ears.

The recommended minimum value for this metric is 220 pixels. You can adjust the recommended minimum value by going to **Tools** -> **Preferences**, clicking on the **Recognition** tab, then adjusting the **For learning/strangers** slider in the **Minimum Required Face Size** section.

Note that only the shortest side of the image is used for the purpose of determining the metric value. For example, a facial image that is 200 x 300 (including the margin) would be classified as *Marginal*, since the shortest side (200) falls in the *Marginal* range.

Quality Label	Metric Value	Description
Excellent	260 px and greater	Full recognition accuracy can be expected under all conditions.
Good	210 px - 260 px	Very good recognition accuracy can be expected in general but may confuse closely related family members.
Marginal	160 px - 210 px	Good recognition but may result in occasional failures.
Poor	110 px - 160 px	Recognitions can be performed to significant extent but may produce false recognitions of blurry or otherwise not clearly visible faces.
Very Poor	60 px - 110 px	Recognitions can still be performed but with significant possibility of confusing similar faces.

19.5 Occlusion

Occlusion represents how much of the face is occluded. Faces can be occluded by masks, baseball caps, or even the person's hands held between the face and the camera. The default recommended maximum value for this metric is .5. You can adjust the recommended maximum value by going to **Tools** -> **Preferences**, clicking on the **Recognition** tab, then adjusting the **For learning/strangers** slider in the **Maximum allowed occlusion** section.

Quality Label	Metric Range	Description
Occluded	0.5 - 1.0	At least one of the facial features is not clearly visible thus potentially preventing full recognition accuracy. Recognition based on occluded features will not be possible and incorrect recognition of similar faces occluded in similar manner is possible. Recognition is generally possible as long as two out of three key features (eyes, nose, mouth) are visible.
Not Occluded	0.0 - 0.5	All facial features are clearly visible and full recognition accuracy can be achieved.

19.6 Sentiment

Sentiment represents how happy (a positive sentiment score) or angry (a negative sentiment score) a face is. 0 sentiment (a neutral or serious expression) yields the most accurate facial recognition.

20 Use Event Analytics to Gain Insight

Note: This feature is only available on macOS.

The **Event Analytics** page displays an overview of the number and type of events occurring for each connected camera.

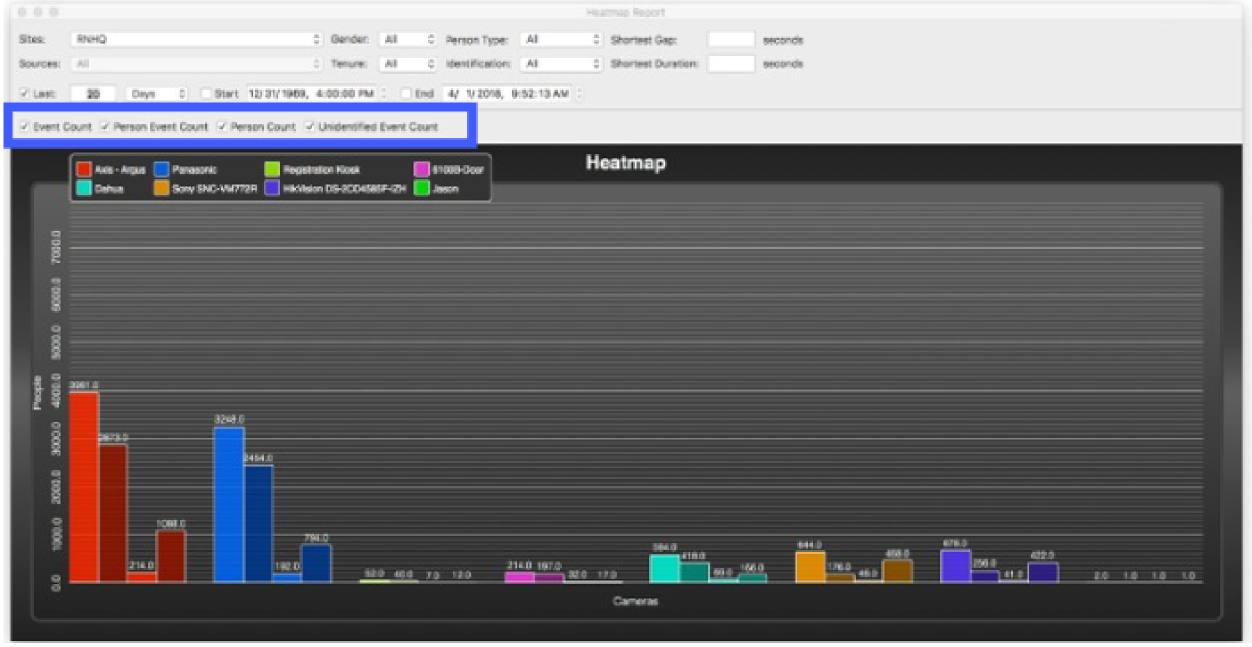
20.1 Use the Heat Map Report

The Event Heat Map shows the number of event occurrences per camera source. It lets you filter, sort, and view data to gain insight into event activity within your SAFR system.

The following types of counts are reported in the Heat Map:

- **Event Count**: Total number of events.
- **Person Event Count**: Number of known person appearance events.
- **Person Count**: Number of unique known persons.
- **Unidentified Event Count**: Number of events associated with unknown persons.

You can show or hide event counts by checking or unchecking the corresponding check boxes.



Heatmap Report

Sites: RNHQ   Gender: All   Person Type: All   Shortest Gap: seconds

Sources: All   Tenure: All   Identification: All   Shortest Duration: seconds

☑ Last: 30 Days   □ Start: 12/31/1969, 4:00:00 PM   □ End: 4/ 1/2018, 9:52:13 AM

☑ Event Count ☑ Person Event Count ☑ Person Count ☑ Unidentified Event Count

20.2 Use Heat Map Filters

You can filter displayed information using the following settings:

- **Sites**: Filter by sites.
- **Sources**: Filter by camera sources.
- **Tenure**: Filters based on a person being new in the chosen time interval or known from before the chosen time interval.
- **Person Type**: Filter by person type.
- **Id Class**: Filters by identity classification. This filter must be one of the following:
  - Unrecognizable
  - Stranger
  - No-Concern
  - Concern
  - Threat

- **Shortest Gap**: Multiple events that occur within the specified number of seconds or less are presented as one combined event.
- **Shortest Duration**: After combining events based on the **Shortest Gap** setting, the **Shortest Duration** setting filters out any events shorter than the specified number of seconds.

21 Actions Overview

In SAFR an action is essentially a script or macro that communicates a desired action in a language or protocol the receiving device or system understands. It can be written in any language supported by the computer where Actions Relay Event Service (ARES) is installed. It only needs to be invocable as an executable directly or through the use of another executable (usually a script interpreter such as Python).

21.1 Actions Components

These are the principle components involved with actions:

- **Actions Relay Event Service (ARES)**: ARES is a cross-platform Java application that acts as an event listener that dispatches configured actions in response to events, as defined in the SAFRActions.config file. ARES can provide replies on any event to be handled by the client originating the event and is normally installed as a service by either the SAFR Platform or SAFR Desktop installers. It is constantly active and is automatically started by the operating system on power-up.
- **SAFRActions.config**: The SAFRActions.config file defines which events will trigger specified actions. It also can specify additional condition constraints before the action(s) will trigger.
- **SAFR Actions**: Only available on macOS and Windows. SAFR Actions is a GUI tool that makes editing the SAFRActions.config file much easier. It presents the JSON information of the config file in a visual and easy to understand manner and offers drop-down menus so you can quickly and easily see what values are available and valid. SAFR Actions makes the JSON element hierarchies easy to understand, and ensures that your changes will validate against the SAFRActions.config JSON schema.

21.2 SAFRActions.config Overview

```
<name: value connection attributes>
rules: [
  {
    event: {  },
    triggers: [
        <time of day and week properties>
        actions: [  ],
        reply: {  },
        conditionalReply: {  },
    ],
    excludeDates: [  ]
  }
]
noTriggerReply: {  }
nFactorDef: [ {  }, { }, ... ]
emailDef: [  {  }, {  }, ... ]
smsDef: [  {  }, {  }, ... ]
```

- **rules**:
    - 1 or more rules can be defined.
    - When an event occurs each rule is checked to see if any of its events match.
    - A rule's event matches an occurring event when:
        - All attributes rules[i].events match the event.
    - Each rule has 1 or more triggers.
        - Each trigger inside a matching rule is fired as long as the time of day conditions match. **Exception**: If 2 *triggerId*s are identical only the first trigger is fired.
    - Each trigger has one or more actions.
        - Actions are either:
            - A shell command or a batch/shell script to be executed.

        - A send email command that has the syntax of: `@emailSend <value of emailDef.label>`
    - All actions are run asynchronously unless a *conditionalReply* is specified in which case the first rule is run synchronously (and the return code of that rule is used for the conditionalReply) while all other rules are run asynchronously.
- *noTriggerReply* is used to perform a reply if none of the triggers are fired.
- *nFactorDef* can define 2 or more conditions that must occur within the specified time window.
- *emailDef* defines one or more email message attributes (subject, from, message, etc).
- *smsDef* defines one or more Short Message Service (SMS) messages.

21.2.1 Examples

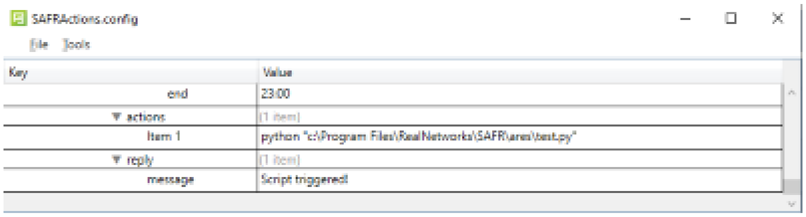
- Send email when visitor arrives during work hours
    - rules
        - Rule 1
            - event (hasPersonId=false)
            - trigger (day/hours: 8-5, M-F)
                - action: @emailSend visitorEmail
    - emailDef
        - label=visitorEmail
        - subject="Visitor Arrived"
        - message="A visitor has arrived at #I - #S."
        - . . .
- Log all events to a CSV and send one type of email for a known person event and another for a threat event.
    - rules
        - Rule1 (known person email)
            - event (hasPersonId=true, idClass=No-Concern)
            - trigger
                - action: @emailSend knownEmail
        - Rule 2 (threat email)
            - event (hasPersonId=true, idClass=[Threat, Concern])
            - trigger
                - action: @emailSend threatEmail
        - Rule 3 (log)
            - trigger
                - action: ".\scripts\log_event.bat "#D" "#N" "#F" . . . "
                    - If editing config file, escape backslash or quotes with another backslash. (In SAFR Actions no escaping is needed.)
                    - The file 'log_event.bat' should be placed in `C:\Program Files\RealNetworks\SAFR\ares\scripts` (for Windows) or `/Library/RealNetworks/SAFR/ares/scripts` (for macOS).
    - emailDef
        - 1 (label=knownEmail, subject, message, etc)
        - 2 (label=threatEmail, subject, message, etc)

21.3 Long File Names

- When using long file names for actions on Windows machines, the file names need to be escaped correctly:

```
"actions": [
           "python \"c:\\Program
               Files\\RealNetworks\\SAFR\\ares\\test.py\""
         ]
```

- Within the SAFR Actions GUI the same entry appears as follows:



Key	Value
end	23:00
▼ actions	(1 item)
Item 1	python "c:\Program Files\RealNetworks\SAFR\ares\test.py"
▼ reply	(1 item)
message	Script triggered!

22 Actions Relay Event Service (ARES)

ARES is a cross-platform Java application that acts as an event listener which dispatches configured actions (i.e. macros) in response to events. The recommended Java version is 9.0.4 or later. ARES can dispatch replies on any event detected by your SAFR system and is normally installed as a service by either SAFR Platform or SAFR Desktop installers. It is constantly active and is automatically started by the operating system on power-up.

22.1 ARES Installation Locations

- For Windows: `C:\Program Files\RealNetworks\SAFR\ares`
- For macOS: `/Library/RealNetworks/SAFR/ares`
- For Linux: `/opt/RealNetworks/SAFR/ares`

22.2 Command Line Options

You can manually start your ARES service by running the following command.

```
java -jar Ares.jar
```

The command line supports the following options:

- **-u <UserId>**: Provides the SAFR account user ID. The user ID that is passed into ARES via the command line overrides whatever user ID is configured within *SAFRActions.config*.
- **-p <Password>**: Provides the SAFR account password. The password that is passed into ARES via the command line overrides whatever password is configured within *SAFRActions.config*.
- **-s**: Saves the user ID and/or password to *SAFRActions.config*. If a password is saved, it's encrypted before it's saved.
- **-q**: Turns on quiet mode, which suppresses most console output.

Linux users should use the **-s** command line option to save their **password** to *SAFRActions.config* so it will be stored encrypted. Windows users should either use the **-s** command line option or the SAFR Actions GUI tool to save their **password** to *SAFRActions.config* so it will be stored encrypted.

22.3 Reconfiguration

- ARES dynamically applies any changes to the *SAFRActions.config* file without restarting:
  - ARES examines the *SAFRActions.config* file every 2 seconds for any changes.
- When a change is detected, ARES reads the change and automatically reconfigures itself accordingly. (Event polling is suspended briefly and then promptly resumed after reconfiguartion is complete.)
- The reconfiguration action is indicated in the log:

  ```
  --- RECONFIGURED at <date>
  ```

22.4 Console Output

- When started, ARES displays any errors or warnings based on the contents of the *SAFRActions.config* file.
- ARES displays all received events, triggered actions, and replies issued unless it was given the **-q** (quiet) option when it was started.

Tip: In the Mac terminal or in the Windows Cygwin shell, the `tail -f ares.log` command is a convenient way to monitor the SAFR Action service in real time.

23 SAFRActions.config

The SAFRActions.config file defines which events will trigger specified actions. You can also specify additional condition constraints before the action(s) will trigger. It also contains basic configuration information so that ARES can communicate with other SAFR components, such as the Event Archive.

23.1 SAFRActions.config JSON Schema

```
{
   environment : "string",
                 <optional,
                 - values: "LOCAL", "DEV", "INT2", "PROD", "Custom"
                 - if not specified assumed PROD >
   eventServer : "string",
                 <optional,
                 - required in case of Custom environment
                 - only affects Custom environment>
   replyServer : "string",
                 <optional,
                 - only affects Custom environment>
   coviServer : "string",
                 <optional,
                 - only affects Custom environment>
   reportServer : "string",
                 <optional,
                 - only affects Custom environment>
   configServer : "string",
                 <optional, "https://cvos.real.com" for production
                    environment
                 - if specified config is retrieved from the cloud using
                    the
                  following address: <configServer>/obj/ares/<aresId> >
   userId :      "string", <optional>
   userPwd :     "string", <optional, encrypted or open text>

   directory :   "string", <required>
   site :        "string", <optional>
   source :      "string", <optional>

   aresId :      "string", <optional>

   maxEventLatency: <long>,  <optional, in milliseconds, default = 8000>

   logActionResponses: <bool>,  <optional, default = false>

   rules: [
      {
        event : {
          type: [ "string", ... , "string" ],
                 <optional, values=(person, badge, action or object),
                    default = all>
          personType: [ "string", ... , "string" ],
                 <optional, default = all, "" = no personType>
          personTags: [
                 [ "string", ... , "string" ],
                 ...
                 [ "string", ... , "string" ]
          ]
                 <optional, default = all>
          tagType: [ "string", ... , "string" ]
                 <optional, values=(april), default = all, "" = no
                    tagType>
          tagId: [ "string", ... , "string" ],
                 <optional, values=(Ids of tagType) default = all, "" =
                    no tagId>
          actionType: [ "string", ... , "string" ],
                 <optional, values=(smileToActivate) default = all, "" =
                    no actionType>
          actionId: [ "string", ... , "string" ],
                 <optional, default = all, "" = no actionId>
          directionId: [ "string", ... , "string" <"left", "right", "up",
             "down"> ],
                 <optional, default = all, "" = no directionId>
          ended: <boolean>,
                 <optional, default = false>
          name: [ "string", ... , "string" ],
                 <optional, default = all, "" = no name>
          company: [ "string", ... , "string" ],
                 <optional, default = all, "" = no company>
          moniker: [ "string", ... , "string" ],
                 <optional, default = all, "" = no moniker>
          personId: [ "string", ... , "string" ],
                 <optional, default = all, "" = no personId>
          hasPersonId: <boolean>,
                 <optional, default = all>
          hasName: <boolean>,
                 <optional, default = all>
          hasMoniker: <boolean>,
                 <optional, default = all>
          hasRootEventId: <boolean>,
                 <optional, default = all>
          gender: [ "string", ... , "string" ],
                 <optional, default = all>
          age: [
                 <optional, default = all>
             {
                min: <float>,
                max: <float>
             },
             ...
          ],
          smile: <boolean>,
                 <optional, default = all>
          avgSentiment: [
                 <optional, default = all>
             {
                min: <float>,
                max: <float>
```

```
          },
          ...
       ],
       liveness: {
              <optional, default = all>
          min: <float>,
          max: <float>
       },
       livenessConfirmed: <boolean>,
              <optional, default = all>
       mask: <boolean>,
              <optional, default = all>
       similarityScore: {
              <optional, default = all>
          min: <float>,
          max: <float>
       },
       occlusion: {
              <optional, default = all>
          min: <float>,
          max: <float>
       },
       site: "string",
              <optional if specified at the root>
       source: "string",
              <optional if specified at the root>
       idClass: [ "string", ... , "string" ],
              <optional, default = all, "" = no idClass>
       directGazeDuration: {
              <optional, default = all>
          min: <long>,
          max: <long>
       }
       objectType: [ "string", ... , "string" ]
              <optional, default = all, "" = no objectType>
       objectId: [ "string", ... , "string" ],
              <optional, default = all, "" = no objectId>
    }
    triggers : [
       {
          triggerId : "string",
              <optional>
          daysOfWeek: ["Mon","Tue","Wed","Thu","Fri","Sat","Sun"],
              <optional, default = all>
          timesOfDay: [
              <optional, default = all>
             {
                start: "11:00",                  <required>
                end: "17:00"                     <required>
             },
             ...
          ],
          actions: [
              <required - can be empty (no actions)>
          "string",
          ...
       ],
       reply: {
           <optional, default = no reply>
          "replyDelay": long,
               <optional, in milliseconds, default = 0>
          "message": "string",
               <optional, default = no message>
          "disposition": double,
               <optional, range [-1 .. 1], default = 1>
          "tags": [ "tag1", ... "tagN" ]
               <optional, default = no tags>
       },
       conditionalReply: [
           <optional, default = no conditional reply>
           {
                "actionResponse": [ integer, ..., integer ],
                     <required>
                "replyDelay": long,
                     <optional, in milliseconds, default = 0>
                "message": "string",
                     <optional, default = no message>
                "disposition": double,
                     <optional, range [-1 .. 1], default = 1>
                "tags": [ "tag1", ... "tagN" ]
                     <optional, default = no tags>
           }
           ...
       ],
    },
    ...
 ],
 excludeDates : [
           <optional, default = none>
          "7/4",
          "12/25",
          "4/10/2017",
          ...
 ],

 triggerFrequencyLimit : {
     <optional, default = unlimited>
    "minSeparationInterval" : long,
         <optional, in milliseconds, default = 0>,
    "spanRootPersonIds": bool,
         <optional, default = false>
    "spanSources" : bool,
         <optional, default = false>
    "spanSites" : bool
         <optional, default = false>
 }

}
```

```
   ...
],
noTriggerReply: {
              <optional, default = no reply>
   "replyDelay": long,
                  <optional, in milliseconds, default = 0>
   "message": "string",
                  <optional, default = no message>
   "disposition": double,
                  <optional, range [-1 .. 1], default = -1>
   "tags": [ "tag1", ... "tagN" ]
                  <optional, default = no tags>
},
nFactorDef: [
   {
      "name": string,
         <required>
      "failOnMismatch": string,
         <optional: "delayed"/"immediate"/"none", default = "delayed">
      "maxDelay": <milliseconds>,
         <optional, default = 60000 (1min)>
      "factors": [
         "<factor_name>|<factor_value>",
         ...
      ],
      "actions": [
         "<action_command>",
         ...
      ]
   },
   ...
],
emailDef: [
   {
      "label": string,
         <required>
      "recipients": [ "recipient1", ... "recipientN" ],
         <required, escape sequences can be used>
      "subject": string,
         <required, escape sequences can be used>
      "cc": [ "cc1", ... "ccN" ],
         <optional, escape sequences can be used>
      "bcc": [ "bcc1", ... "bccN" ],
         <optional, escape sequences can be used>
      "message": string,
         <optional, escape sequences can be used>
      "attachments": [ "attachment1", ... "attachmentN" ],
         <optional, escape sequences can be used
          http://, https://, cvos:// url schemes are supported>
   },
   ...
]
smsDef: [
   {
         "label": string,
            <required>
         "recipients": [ "recipient1", ... "recipientN" ],
            <required, escape sequences can be used, phone numbers using
               the the E.164 format required>
         "maxPrice": string,
            <optional>
         "message": string,
            <optional, escape sequences can be used>
      },
      ...
   ],
}
```

- Events that are older than maxEventLatency will be ignored. Event time is defined as the difference between the time the event was generated - as measured by the SAFR Cloud (or machine Platform is running) and the time the event is processed – as measured on the machine the SAFR Actions app is running.

23.2 rules

23.2.1 event

- For rules.events that allow arrays, the new event must contain all the specified array elements to match. For example, if a config file specified rules.events.personType as follows:

  ```
  personType: [
      "staff",
      "admin",
      "guest"
  ],
  ```

  Then the new event's personTags array would have to have all 3 specified personTypes for it to match the rule.

- personTags: all elements in one of sub-arrays need to exist in event's personTags array to match the rule.

23.2.2 trigger

- Event (id) can trigger actions only once (albeit multiple triggers can be activated simultaneusly).
- Event (id) can trigger replies only once per reply context (triggered, notTriggered). Multiply replies can be triggered simultaneously (one reply per triggered action).
- triggerId - ID Unique within the triggers array used in rare case where you want only 1 trigger to fire. If triggerId is same on 2 or more, only 1st of all matching get triggered.
- Useful if date filters are overlapping and during overlap times only wish to actions from single trigger.

23.2.3 conditionalReply and reply

- disposition refers to how the reply should be perceived by the recipient:
  - Replies with disposition in range [-1 .. 0 > are interpreted as negative replies and can thus be expected to be presented (color, sound, voice) in manner consistent with rejection.
    - Value of 0 is a neutral reply and can thus be expected to be presented in a neutral manner (color, sound, voice).
    - Replies with disposition in range <0 .. 1] are interpreted as positive replies and can thus be expected to be presented (color, sound, voice) in manner consistent with acceptance.

- When conditional reply is specified, non-conditional reply is used only as catch-all if none of the action response codes match.
- When conditional reply is specified, execution of the FIRST action in trigger will occur in blocking manner to enable retrieval of the response code from that FIRST action.
  - If any other actions are specified, they will be performed in non-blocking manner and their response codes will not be retrieved or used.
- When conditional reply is not specified, execution of all actions will occur in non-blocking manner.
- A reply is generated as follows:
  - One or more matching conditionalReply entries are sent
  - In addition, either the reply or noTriggerReply is sent
- URL used to post the reply: `<replyServer>/stream/reply.<Base64(event Id)>`
  - By default the reply is posted to the CVOS server (replyServer)
  - POST is a file of the following format.
  - The reply object (JSON file) can be obtained by querying the CVOS server after some delay after the event was fired

23.2.4 actions

- Each action is a command string that will be executed.
- Commands are executed asynchronously unless conditionalReply is set
- If conditionalReply is set, the first command is executed synchronously.
- Some Windows programs (particular Windows programs that do not have a message pump) may not run in background and block until the command returns.
- If multiple actions are defined, each action is executed in sequence.
- For information on the syntax for emails, see Email Actions below.
- For information on the syntax for SMS notifications, see SMS Actions below.

23.3 Action and Reply Message Escape Sequences

```
#N - name
#F - first name (name prefix up to first white-space)
#U - surname (name postfix: staring after first white-space sequence to
   the end of name string)
#T - person type
#S - source
#I - site
#D - person id
#R - root person id
#E - person external id
#G - gender
#A - age       (###)
#M - sentiment (#.##)
#L - smile     (true/false)
#V - event type
#v - event id
#B - tag type
#C - action type
#b - tag id
#c - action id
#k - direction id
#s - event start time (milliseconds since epoch)
```

```
#r - event start date/time (local time)
#p - validation phone
#e - validation email
#H - home location
#t - personTags (comma separate list of personTags)
#O - company
#m - moniker
#<d>m - moniker substring (delimited by white-space)
        indexed by single decimal digit 0-9 .  E.g.:  #0m or #3m
#l - similarityScore (#.####)
#a - idClass
#Z - directGazeDuration
#o - objectType
#d - objectId
#u - occlusion (#.##)
#i - liveness (#.##)
#n - livenessConfirmed (true/false)
#z - mask (true/false)
```

23.4 N-factor Actions

- nFactor actions are started via internal @nFactorStart actions within the standard trigger actions array:

```
{
   triggerId : "string",
   ...
   actions: [
      "@nFactorStart <name>",
      ...
   ],
   reply: {
      ...
   },
   conditionalReply: [
      ...
   ]
}
```

When the action starts, the following occurs:

- @nFactorStart actions are first resolved for escape sequences
- factors (names and values) defined in the corresponding nFactorDef are also resolved for escape sequences
- actions defined in the corresponding nFactorDef are also resolved for escape sequences
- eventStartTime is retrieved from the triggering event

Response codes for nFactorStart actions:

- 0 = nFactor monitoring for action started successfully

nFactorStart-ed actions are resolved via nFactorResolve commands. When all factors needed for the actions are resolved, actions are executed:

```
{
   triggerId : "string",
   ...
   actions: [
      "@nFactorResolve <name> <factor_name>|<factor_value>",
        ...
    ],
    reply: {
        ...
    },
    conditionalReply: [
        ...
    ]
}
```

- At the time of resolving the following occurs:
    - @nFactorResolve actions are first resolved for escape sequences.
    - Each factor can be resolved by at most one not yet resolved factor requirement.
- Response codes for nFactorResolve actions:
    - 0 = resolved last unresolved factor
        - Executed action response supersedes
    - >=1 resolved other than last unresolved factor
    - -1 = no matching <Site>/<Source>/<name>
    - -2 = <mismatched factor - ignored since failOnMismatch = none>
    - -3 = <matches but already resolved>
    - -4 = <matches but too late to resolve>
    - -5 = <mismatched factor - error since failOnMismatched = delayed/immediate>
    - -6 = unknown (i.e. not defined in nFactorDef) factor_name
- @nFactorStartOrResolve combines starting and resolving into one action. It's usually used for generating pseudo events from monikers.

    ```
    {
        triggerId : "string",
        ...
        actions: [
            "@nFactorStartOrResolve <name> <factor_name>|<factor_value>",
            ...
        ],
        reply: {
            ...
        },
        conditionalReply: [
            ...
        ]
    }
    ```

@personEventFromMoniker action generates a pseudo person event from moniker created by combining all the resolved factor values (separated by space) in the order listed in factors array. The generated event is of type *person* which is populated with the meta-data of person with moniker matching the assembled moniker value.

```
{
    nFactorDef : [ {
        factors : [
                "moniker|**",
                "moniker|1**",
                "moniker|2**",
                "moniker|3**"
```

```
      ],


      actions : [
            "@personEventFromMoniker"
      ]
      }
   ]
}
```

23.5 Email Actions

To send emails using actions, you must do the following:

1. Obtain an SMTP server account that you can use to send emails.
2. Configure SAFR so that it's ready to use your SMTP server account to send emails. You can do this from the Status page of the Web Console. On Windows machines, you can also do this via **Tools -> Configure Email Server** in SAFR Actions.
3. Configure the emailDef section of the SAFRActions.config, as described below. Note that your emailDef section can define multiple emails, each one being identified by the `label` field.

```
emailDef: [
    {
        "label": string,
            <required>
        "recipients": [ "recipient1", ... "recipientN" ],
            <required, escape sequences can be used>
        "subject": string,
            <required, escape sequences can be used>
        "cc": [ "cc1", ... "ccN" ],
            <optional, escape sequences can be used>
        "bcc": [ "bcc1", ... "bccN" ],
            <optional, escape sequences can be used>
        "message": string,
            <optional, escape sequences can be used>
        "attachments": [ "attachment1", ... "attachmentN" ],
            <optional, escape sequences can be used
                http://, https://, cvos:// url schemes are supported>
    },
]
```

   - **label**: The label used to identify this particular email.
   - **recipients**: One or more email addresses where the email will be sent.
   - **subject**: The text that will appear in the email's subject line.
   - **cc**: List of email addresses that will be cc'ed on the email.
   - **bcc**: List of email addresses that will be bcc'ed on the email.
   - **message**: The text that will be the body of the email.
   - **attachments**: The location of any attachments you want to attach to the email.

4. In the `actions` field of SAFRActions.config, enter a string with the following syntax: "@emailSend <label>", where <label> = the label of whichever email within your SAFRActions.config that you want to use.

23.6 SMS Actions

To use Short Message Service (SMS) notifications within actions, you must do the following:

1. Obtain an AWS account which is configured for your region so it can send SMS messages.

2. Configure SAFR so that it's ready to use your AWS account to send SMS notifications. You can do this from the Status page of the Web Console. On Windows machines, you can also do this via **Tools -> Configure SMS Sender** in SAFR Actions.

3. Configure the smsDef section of the SAFRActions.config, as described below. Note that your smsDef section can define multiple SMS messages, each one being identified by the `label` field.

```
smsDef: [
   {
      "label": string,
         <required>
      "recipients": [ "recipient1", ... "recipientN" ],
         <required, escape sequences can be used, phone numbers using the
            the E.164 format required>
      "maxPrice": string,
         <optional>
      "message": string,
         <optional, escape sequences can be used>
   },
]
```

   - **label**: The label used to identify this particular SMS message.
   - **recipients**: The list of recipients to receive the SMS message, formatted using the E.164 format. (e.g. +2065551313)
   - **maxPrice**: The maximum amount in USD that you are willing to spend to send the SMS message. Amazon SNS will not send the message if it determines that doing so would incur a cost that exceeds the maximum price. See the description of the `AWS.SNS.SMS.MaxPrice` attribute here for more information about this field.
   - **message**: The text message to be sent.

4. In the `actions` field of SAFRActions.config, enter a string with the following syntax: "@smsSend <label>", where <label> = the label of whichever SNS message within your SAFRActions.config that you want to use.

24 SAFR Actions

SAFR Actions is a GUI tool to aid users to edit the SAFRActions.config configuration file. It comes already installed with SAFR Platform and SAFR Desktop.

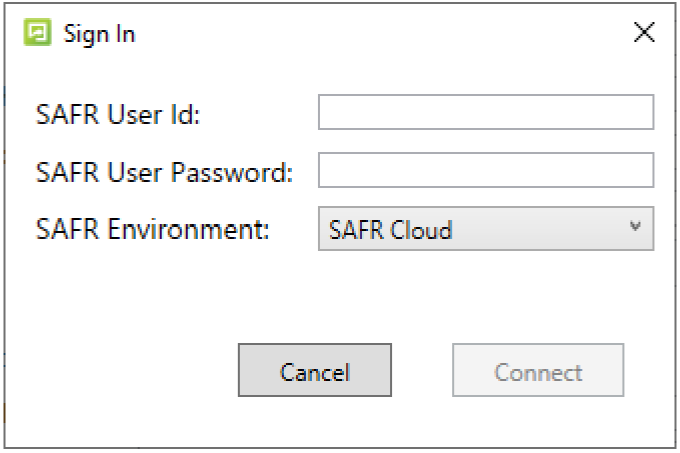
SAFR Actions is generally very light in terms of system resource usage but can be burdensome if the rate of events requiring handling is high (for example, hundreds per second) and actions scripts are computationally or I/O intensive. However, this is not a common occurrence.

24.1 Configure Email Server

Only available on Windows.

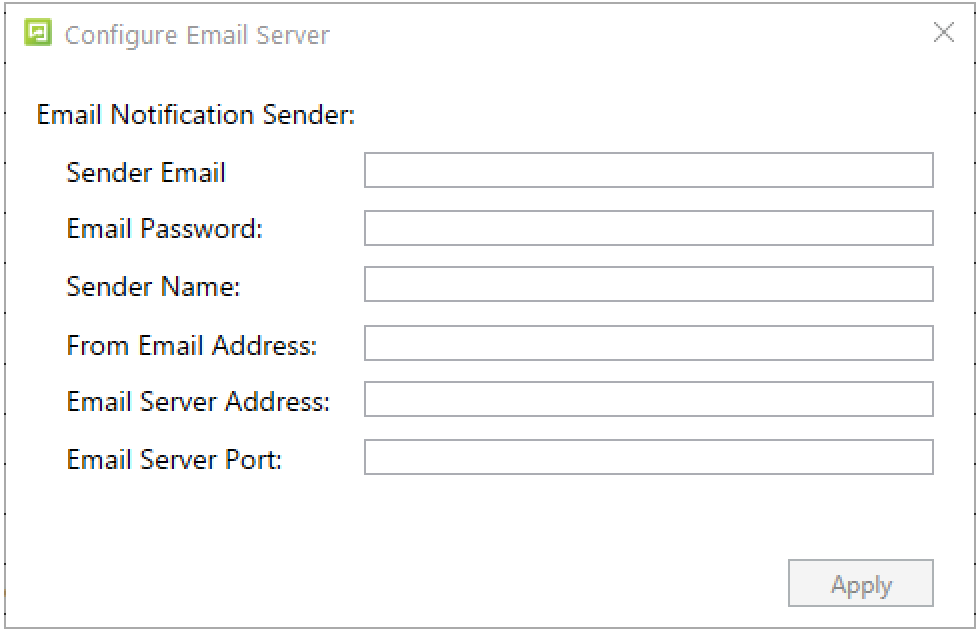
Enables SAFR's actions to send emails. Before you can configure SAFR to send emails, make sure you obtain an SMTP server account that you can use to send emails.

1. Get an SMTP Server account you can use for sending emails.
2. Within the SAFR Actions GUI, select **Configure Email Server...** from the **Tools** drop down menu at the top.
3. You'll be prompted to enter your SAFR credentials.



SAFR Environment: SAFR Cloud

4. You'll be presented with the following dialogue. Fill out all the fields of the dialogue.



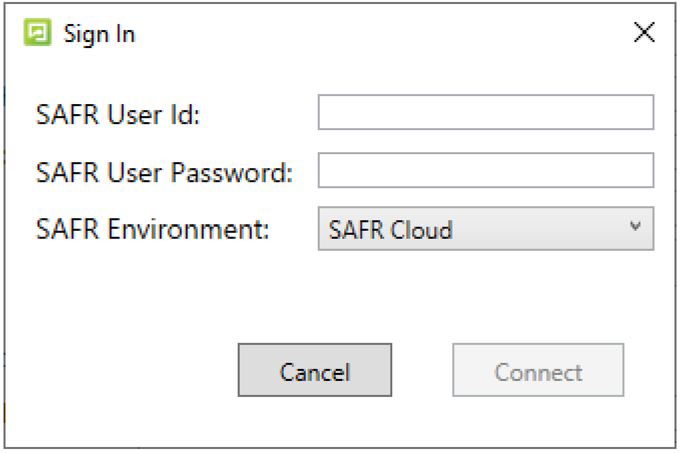
- **Sender Email**: The email username of the SMTP account. (e.g. Susan.Johnson@gmail.com)

   - **Email Password**: The password for the SMTP account.
   - **Sender Name**: The display name on the “From” line. (e.g. Susan Johnson)
   - **From Email Address**: The email address that will appear on the “From” line. This feature isn’t supported by all email servers; if this field isn’t used then the *Sender Email* value is used for the “From” line.
   - **Email Server Address**: The address of the SMTP email server.
   - **Server Port**: The email server port. The default port for SMTP is 587.
5. Click **Apply**. ## Configure SMS Sender

Only available on Windows.

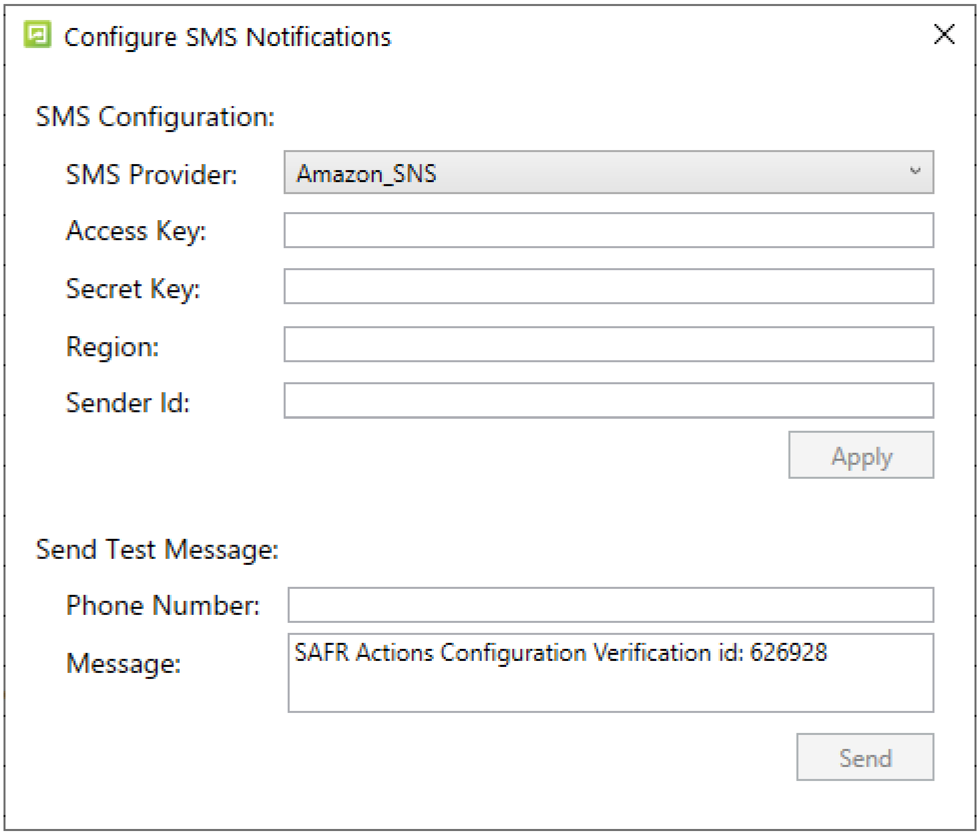
To configure SAFR so that it can send short message service (SMS) messages, do the following:

1. Set up an AWS account. Make sure your AWS account is configured for your region so that it can send SMS messages.
2. Within the SAFR Actions GUI, select **Configure SMS Sender...** from the **Tools** drop down menu at the top.
3. You’ll be prompted to enter your SAFR credentials.



SAFR Environment: SAFR Cloud

4. You’ll be presented with the following dialogue. Fill out all the fields of the dialogue.



Configure SMS Notifications

SMS Configuration:

SMS Provider: Amazon_SNS

Access Key:

Secret Key:

Region:

Sender Id:

Send Test Message:

Phone Number:

Message: SAFR Actions Configuration Verification id: 626928

- **SMS Provider**: The SMS provider that you're using. This value will always be `Amazon_SNS`.
- **Access Key**: Your Amazon SNS Access Key.
- **Secret Key**: Your Amazon SNS Secret Key.
- **Region**: The region of your Amazon SNS.
- **Sender Id**: The name that will be used to send the SMS messages.
- **Send Test Message**: Configure the test message that will be sent after you finish setting up SMS.
  - **Phone Number**: The phone number to which the test message will be sent. It should be in the E.164 format. (e.g. +2065551313)
  - **Message**: The text message that will be sent to the phone number specified above.

5. Click **Apply**, then click **Send**

25 SAFR Server Clusters

At some point, your SAFR system's capacity and/or performance may degrade if the number of face recognition requests sent to your SAFR Server overwhelms your server's capacity. (Performance problems may also arise if the number of people in your Person Directory becomes too large.) Fortunately, you can install additional SAFR Servers on other machines in order to increase your SAFR system's capacity, improve performance, and improve resiliency. The first SAFR Server you install is automatically your primary server, while all additional servers are secondary servers.

In order to install additional servers, you must first install an SSL certificate on your primary server. See SSL Certificate Installation for information about how to do this.

Note: You can change which machine is the primary server by uninstalling the primary server, waiting 24 hours, and then re-installing the SAFR Server on a different machine. The 24 hour wait time can be avoided if you contact your SAFR Account Manager and ask them to manually reset your IP address.

25.1 Understand When to Scale

A single SAFR Server that's also running a Desktop Client can handle up to 16 cameras, (assuming each camera view contains just a single face), as long as the host machine meets the recommended hardware requirements If the machine running the server doesn't have any cameras connected directly to it, then the server's capacity increases to 25 cameras, again assuming that each camera view contains a single face. A higher number of faces per camera or a higher number of cameras requires either vertical scaling of a single server (i.e. more or faster CPUs) or horizontal scaling by installing more SAFR Servers.

Another possible performance bottleneck is the network throughput of the primary server. You may want to monitor its network throughput during maximum concurrency times to make sure the network is not over-saturated.

25.2 Load Balancing Configurations

For prescribed deployments, the system requirements of the Desktop Client need to be combined with those of SAFR Server. A single Desktop Client typically handles up to 16 cameras as long as it is equipped with a GPU card (see SAFR System Requirements). In this way, running SAFR Server and the Desktop Client on the same machine using the recommended configuration can host up to 16 cameras, assuming each camera view contains with a single face.

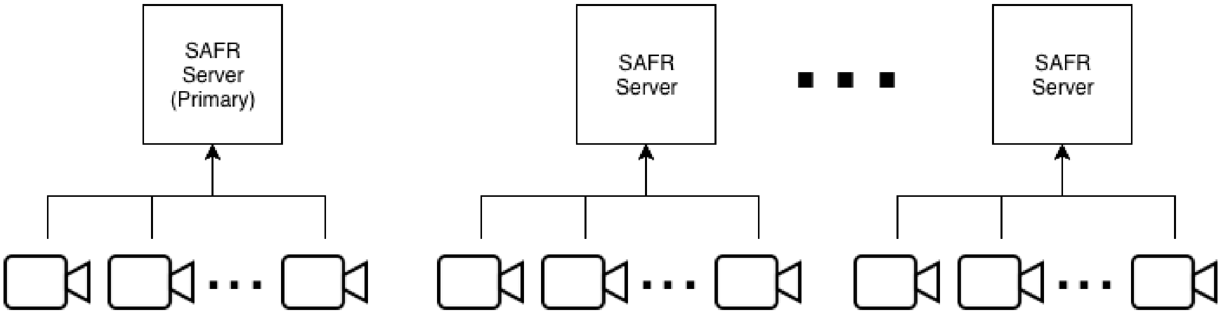
There are three different load balancing configurations you can choose from.

- **Prescribed Load Balancing Configuration**: Cameras are connected to Desktop Clients or Video Recognition Gateway (VIRGO) video feeds running on the same machines that are hosting your SAFR Servers. This gives you tight control over how your face recognition load is distributed, since the video feeds' face recognition requests are processed on the same machine where the video feeds are connected. The system requirements of the Desktop Client need to be combined with those of SAFR Server when calculating the system requirements for a machine hosting the SAFR Server and Desktop Client. A single Desktop Client typically handles up to 16 cameras as long as it is equipped with a GPU card (see SAFR System Requirements). Thus, a machine running SAFR Server and the Desktop Client which meets the recommended system requirements can host up to 16 cameras, assuming each camera view contains just one single face.
- **Software-Based Load Balancing Configuration**: In this configuration the machines hosting SAFR Servers do not also have cameras connected to them. All face recognition requests are initially sent to the primary server, and the primary server acts as the load balancer for the server cluster.
- **External Load Balancing Configuration**: In this configuration all face recognition requests are directed at one or more external load balancer(s), which handle load balancing duties for the SAFR system.

25.2.1 Prescribed Load Balancing Configuration

In the prescribed configuration, you run multiple SAFR Servers by connecting cameras to Desktop Clients or VIRGO video feeds running on the same machines that are hosting SAFR Servers. In this way, you have tight control over which servers take the video feed load. This is also a useful configuration for systems with very low video feed count totals where running a Desktop Client on a separate machine from the SAFR Server would take more resources than are required for the given use case.

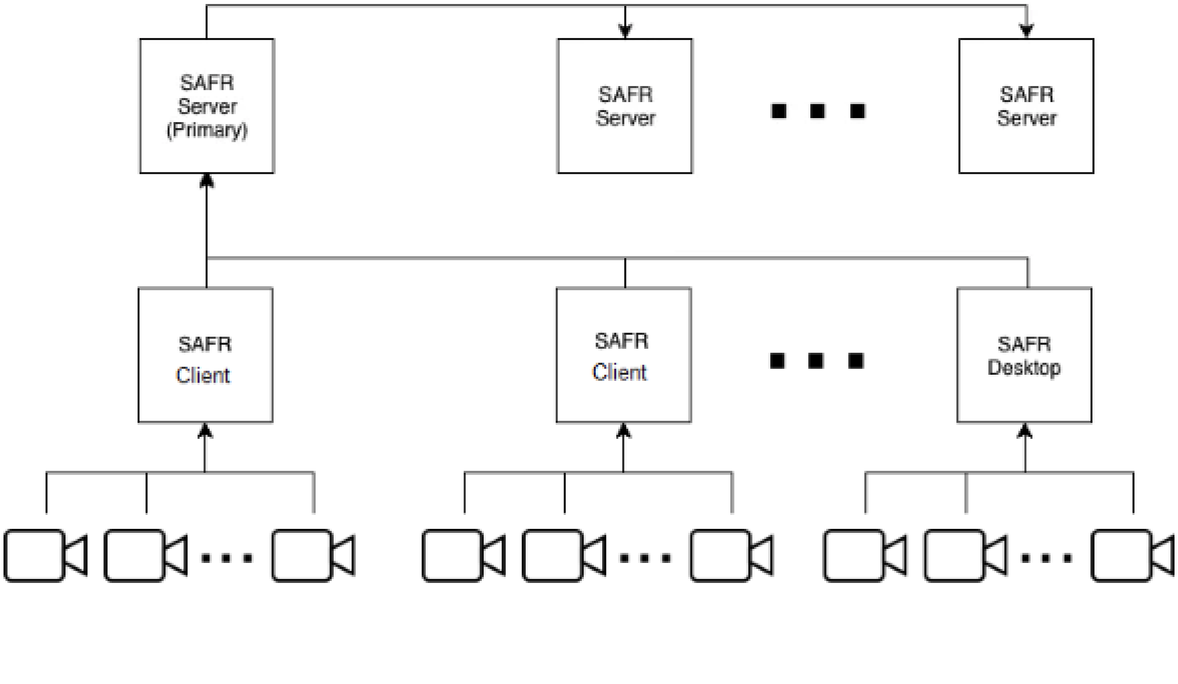
The following diagram illustrates this configuration:



Most services (e.g. face service, events, and reports) are performed on the server where recognition requests are sent.

25.2.2 Software-Based Load Balancing Configuration

In the software-based load balancing configuration, cameras aren't connected to machines running SAFR Servers. When newly installed secondary servers are configured, they check in with the primary server and announce that they're ready to receive load-balanced traffic. All recognition requests go through the primary server. which balances the load among itself and all other servers in the SAFR system. The following illustration demonstrates this setup:

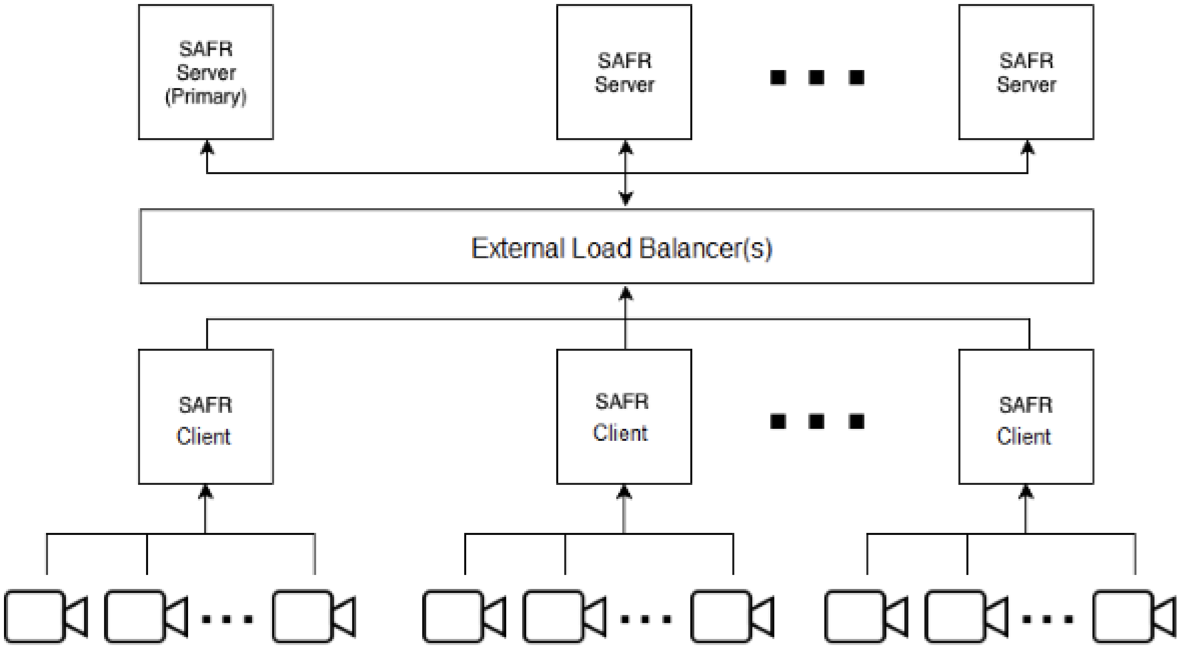


25.2.3 External Load Balancing Configuration

The software-based load balancing configuration has the limitation that the primary server is a single point of failure. All traffic is routed through the primary server before any traffic is redirected to the rest of the servers. If the primary server is down, all traffic will stop. External load balancing is an alternate configuration that can be used to provide a more robust setup that can better deal with server failure.

When using an external load balancing configuration, all network traffic is first routed to one or more load balancer(s), and the load balancer(s) proxy requests to the backend servers over either HTTP or HTTPS. HTTP would be OK in situations where network traffic is isolated to a trusted network, or when network sniffing by non-target hosts is impossible.

If HTTPS is used to proxy traffic to SAFR servers, you should manually disable load balancing on all secondary servers as described below so that the primary server isn't double load balancing traffic to them. A valid (i.e. non self-signed) SSL certificate would still need to be installed and configured on the primary server. Secondary servers should be fine with the default (i.e. self-signed) certificate, if your load balancer allows it.



25.2.4 Manually Configure Load Balancing Traffic

SAFR Servers can be manually enabled or disabled to accept load balancing traffic.
Note: If the server you want to disable is the only one configured to take traffic, you receive a warning and prompt to continue. In this case, should you proceed, your system will most likely go offline.

Disable Load Balancing Traffic

To stop receiving traffic on a server, log in to a shell on the server and run the appropriate command for your server's OS:

OS	Command
Windows	`"C:\Program Files\RealNetworks\SAFR\bin\server-status.py" --disable`
macOS	`/Library/RealNetworks/SAFR/bin/server-status.py --disable`

OS	Command
Linux	`sudo /opt/RealNetworks/SAFR/bin/server-status.py --disable`

It may take up to one minute for the desired traffic state to change.

Enable Load Balancing Traffic

To resume receiving traffic on a server, log in to a shell on the server and run the appropriate command for your server's OS:

OS	Command
Windows	`"C:\Program Files\RealNetworks\SAFR\bin\server-status.py" --enable`
macOS	`/Library/RealNetworks/SAFR/bin/server-status.py --enable`
Linux	`sudo /opt/RealNetworks/SAFR/bin/server-status.py --enable`

It may take up to one minute for the desired traffic state to change.

26 Add Secondary Servers

The first SAFR Server you install will automatically become the primary server. All subsequent servers you install will be secondary servers. There are two types of secondary servers:

- **Simple**: Does not replicate the database data.
- **Redundant**: Replicates database data from the primary server, possibly providing failover functionality. (See the Database and Object Storage Redundancy topic for details.)
  Note: Only Windows and Linux SAFR Servers can become redundant secondary servers.

26.1 Add a Secondary Server While Connected to the Internet

If your system is connected to the Internet, do the following to add a secondary server:

1. Download and install SAFR Platform on the additional machine.
2. Log in to the he SAFR auto-discovery process:
   - Connect the Desktop Client to the primary server (for macOS and Windows) as described here.
   - Connect your Web Console to the primary server (for Linux) as described here.
3. During auto-discovery, the following automatically happens:
   1. The secondary server contacts a SAFR Licensing Server in the cloud to acquire a license.
   2. The SAFR Licensing Server authenticates the SAFR account credentials.
   3. The SAFR Licensing Server identifies the license and deployment type.
   4. A suitable license is returned to the secondary server and information about the primary server is returned to the secondary server, including the hostname.
4. If your new secondary server is on a Windows or Linux machine, you will be prompted to choose which kind of secondary server you want: simple or redundant. If your new secondary server is on a macOS machine no prompt will occur; macOS secondary servers are always simple.
5. Auto-discovery will now continue, with the following automatically occurring:
   1. The secondary server re-configures itself to reference the primary server.
   2. The secondary server registers itself with the primary server.
   3. The primary server updates its local database.
   4. If you're using a Software-Based Load Balancing Configuration, the primary server now adds the new secondary server to its load balancer configuration and uses it as an additional node in its cluster.

26.2 Add a Secondary Server While Offline

If you are not connected to the Internet, you can still connect your new secondary server to the primary server, but the auto-discovery process is not available. You must instead manually configure the newly installed secondary server to locate the primary server. If your new secondary server is on a Windows or Linux machine, you'll need to choose which kind of secondary server you want: simple or redundant. If your new secondary server is on a macOS machine no such decision is required; macOS secondary servers are always simple.

1. Download and install SAFR Platform on the second machine.

2. Run the *safr-worker* script on your secondary server by doing the following:

   For macOS:

   1. Open Terminal.

   2. Run the following command, substituting the primary SAFR hostname for HOSTNAME:

      ```
      sudo /Library/RealNetworks/SAFR/bin/safr-worker HOSTNAME
      ```

   For Windows

1. On the primary server record the contents of `C:\ProgramData\RealNetworks\SAFR\mongo\.adminpass` and `C:\ProgramData\RealNetworks\SAFR\mongo\mongod.keyfile`
2. On the new secondary server, open a command prompt by right-clicking on the **Start** menu, selecting **Run**, and entering `cmd`.
3. If you want it to be a simple secondary server, in the new command prompt run the following command, substituting the password from .adminpass in Step 1 for PASSWORD and the primary server hostname for HOSTNAME:

   ```
   python "C:\Program Files\RealNetworks\SAFR\bin\safr-worker.py" -p
      PASSWORD HOSTNAME
   ```

   OR
4. If you want it to be a redundant secondary server, in the new command prompt run the following command, substituting the `mongod.keyfile` contents from Step 1 for KEYFILE, the password from .adminpass in Step 1 for PASSWORD, and the primary server hostname for HOSTNAME:

   ```
   python "C:\Program Files\RealNetworks\SAFR\bin\safr-worker.py" -s KEYFILE
      -p PASSWORD HOSTNAME
   ```

For Linux

1. On the primary server, record the contents of `/opt/RealNetworks/SAFR/mongo/.adminpass` and `/opt/RealNetworks/SAFR/mongo/mongod.keyfile`
2. On the primary server, open `/opt/RealNetworks/SAFR/virgo/config/virgo-factory.conf`
3. Within that file, look for a section that looks something like:

   ```
   "global":{
       "environment": "CUSTOM",
       "user-id": "ubuntu18int2tst",
       "user-encrypted-password":
           "%qy4Effq2cxYUrEopuIFSY3LE22hDBMOG6NdeqTWfok4=",
       "status-interval":5000,
       "remote-control-enabled":true
   },
   ```

4. Record the “user-id” and “user-encrypted-password” values. These will be your SAFRUSER and SAFRPASSWORD values in the steps below.
5. If you want it to be a simple secondary server, on the new secondary server run the following command.

   ```
   sudo python /opt/RealNetworks/SAFR/bin/safr-worker.py -p PASSWORD -u
      SAFRUSER -x SAFRPASSWORD HOSTNAME
   ```

   where:

   - **PASSWORD** = the password from .adminpass in Step 1
   - **SAFRUSER** = “user-id” from Step 4. Don’t include the enclosing double quotation marks.
   - **SAFRPASSWORD** = “user-encrypted-password” from Step 4. Don’t include the enclosing double quotation marks.
   - **HOSTNAME** = the primary server hostname

   OR
6. If you want it to be a redundant secondary server, on the new secondary server run the following command.

```
sudo python /opt/RealNetworks/SAFR/bin/safr-worker.py -s KEYFILE -p
    PASSWORD -u SAFRUSER -x SAFRPASSWORD HOSTNAME
```

where:

- **KEYFILE** = the contents of `mongod.keyfile` from Step 1
- **PASSWORD** = the password from .adminpass in Step 1
- **SAFRUSER** = “user-id” from Step 4. Don’t include the enclosing double quotation marks.
- **SAFRPASSWORD** = “user-encrypted-password” from Step 4. Don’t include the enclosing double quotation marks.
- **HOSTNAME** = the primary server hostname

26.3 Error Messages

When attempting to join a new secondary server, you might encounter the following error messages:

Error Message	Description
System is offline	Network or system connectivity issue. Attempt to access the system at a later time.
SAFR master host is not reachable	Ensure all servers are connected to the same network and try again.
Improperly configured SSL certificate	SSL certificates are required to set up multiple servers. See the SSL Certificate Installation page for information about how to install an SSL certificate.
Secure connection error. Check server for valid SSL certificate	SSL certificates are required to set up multiple servers. See the SSL Certificate Installation page for information about how to install an SSL certificate.
Incomplete server connection	Attempt to join again; a persistent issue may require either uninstalling and reinstalling SAFR Platform on your servers or contacting your SAFR support representative.

26.4 Secondary Server Health Checks

- At startup each server, both primary and secondary, registers itself by posting its status to the database on the primary server.
- The primary server directs requests to all secondary servers in a *least connection method* that keeps the load evenly balanced among all secondary servers.
- As long as a given secondary server remains healthy, the primary server keeps that secondary server in its load balance rotation.
- Status information about all secondary servers is stored in the database on the primary server.
- Every minute all servers (the primary server as well as the secondary servers) send a status update to the database on the primary server.
- Every five seconds, the primary server attempts to ping all servers (the primary server as well as all secondary servers) via the SAFR heath check API.
- If the health check fails for a given secondary server for 15 seconds (i.e. for 3 health check API calls in a row), that secondary server is removed from load balancing rotation and face recognition requests are no longer routed to it. If the health check succeeds for the removed secondary server for ten seconds (i.e. for 2 health check API calls in a row), the secondary server is returned to the load balancing rotation and resumes accepting face recognition requests.
- If a secondary server’s status has not been reported for over five minutes, it is removed from the load balancer configuration. In this case, it is no longer sent face recognition requests or health check API

calls.

- If a secondary server has been pulled out of rotation for not responding to health checks, or is removed from the load balancer configuration for not reporting status for more than five minutes, it can still be put back in rotation through any of the following:
    - If a network interruption prevents the secondary server from sending a request, the secondary server continues to send a status update at its regularly scheduled interval after it goes back online and its status is updated in the primary server.
    - If the secondary server is restarted, it sends a status update after all services are started and ready.
    - If the secondary server IP address is changed, the secondary server must be manually restarted to force it to send a status update to the primary server with the new IP address.

27 Database and Object Storage Redundancy

27.1 Database Redundancy

The first SAFR Server you install will automatically become the primary server. All subsequent servers you install will be secondary servers. There are two types of secondary servers:

- **Simple**: Does not replicate database data.
- **Redundant**: Replicates database data from the primary server, possibly providing failover functionality. (See the Failover Functionality section below for details.)
  Note: Only Windows and Linux SAFR Servers can become redundant secondary servers.

With both types of secondary servers services such as feed management, reports, and the Web Console are not load-balanced and are always served from the primary server.

27.1.1 Failover Functionality

If there are at least two redundant secondary servers (three servers total), failover functionality is enabled. This means that if the primary server goes offline and both of the first two installed redundant secondary servers are still online, one of the redundant secondary servers will become the new primary server and the server cluster will continue to function as normal.

If additional redundant secondary servers are installed beyond the first two, database data will be replicated on them, but they don't count for the purpose of failover functionality.

27.2 Object Storage Redundancy

Note: Object Storage Redundancy is only available on Windows and Linux.

The Object Storage Service is used for storing objects, such as profile and event images, as well as ephemeral data, such as event reply messages.

The service can operate in a redundant configuration when you have multiple SAFR servers running. All redundant secondary servers are load-balanced by the primary server for all Object Storage Service requests it receives.

27.2.1 Shared Object Storage (Network Storage)

Using shared object storage provides a shared location for each server to save and retrieve objects from. This provides each Object Storage Server with access to all of the objects, rather than just objects saved to their local storage.

Shared storage also provides an easier backup process, as you only have to run it from the primary server.

27.2.2 Local Object Storage (Not Recommended)

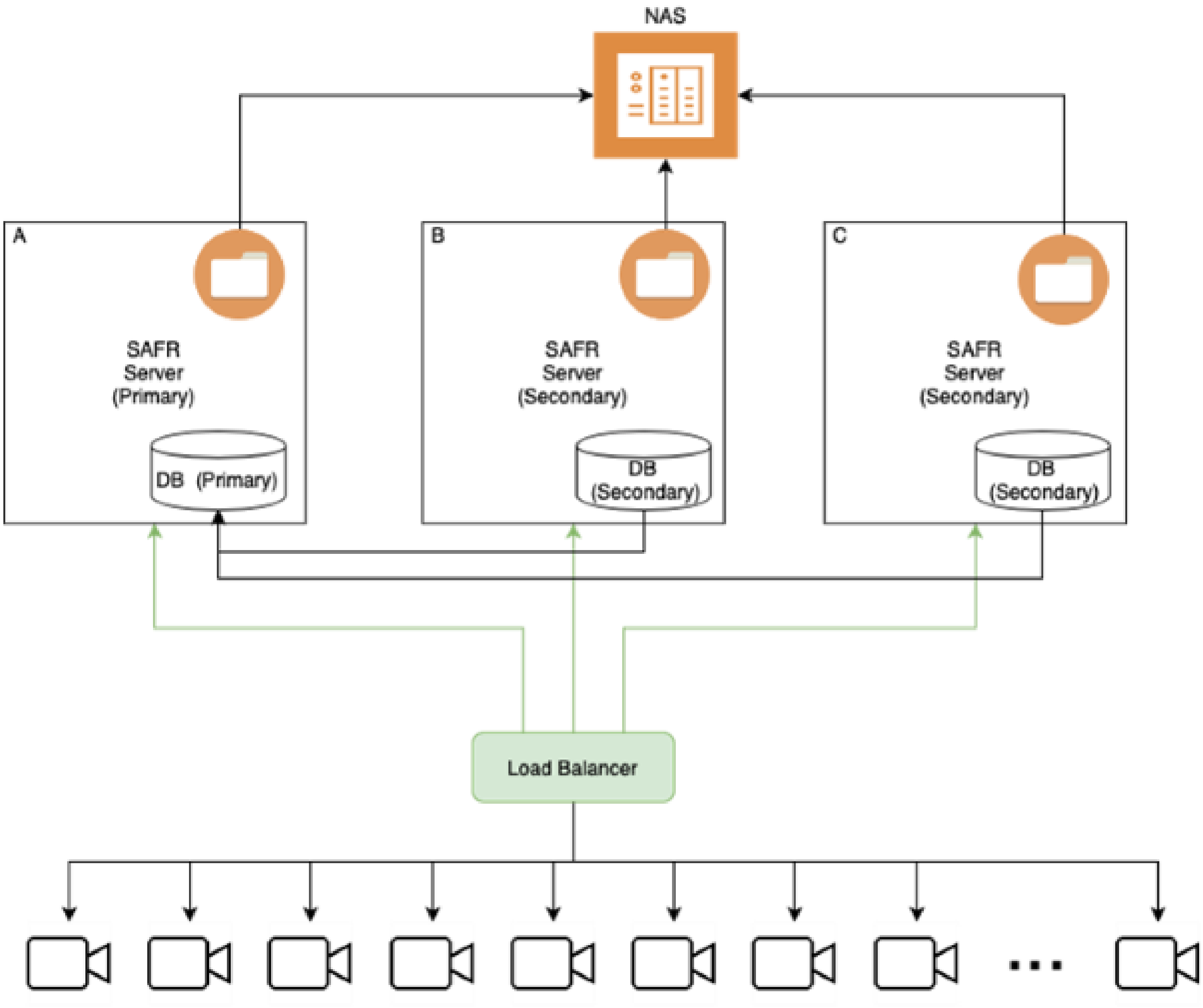
By default all redundant servers will save objects locally, and ask other Object Storage Servers for objects it does not have locally.

When you're using local object storage, you will lose access to all objects that are only stored by an offline Object Storage Server until the server becomes healthy again. If that server's objects are lost, and you do not have backups, they will be unrecoverable.

Backups must be run on every redundant server that has Object Storage enabled.

27.3 External Load Balancing (ELB) Walkthrough

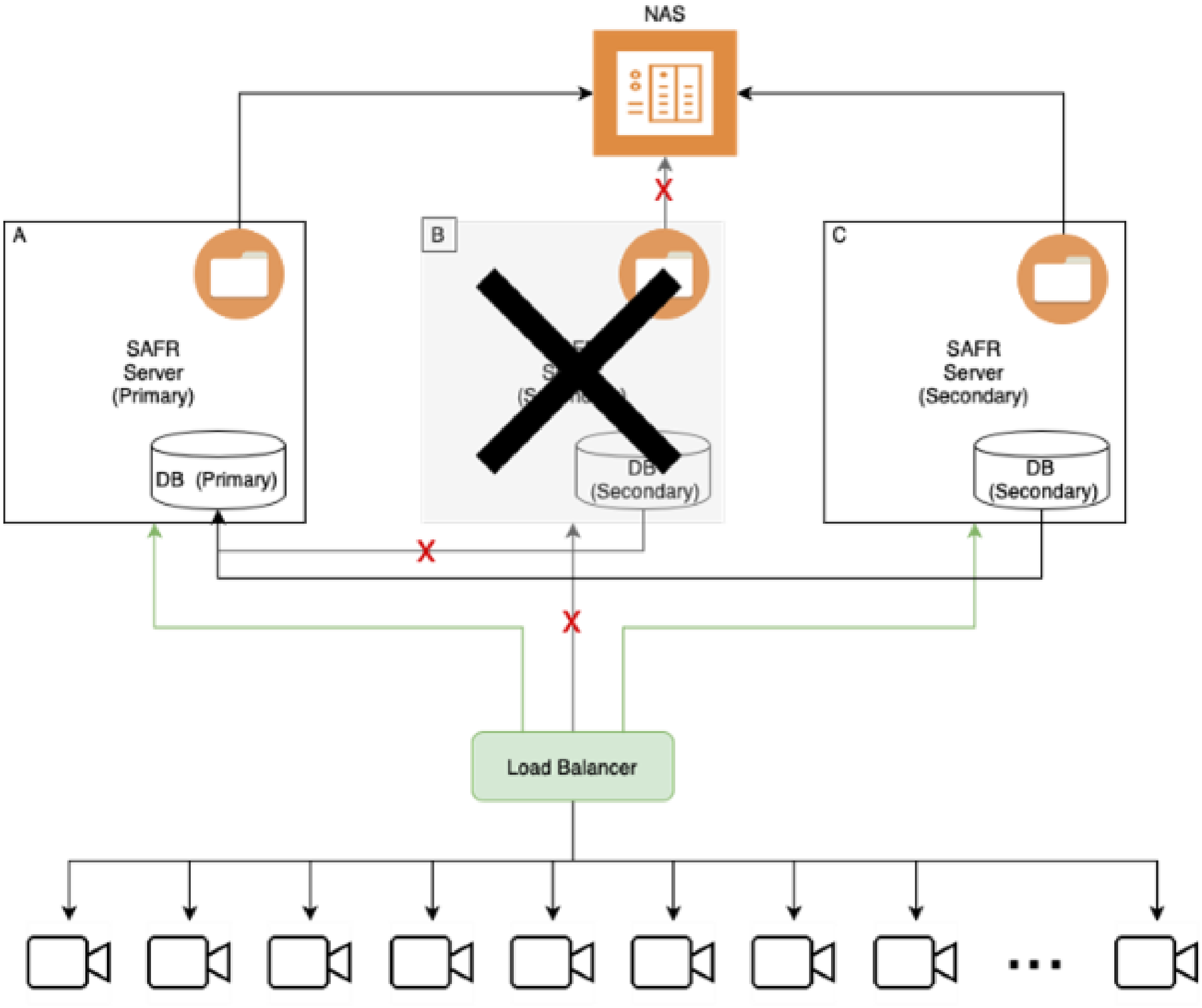
This section describes the functionality of a common large scale deployment configuration: an external load balanced system with 2 redundant secondary servers and a network-attached storage (NAS), a type of shared object storage.



When all the servers are working, this configuration has the following properties:

- Face recognition requests are distributed across all three servers.
- Server A's database is the primary database (i.e. performs database writes).
- All three server share in the database read load.
- All three servers read and write directly to the NAS.
- There isn't a single point of failure.
- Data is redundantly protected across all three servers.

27.3.1 Secondary Server Failure in an ELB Configuration



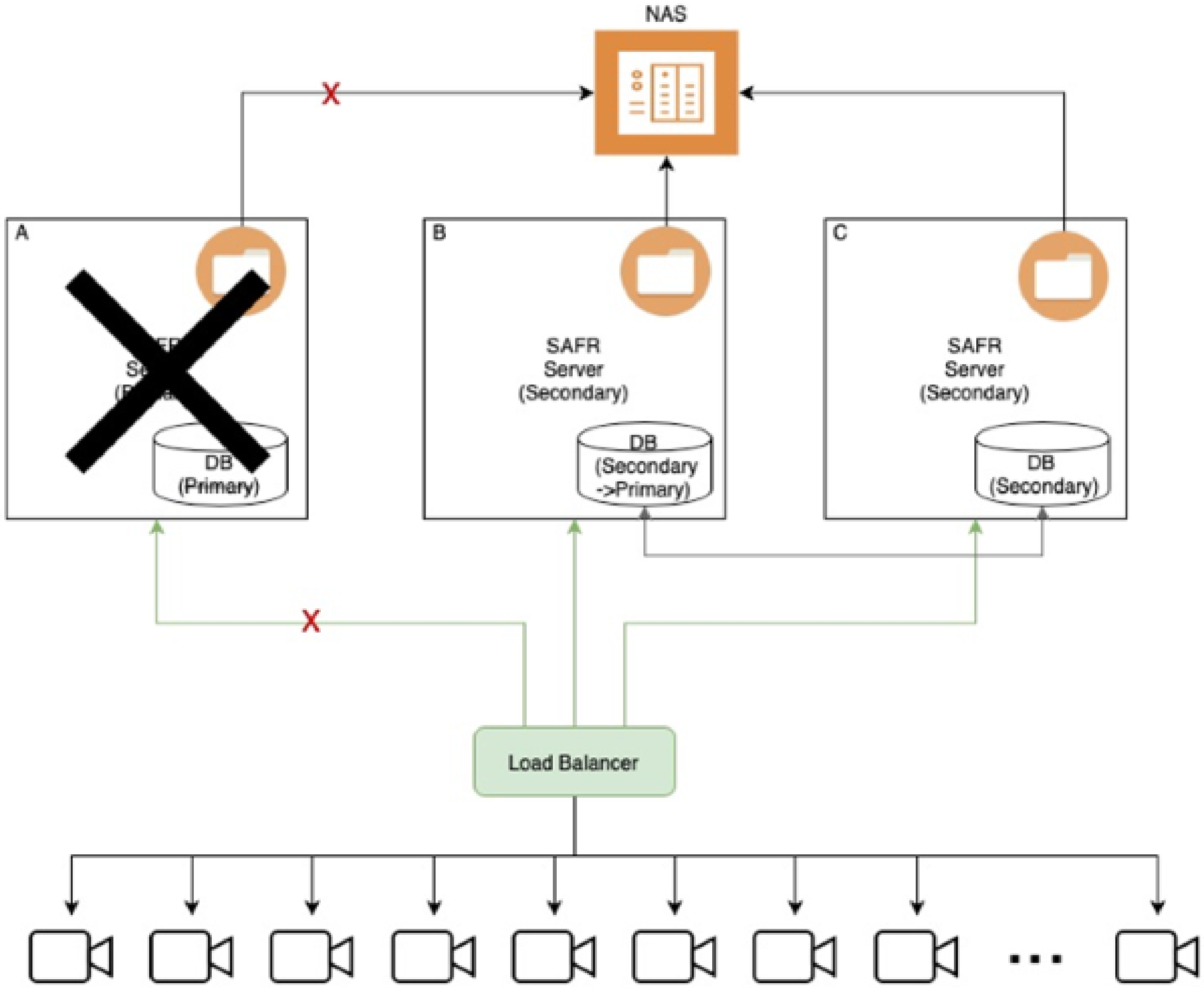
When one of the redundant secondary servers fail, the following occurs:

- Face recognition requests are handled by servers A (primary) and C (secondary).
- Server B is removed from rotation by the load balancer.
- Server B is removed from the database replica set.
- Server A's database continues operating as the primary database (i.e. it performs database writes).
- Servers A and C share in the database read load.
- Both remaining servers read and write directly to the NAS.

Impact:

- No service outage occurs, but longer latency may result.
- Data remains redundantly protected.
- There isn't any impact to object storage.

27.3.2 Primary Server Failure in an ELB Configuration



When the primary server fails, the following occurs:

- Face recognition requests are handled by servers B (secondary) and C (secondary).
- Server A is removed from rotation by the load balancer.
- Server A is removed from the database replica set.
- The server B database takes over as the primary database (i.e. it performs database writes).
- Servers B and C share in the database read load.
- Both remaining servers read/write directly to the NAS.

Impact:

- No service outage occurs, but longer latency may result.
- Data remains redundantly protected.
- There isn't any impact to object storage.

27.4 Migrate from Local to Shared Storage

If you start with local storage but later decide to move to shared storage, you will need to consolidate all of your objects to the new shared storage location, delete the local copies, and then mount the shared storage to the correct location. To do this, do the following:

1. Back up both the primary and redundant secondary servers to ensure you have a full backup of all SAFR content.
    - **On Linux**:
        - **Primary**: `python /opt/RealNetworks/SAFR/bin/backup.py`
        - **Redundant Secondaries**: `python /opt/RealNetworks/SAFR/bin/backup.py -o`
    - **On Windows**:
        - **Primary**: `python "C:\Program Files\RealNetworks\SAFR\bin\backup.py"`
        - **Redundant Secondaries**: `python "C:\Program Files\RealNetworks\SAFR\bin\backup.py" -o`
2. Stop all primary and redundant secondary servers by using the **stop** command. This can be done by doing the following on each server:
    - **On Linux**: `/opt/RealNetworks/SAFR/bin/stop`
    - **On Windows**: `"C:\Program Files\RealNetworks\SAFR\bin\stop.bat"`
3. Mount the new shared storage to a temporary location on primary and redundant secondary servers.
4. Copy all files from the primary server and every redundant secondary server(s) to the temporary location of the shared storage. from within the following paths:
    - **On Linux**: `/opt/RealNetworks/SAFR/cv-storage`
    - **On Windows**: `C:\ProgramData\RealNetworks\SAFR\cv-storage`
5. Delete or move the contents of the CV Storage folder on each primary and redundant secondary server as specified below.
    - **On Linux**: `/opt/RealNetworks/SAFR/cv-storage`
    - **On Windows**: `C:\ProgramData\RealNetworks\SAFR\cv-storage`
6. Unmount the temporary location of the new shared storage.
7. Mount the shared storage to the correct CV Storage location, or create a symlink to the shared storage location.
8. Start the primary and redundant secondary servers by using the **start** command. On each server, do the following:
    - **On Linux** `/opt/RealNetworks/SAFR/bin/start`
    - **On Windows** `"C:\Program Files\RealNetworks\SAFR\bin\start.bat"`
9. Disable any automatic backups on redundant secondary servers.
    - Now that you're using shared storage, only the primary server needs to be backed up. Disable any automatic backups you may have configured on your secondary servers.

27.5 Simple Secondary Server Behavior with Local Object Storage

On simple secondary servers, the Object Storage Service will operate in proxy mode.

Object Storage Servers operating in proxy mode will not attempt to use their own storage for objects, but will instead proxy the request to Object Storage Services that are running on either the primary server or on a redundant secondary server. If the redundant server it contacts doesn't have the object, the contacted redundant server will ask all other redundant servers for the object.

The list of servers that run the Object Storage Service is stored in the database and updated every minute. If a host does not respond within a timeout, it is de-prioritized.

27.6 Redundant Secondary Server Behavior with Local Object Storage

On both the primary server and on redundant secondary servers the Object Storage Service stores new objects in storage.

When a server receives a request for a file it does not find in its storage, it will request the object from other Object Storage Servers via HTTPS, and return the object if found. (The same applies for DELETEs.) This allows multiple Object Storage Servers to operate without using shared network storage, with each server saving a subset of the total objects, and relaying requests for other objects to its neighbors.

Even when using shared network storage, sometimes a request will come in for a new object before it is visible to all systems on the shared storage. The Object Storage Service will ask all the other Object Storage Servers for the object until it finds one that has the object.

27.7 Backup and Restore

The SAFR backup and restore process when using shared network storage is straightforward - you just need to back up the primary server. This will back up all configs, database content, and Object Service Storage objects.

When using local storage, however, the objects are distributed to multiple servers, so the backup must be run on the primary server as well as all redundant secondary servers.

The primary server should run a regular backup, while the redundant secondary servers run an '*objects only*' backup. The difference is just the addition of the "**-o**" flag to the backup script.

When restoring multiple backups, you can restore them all to the primary server, or you can restore the '*object only*' backups back to the same servers that they were backed up from.

27.7.1 Backup for Local Storage

- **On Linux**
  - **Primary**: `python /opt/RealNetworks/SAFR/bin/backup.py`
  - **Redundant Secondaries**: `python /opt/RealNetworks/SAFR/bin/backup.py -o`
- **On Windows**
  - **Primary**: `python "C:\Program Files\RealNetworks\SAFR\bin\backup.py"`
  - **Redundant Secondaries**: `python "C:\Program Files\RealNetworks\SAFR\bin\backup.py" -o`

27.7.2 Restore for Local Storage

- **On Linux**
  - **Primary**: `python /opt/RealNetworks/SAFR/bin/restore.py BACKUPFILENAME`
  - **Redundant Secondaries**: `python /opt/RealNetworks/SAFR/bin/restore.py -o BACKUPFILENAME`
- **On Windows**
  - **Primary**: `python "C:\Program Files\RealNetworks\SAFR\bin\restore.py" BACKUPFILENAME`
  - **Redundant Secondaries**: `python "C:\Program Files\RealNetworks\SAFR\bin\restore.py" -o BACKUPFILENAME`

27.8 Example Shared Storage Configurations

Below are two example shared storage configurations.

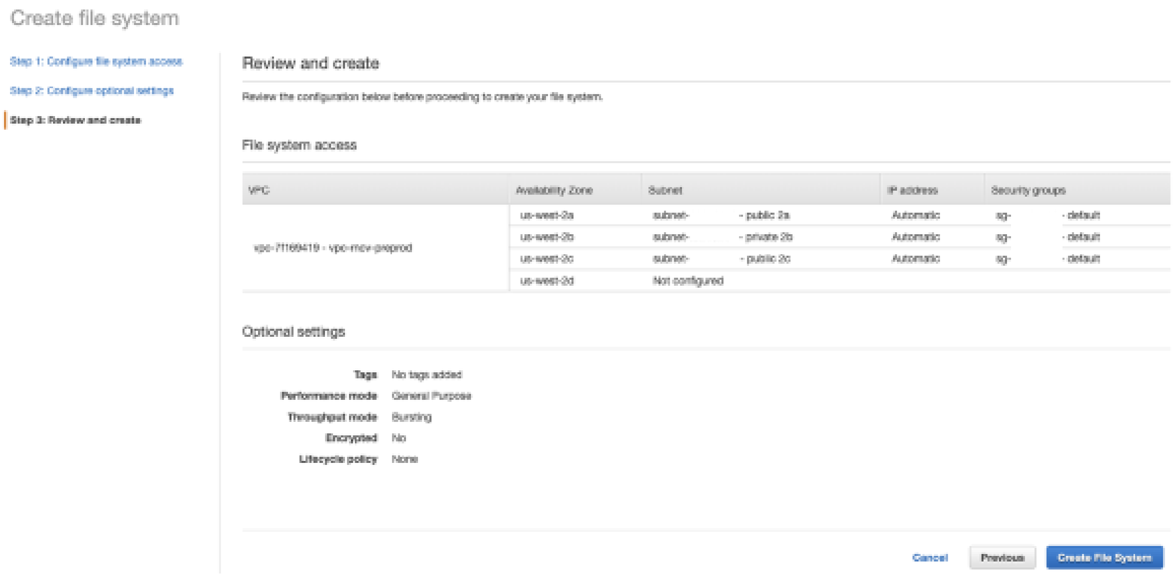
27.8.1 Linux

Shared storage on Linux is very straightforward. Simply mount your shared storage to the `/opt/RealNetworks/SAFR/cv-storage` location.

1. Stop SAFR.

   ```
   /opt/RealNetworks/SAFR/bin/stop
   ```

2. Create a shared storage location. The example below uses Amazon's Elastic File System (EFS).



Create file system

Step 1: Configure file system access
Step 2: Configure optional settings
Step 3: Review and create

Review and create

Review the configuration below before proceeding to create your file system.

File system access

VPC	Availability Zone	Subnet		IP address	Security groups	
vpc-71169419 - vpc-mcv-preprod	us-west-2a	subnet-	- public 2a	Automatic	sg-	- default
	us-west-2b	subnet-	- private 2b	Automatic	sg-	- default
	us-west-2c	subnet-	- public 2c	Automatic	sg-	- default
	us-west-2d	Not configured				

Optional settings

Tags	No tags added
Performance mode	General Purpose
Throughput mode	Bursting
Encrypted	No
Lifecycle policy	None

3. Edit `/etc/fstab` to create a mount point of `/opt/RealNetworks/SAFR/cv-storage` for your shared storage. The specific mount options should be provided by your specific storage service or device.

   ```
   fs-12345678.efs.us-west-2.amazonaws.com:/
       /opt/RealNetworks/SAFR/cv-storage nfs4
       nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600,retrans=2,_netdev
       0 0
   ```

4. Mount the remote share.

   ```
   sudo mount -a
   ```

5. Start SAFR.

   ```
   /opt/RealNetworks/SAFR/bin/start
   ```

27.8.2 Windows

Windows cannot mount a shared storage location directly to `C:\ProgramData\RealNetworks\SAFR\cv-storage`. It must instead create a symbolic link by doing the following:

1. Stop SAFR.

   ```
   "C:\Program Files\RealNetworks\SAFR\bin\stop.bat"
   ```

2. Create a shared storage location.

3. Delete the existing `C:\ProgramData\RealNetworks\SAFR\cv-storage` by running `rmdir /q /s C:\ProgramData\RealNetworks\SAFR\cv-storage` in an administrative command prompt. Deleting the existing cv-storage allows you to create a symbolic link from the *cv-storage* location to your shared storage location.
   Note: Be sure you either followed the migration steps above to consolidate your objects onto the new shared storage location, or that you're doing this on a new system without any data.

4. Create the symbolic link from `C:\ProgramData\RealNetworks\SAFR\cv-storage` to your shared storage location. To do this, run the appropriate command in an administrative command prompt:

   - If you're using a mapped network drive, run

     ```
     mklink /d C:\ProgramData\RealNetworks\SAFR\cv-storage Z:\
     ```

- If you're using an SMB share, run

  ```
  mklink /d C:\ProgramData\RealNetworks\SAFR\cv-storage \\servername\share
  ```

5. Start SAFR.

   ```
   "C:\Program Files\RealNetworks\SAFR\bin\start.bat"
   ```

28 SSL Certificate Installation

A properly installed secure sockets layer (SSL) certificate is critical to the secure operation of your SAFR Server. SAFR uses SSL certificates to establish secure network connections and data transfers. (i.e. https connections) SAFR requires https connections between SAFR Servers and between SAFR Servers and iOS Mobile Clients. None of the other SAFR components require https connections.

Before you can install an SSL certificate on your SAFR Server, you must first configure a Domain Name System (DNS) hostname for your server within your network domain, as described below.

28.1 DNS Hostnames

If you do not currently have a domain, you need to first obtain a domain name registered and configured with an accredited domain registrar.

28.1.1 How to Obtain a Domain Name

In order to set up a DNS, you need a domain within which you can register hostnames. ICANN maintains a list of accredited registrars from which to choose.

The following is a list of common registrars:

- GoDaddy
- Google Domains
- AWS
- HostGator

Follow the processes on these websites to find, purchase, and configure your domain name. Most registrars offer the ability to host your DNS for you and most also give you a web interface for managing it.

The following links lead to instructions on how to modify DNS entries:

- GoDaddy
- Google Domains
- AWS
- HostGator

After you have your domain, you can create a DNS hostname entry for your SAFR Server.

28.1.2 What a DNS Hostname Entry Does

DNS is a system that translates a hostname to a network IP address. For example, when a user types `www.example.com` into their browser, DNS servers resolve it to the IP address where the website is hosted.

To provide this translation, DNS requires an entry for each hostname. This entry typically takes the form of an *A record* (the A stands for “Address”) which defines the hostname to IP address translation in DNS. An *A record* is the most basic type of syntax used in DNS records.

The following is an example of an *A record*:

```
safr.example.com    A   12.34.56.78
```

28.1.3 Set Up a DNS Hostname Entry for your Primary Server

DNS can be managed in numerous ways. This might be a text file or a web interface for configuring the DNS entries. If you are not sure, contact the person managing the domain name for your network.

28.1.4 What Type of IP Address Should You Use?

You should use a static IP address. If you instead choose to use DHCP to get a dynamically assigned IP address, and your IP address happens to change, your DNS hostname entry will stop working until you update the entry.

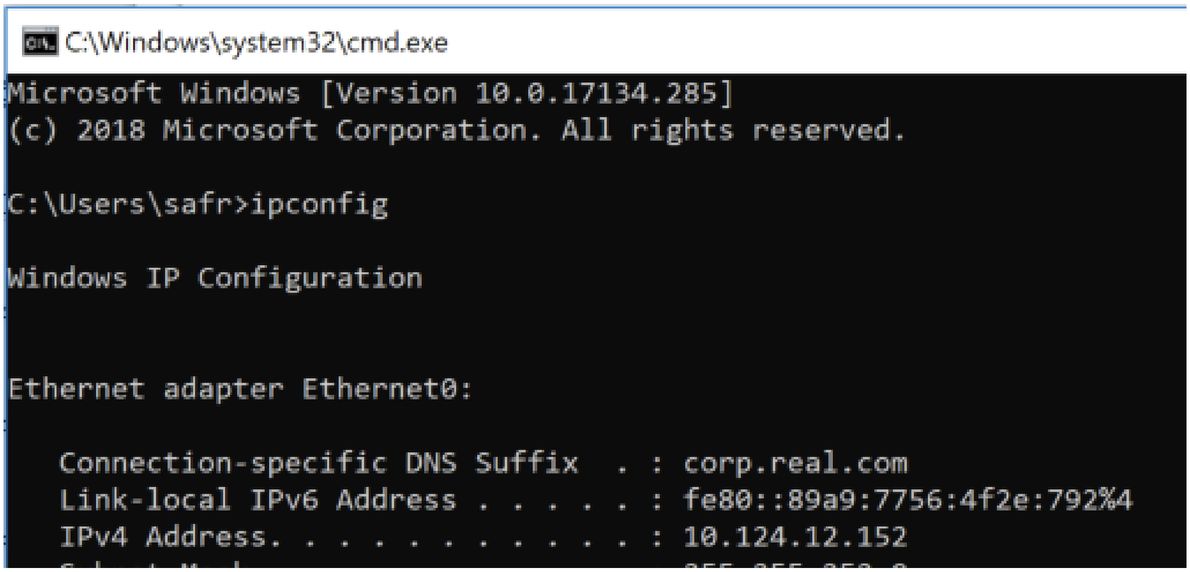
28.1.4.1 Configure a Static IP

1. Obtain a static IP from your network administrator. The information should include the following:
   - Static IP address
   - Subnet mask
   - Default gateway
2. Configure your system as described below:
   - For Windows, see https://www.howtogeek.com/howto/19249/how-to-assign-a-static-ip-address-in-xp-vista-or-windows-7/.
   - For macOS, see https://www.howtogeek.com/howto/22161/how-to-set-up-a-static-ip-in-mac-os-x/.

The IP address should be the internal IP address of the computer running the SAFR Server. This should not be your public IP address because the public IP address usually points at your router, modem, or similar device. The internal IP address is the IP used locally by the computer. It can be determined by doing the following:

For Windows 10

1. Open a command prompt (cmd.exe).
2. Run ipconfig.
3. The IP address is listed as the IPv4 Address.

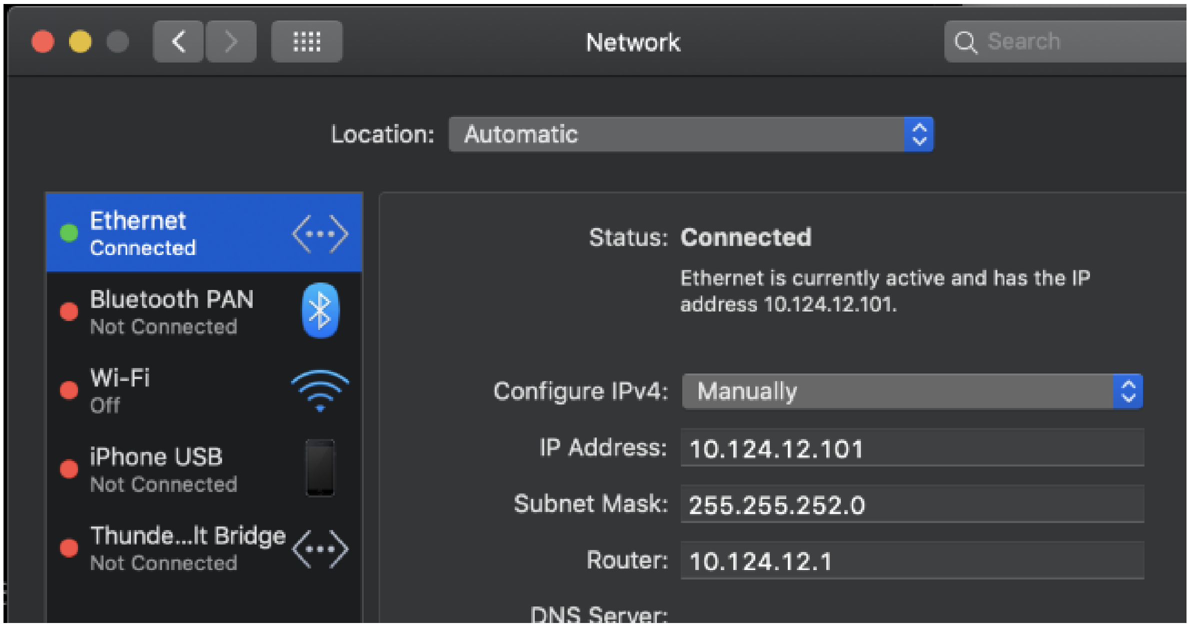


C:\Windows\system32\cmd.exe

```
Microsoft Windows [Version 10.0.17134.285]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\safr>ipconfig

Windows IP Configuration


Ethernet adapter Ethernet0:

   Connection-specific DNS Suffix  . : corp.real.com
   Link-local IPv6 Address . . . . . : fe80::89a9:7756:4f2e:792%4
   IPv4 Address. . . . . . . . . . . : 10.124.12.152
```

For macOS

1. Open **System Preferences**.
2. Open **Network**.
3. Click the active network connection (usually WiFi or Ethernet).
4. The IP address is displayed in the dialog.



Location: Automatic

Ethernet Connected

Bluetooth PAN Not Connected

Wi-Fi Off

iPhone USB Not Connected

Thunde...lt Bridge Not Connected

Status: Connected

Ethernet is currently active and has the IP address 10.124.12.101.

Configure IPv4: Manually

IP Address: 10.124.12.101

Subnet Mask: 255.255.252.0

Router: 10.124.12.1

DNS Server:

28.2 SSL Certificates

After you have configured a DNS hostname for your primary server, you can now install an SSL certificate.

28.2.1 What an SSL Certificate Does

SSL certificates are small data files that digitally bind a cryptographic key to an organization's information. When installed on a server, an SSL certificate allows secure connections from the server to a browser or other program and protects sensitive information.

A common use for SSL certificates is to enable a web server to provide a secure connection with a web browser (i.e. an https:// connection instead of an http:// connection).

28.2.2 Obtain an SSL Certificate

SSL certificates need to be issued from either a trusted certificate authority or from an accredited domain registrar.

Browsers, operating systems, and mobile devices maintain lists of trusted certificate authority root certificates, which must be present on a computer for it to trust the certificate.

The following is a list of popular certificate authorities from which you can obtain an SSL certificate:

- Comodo
- IdenTrust
- GoDaddy
- GlobalSign
- Digicert
- Certum
- Entrust

Go to ICANN for a complete list of accredited domain registrars.

Because SAFR uses Apache as its web server, request SSL certificate files for Apache web server. You will receive the following three files SAFR uses to configure the Apache web server:

- **Key**: This is your key file and should not be shared publicly.
- **Certificate**: The SSL certificate for your domain.

- **Ca__bundle**: Signer root/intermediate certificate. This file is optional; it's not always provided by the SSL certificate provider.

Note: Self-signed certificates do not work.

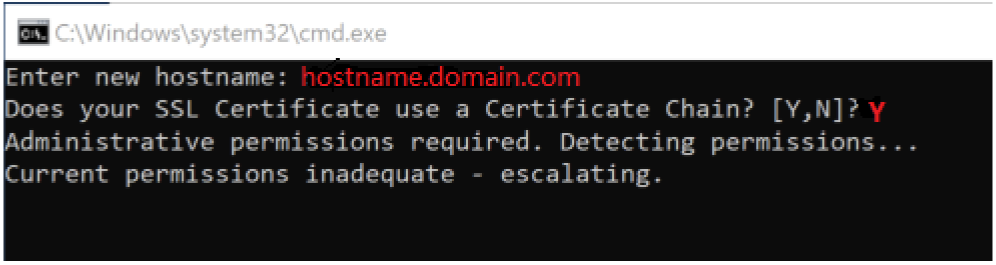
28.2.3 Provision SSL Certificates for your Primary Server

Do the following to configure Apache to serve the request over HTTPS:

1. Log in to your primary server.
2. It is recommended that you make a backup of the default SSL files and save them in case you need to perform a rollback to the earlier version.
   - On macOS, back up the following files:
     - `/etc/apache2/ssl/SAFR.key`
     - `/etc/apache2/ssl/SAFR.crt`
   - On Windows, back up the following files:
     - `C:\Program Files\RealNetworks\SAFR\httpd\conf\ssl\SAFR.key`
     - `C:\Program Files\RealNetworks\SAFR\httpd\conf\ssl\SAFR.crt`
   - On Linux, back up the following files:
     - `/opt/RealNetworks/SAFR/httpd/ssl/SAFR.key`
     - `/opt/RealNetworks/SAFR/httpd/ssl/SAFR.crt`
3. Upload the certificate-related files to the SSL certificate folder:
   - SSLCertificateFile – Certificate CRT
   - SSLCertificateKeyFile – `Private.a` key file
4. Change the names of the following files:
   - Rename *_certificate.crt to SAFR.crt
   - Rename *_private.key to SAFR.key
5. If your certificate authority provided an intermediate certificate chain, do the following:
   1. Save your SSL intermediate certificate chain file to the following location:
      - **On macOS**:
        - `/etc/apache2/ssl/SAFR-ca.crt`
      - **On Windows**:
        - `C:\Program Files\RealNetworks\SAFR\httpd\conf\ssl\SAFR-ca.crt`
      - **On Linux**:
        - `/opt/RealNetworks/SAFR/httpd/ssl/SAFR-ca.crt`
   2. Check the `SAFR-ssl-cert.inc` file to connect your SSL certificate to the certificate chain.
      - **On macOS**:
        - `/etc/apache2/other/SAFR-ssl-cert.inc`
        - `#Define ssl_certificate_chain_file "/private/etc/apache2/ssl/SAFR-ca.crt"`
      - **On Windows**:
        - `C:\Program Files\RealNetworks\SAFR\httpd\conf\ssl\SAFR-ca.crt`
        - `#Define ssl_certificate_chain_file "conf/ssl/SAFR-ca.crt"`
      - **On Linux**:
        - `/opt/RealNetworks/SAFR/httpd/ssl/SAFR-ca.crt`
        - `#Define ssl_certificate_chain_file "/opt/RealNetworks/SAFR/httpd/ssl/SAFR-ca.crt"`
      - Certificate file mappings

Certificate file	Certificate file in SAFR
*.domainname.key	SAFR.key
.domainname_chain.crt	SAFR-ca.crt
.domainname_public.crt	SAFR.crt

6. Run the SAFR reconfigure script, as described below.
   - **On macOS**:
     - Open **Applications > Utilities > Terminal** to open a Terminal window.
     - Run the following command after replacing hostname.domain.com with your hostname and domain:
       - `/Library/RealNetworks/SAFR/bin/reconfigure hostname.domain.com`
   - **On Windows**:
     - Enter this command: `"C:\Program Files\RealNetworks\SAFR\bin\reconfigure.bat"`
     - Enter the hostname and click **Yes** when prompted if your SSL certificate uses a certificate chain.
     - Click **Yes** when prompted by *User Account Control*.
   - **On Linux**:
     - Open a Terminal window. Run the following command after replacing hostname.domain.com with your hostname and domain:
       - `/opt/RealNetworks/SAFR/bin/reconfigure hostname.domain.com`
     - Click **Yes** when prompted by *User Account Control*.



```
Enter new hostname: hostname.domain.com
Does your SSL Certificate use a Certificate Chain? [Y,N]? Y
Administrative permissions required. Detecting permissions...
Current permissions inadequate - escalating.
```

7. Verify that your services are running and your SSL certificate is properly installed by opening a browser and opening `https://hostname.domain.com:8085/health`. (Replace hostname.domain.com with your hostname and domain.)

   You should receive the following message:

   ```
   { "status" : "up" }
   ```

28.3 Troubleshoot

Database Service Down

Problem: You receive an error report saying Database (MongoDB) Service Down when you run the **check** command after you install SSL.

Solution: The cause may be that the DNS hostname IP is different from the IP when you installed SAFR without SSL installed.

Use the following workaround:

Add the following line to your primary server /etc/hosts file:

```
127.0.0.1 hostname.domain.com
```

29 On-Premise Licensing

SAFR systems require a license to operate.

29.1 License Limit Metrics

SAFR licenses limit usage according to the following metrics:

- **Expiration date**: The date when the SAFR license expires. After this date, SAFR software discontinues operation.
- **Max Feeds per Hour**: Maximum number of video feeds that can be used at one time by the SAFR system. If you attempt to connect more video feeds than your license allows, the excess video feed connection attempts will all fail. Existing video feeds must be disconnected for a period of 1 hour before new video feeds are allowed to re-use the license.
  Note: If a single camera is providing video feeds to 2 different Desktop Client instances, that counts as 2 video feeds for licensing purposes.
- **Max Faces**: Maximum number of people that can be registered with the SAFR system's Person Directory. Attempting to add people above this limit results in an error.
- **Max Days Between Reports**: The maximum elapsed time that can pass before the SAFR system must report its status to a SAFR License Server. To communicate with the SAFR License Server, your SAFR Server must be able to make connections to cv-instam.real on port 443. Your SAFR Server will discontinue operation if it's unable to reach the SAFR License Server after the specified time has elapsed. If you need to operate your SAFR system on a private network that isn't connected to the Internet, contact your SAFR account manager to acquire a special offline license.
  Note: This metric is only applicable for on-premise deployments.

License limit metrics for your SAFR license can be found on the Status page of the Web Console.

29.2 Licensing for On-Premise Deployments

In on-premise deployments, SAFR licenses are attached to your SAFR system's primary server. The following describes how the SAFR license is managed:

- License Acquisition - Your SAFR Server attempts to acquire a license from the SAFR license server when it's first run. If your SAFR system doesn't have Internet connectivity, see the Offline Licensing section below to see how to obtain a SAFR license.
- Licenses are bound to the primary SAFR Server. If you install one or more secondary servers for the purpose of load balancing or redundancy, the secondary servers acquire their licenses through the primary server.
- If you want to move your primary server to a machine with a different IP address, you must wait 24 hours between uninstalling the server and reinstalling it on the new machine. If you try to reinstall the SAFR Server before 24 hours has elapsed, you will get an unauthorized access error when the SAFR Server unsuccessfully attempts to get a valid license from the SAFR License Server. After 24 hours has elapsed, however, a reinstalled SAFR Server will automatically (and successfully) reacquire a SAFR license.
  Note: The 24 hour wait time can be avoided if you contact your SAFR Account Manager and ask them to manually reset your IP address.
  - Note that the previous behavior only applies to SAFR servers that are **uninstalled**. If, on the other hand, the IP address of your SAFR Server changes or changes to a hostname while the server remains installed, there is no problem; your server simply informs the SAFR License Server of its new IP address or hostname the next time it checks in with the SAFR License Server.

29.3 Offline Licensing

If your SAFR system doesn't have Internet connectivity, do the following to get a SAFR license:

1. Obtain a license request file for the machine on which SAFR Platform is installed.

   1. On the machine that has SAFR Platform installed, install Python 3.X. The Python installer can be downloaded from here: https://www.python.org/downloads/.
   2. On the machine that has SAFR Platform installed, run *get-license-request.py*.

      On Windows:

      1. Open *Command Prompt (Admin)* or *Windows PowerShell (Admin)* by right clicking on the Windows Start menu (located in the bottom left corner of the screen) and selecting the appropriate entry.
      2. Navigate to the folder containing *get-license-request.py*. (usually `C:\Program Files\RealNetworks\SAFR\bin\`)
      3. Run `python get-license-request.py`

      On Linux:

      1. Open Terminal.
      2. Run `sudo python /opt/RealNetworks/SAFR/bin/get-license-request.py`

      On macOS:

      1. Open Terminal.
      2. Run `python /Library/RealNetworks/SAFR/bin/get-license-request.py`
   3. When prompted, enter the SAFR account name and password.
   4. The script will attempt to read *safrports.conf* to communicate with CoVi. If *safrports.conf* can't be found, then the script will use the default port, 8080.
   5. Running the script generates a file called *safr_license_request.json* in the same working directory as the script. Make sure to run the script in a directory that you have write access to.

```
usage: get-license-request.py [-h] [-n HOSTNAME] [-p PORT] [-q] [-v]

Generates a license request specific to your SAFR installation.

optional arguments:
  -h, --help            show this help message and exit
  -n HOSTNAME, --hostname HOSTNAME
                        Host name to your SAFR installation.
  -p PORT, --port PORT  Port to your SAFR installation.
  -q, --quiet           Suppress output.
  -v, --verbose         Enable DEBUG logging.
```

2. Retrieve the license by sending the license request to SAFR Cloud.
   1. Copy the newly generated *safr_license_request.json* file and the *get-license.py* script to the same folder on a machine that has Internet access and has Python 3.X installed. *get-license.py* can be found here:
      - Windows: `C:\Program Files\RealNetworks\SAFR\bin\get-license.py`
      - Linux: `/opt/RealNetworks/SAFR/bin/get-license.py`
      - macOS: `/Library/RealNetworks/SAFR/bin/get-license.py`
   2. Run the *get-license.py* script.

      On Windows:

      1. Open *Command Prompt (Admin)* or *Windows PowerShell (Admin)* by right clicking on the Windows Start menu (located in the bottom left corner of the screen) and selecting the appropriate entry.
      2. Navigate to the folder containing *get-license.py*.
      3. Run `python get-license.py`

   On Linux:

      1. Open Terminal.
      2. Navigate to the folder containing *get-license.py*.
      3. Run `sudo python get-license.py`

   On macOS:

      1. Open Terminal.
      2. Navigate to the folder containing *get-license.py*.
      3. Run `python get-license.py`

   Note: On a macOS machine you might receive an error message about being unable to load SSL root certificates. This is an issue with the way Python handles SSL certificates on Macs. More information about the issue can be found here. A workaround can be found here.

   3. When prompted, enter the SAFR account name and password.

   4. This will generate a file called *safr_license.json* in the current working directory. Be sure to execute in a directory that your user account has write access to.

```
usage: get-license.py [-h] [-p PATH] [-e ENV] [-q] [-v]

Gets a license from SAFR licensing servers.

optional arguments:
  -h, --help            show this help message and exit
  -p PATH, --path PATH  Path to license request file.
  -e ENV, --env ENV     License server environment to communicate with.
  -q, --quiet           Suppress output.
  -v, --verbose         Enable DEBUG logging.
```

3. Install the retrieved license onto your installed SAFR Server.

   1. Run *insert-license.py* to install the license onto your primary SAFR Server.

   On Windows:

      1. Open *Command Prompt (Admin)* or *Windows PowerShell (Admin)* by right clicking on the Windows Start menu (located in the bottom left corner of the screen) and selecting the appropriate entry.
      2. Navigate to `C:\Program Files\RealNetworks\SAFR\bin\`
      3. Copy *safr_license.json* into this folder.
      4. Run `python insert-license.py`

   On Linux:

      1. Open Terminal.
      2. Navigate to `/opt/RealNetworks/SAFR/bin/`
      3. Copy *safr_license.json* into this folder.

    4. Run `sudo python insert-license.py`

   On macOS:

    1. Open Terminal.
    2. Navigate to `/Library/RealNetworks/SAFR/bin/`
    3. Copy *safr_license.json* into this folder.
    4. Run `python insert-license.py`

2. When prompted, enter the SAFR account name and password.

3. The script will attempt to read *safrports.conf* to communicate with CoVi. If *safrports.conf* can't be found, then the script will use the default port, 8080.

```
usage: insert-license.py [-h] [-n HOSTNAME] [-p PORT] [-f FILE] [-q] [-v]

Inserts a license specific to your SAFR installation.

optional arguments:
  -h, --help            show this help message and exit
  -n HOSTNAME, --hostname HOSTNAME
                        Host name to your SAFR installation.
  -p PORT, --port PORT  Port to your SAFR installation.
  -f FILE, --file FILE  Path to license file.
  -q, --quiet           Suppress output.
  -v, --verbose         Enable DEBUG logging.
```

30 SAFR Support Scripts

The SAFR Platform installation includes several scripts to manage and monitor your server. They are located in the bin folder under the SAFR Platform installation location.

- On macOS: `/Library/RealNetworks/SAFR/bin`
- On Linux or Jetson: `/opt/RealNetworks/SAFR/bin`
- On Windows: `C:\Program Files\RealNetworks\SAFR\bin`

Note: Some of the scripts below may not work if you're accessing the SAFR Platform through the NVIDIA Metropolis Application Framework (MAF).

30.1 check Script

The **check** script checks the status of SAFR Server services.

- On macOS, run `/Library/RealNetworks/SAFR/bin/check`
- On Linux or Jetson, run `/opt/RealNetworks/SAFR/bin/check`
- On Windows, run `"C:\Program Files\RealNetworks\SAFR\bin\check.bat"`

30.2 configure-ports Script

The **configure-ports** script customizes the ports SAFR services listen on. This is typically done only if there is a conflict with existing software on the same server.

If port conflicts are detected during SAFR Platform installation, the following occurs:

1. The ports in conflict are reported.
2. Notepad is launched to edit safrports.conf
3. The SAFR Platform installer is automatically relaunched after new non-conflicting ports are chosen.

This script is executed as part of the installation when appropriate, so it doesn't need to be executed manually unless you are changing the port settings after installation.

This script takes no arguments but relies on the safrports.conf file to determine what ports are to be used.

safrports.conf is located at the following locations:

- On macOS: `/Library/RealNetworks/SAFR/safrports.conf`
- On Linux or Jetson: `/opt/RealNetworks/SAFR/safrports.conf`
- On Windows: `C:\Program Files\RealNetworks\SAFR\safrports.conf`

30.3 reconfigure Script

The **reconfigure** script configures the hostname used by the SAFR Server. Run this command when configuring the server to use a DNS hostname with an SSL certificate.

This script can be run with arguments specifying the hostname and whether an SSL certificate chain is used by your SSL certificate. If no arguments are passed, you will be prompted for those values.

This script requires administrator privileges. It automatically asks for admin privileges on Windows and requires `sudo` on macOS and Linux.

- On macOS, run `/Library/RealNetworks/SAFR/bin/reconfigure <HOSTNAME> <SSL CERTIFICATE CHAIN?>`
- On Linux or Jetson, run `/opt/RealNetworks/SAFR/bin/reconfigure <HOSTNAME> <SSL CERTIFICATE CHAIN?>`
- On Windows, run `"C:\Program Files\RealNetworks\SAFR\bin\reconfigure.bat" <HOSTNAME> <SSL CERTIFICATE CHAIN?>`

Examples:

macOS:

- `/Library/RealNetworks/SAFR/bin/reconfigure 192.168.123.124 y`

Linux:

- `/opt/RealNetworks/SAFR/bin/reconfigure 192.168.123.124 n`

Jetson:

- `/opt/RealNetworks/SAFR/bin/reconfigure 192.168.123.124 n`

Windows:

- `"C:\Program Files\RealNetworks\SAFR\bin\reconfigure.bat" 192.168.123.124 y`

30.4 start Script

The **start** script starts all SAFR Server services on the current machine.

- On MacOS, run `/Library/RealNetworks/SAFR/bin/start`
- On Linux or Jetson, run `/opt/RealNetworks/SAFR/bin/start`
- On Windows, run `"C:\Program Files\RealNetworks\SAFR\bin\start.bat"`

30.5 stop Script

The **stop** script stops all SAFR Server services on the current machine.

- On macOS, run `/Library/RealNetworks/SAFR/bin/stop`
- On Linux or Jetson, run `/opt/RealNetworks/SAFR/bin/stop`
- On Windows, run `"C:\Program Files\RealNetworks\SAFR\bin\stop.bat"`

30.6 syscollect Script

The **syscollect** script collects all the necessary logs, stats, and configuration files into a single archive file that can be easily emailed to SAFR sales support engineers.

- On Linux or Jetson, run `python /opt/RealNetworks/SAFR/bin/syscollect.py`. The archive file will be generated at `/opt/RealNetworks/SAFR/syscollect/`.
- On Windows, run `python "C:\Program Files\RealNetworks\SAFR\bin\syscollect.py"`. The archive file will be generated at `C:\Program Files\RealNetworks\SAFR\syscollect\`.

The script accepts the following optional arguments:

- **-h**, **–help**: Lists all the optional arguments available for this script.
- **-p <PATH>**, **–path <PATH>**: Changes where the archive file is generated.
- **-q**, **–quiet**: Runs the script in quiet mode; command line output is suppressed.
- **-v**, **–verbose**: Enables verbose command line output for debugging purposes.

30.7 uninstaller.exe

The **uninstaller** executable removes the SAFR Platform entirely. This closes all SAFR applications, stops all SAFR services, and then removes all SAFR services and data.

On Windows, you must select the optional ProgramData component to remove the config files, logs, and database files.

- On macOS, run `/Library/RealNetworks/SAFR/uninstaller`
- On Linux or Jetson, run `/opt/RealNetworks/SAFR/uninstaller`
- On Windows, run `"C:\Program Files\RealNetworks\SAFR\uninstaller.exe"`

31 SAFR Server Backup and Restore

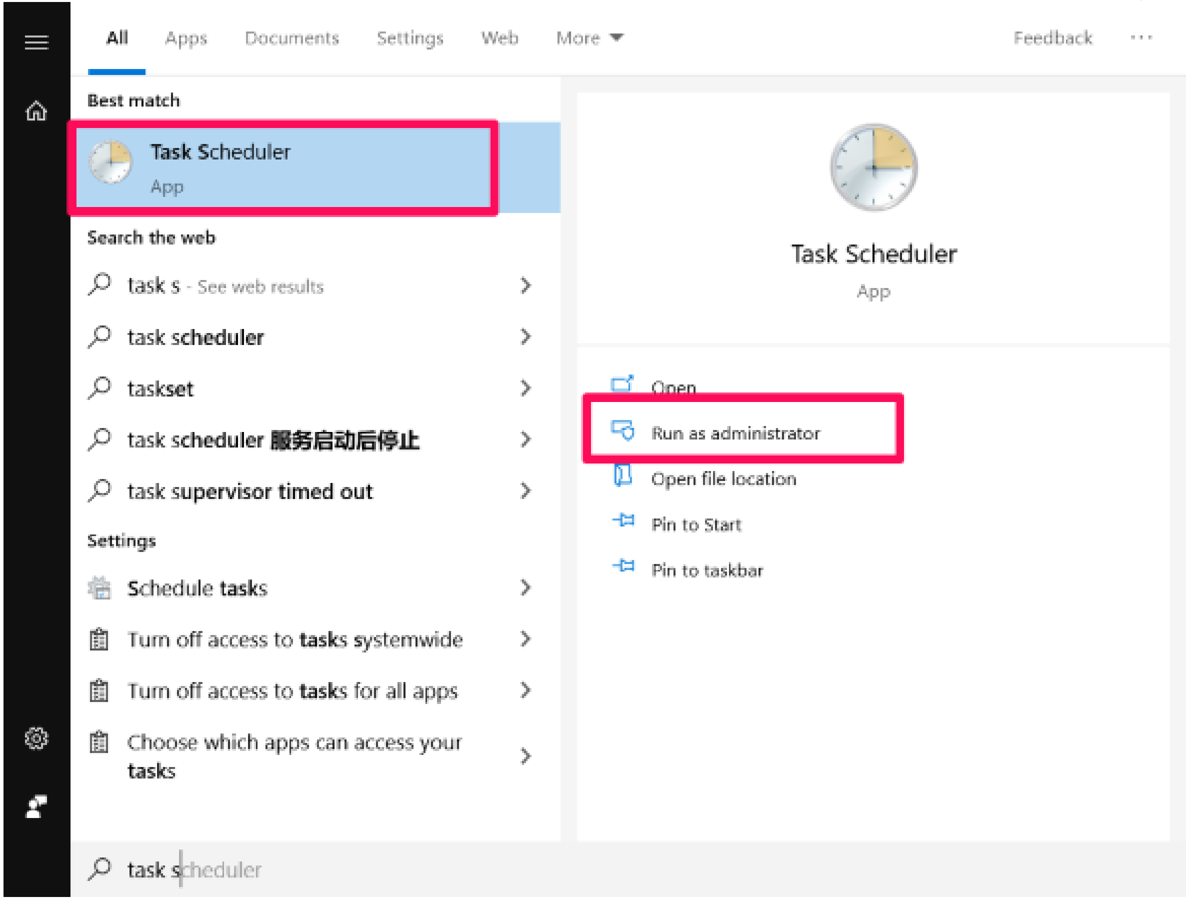
The backup process backs up the entire SAFR Server, including the various databases, configuration files, images, and objects, to a single backup file at a location of your choosing. The restore process restores all the SAFR Server data to any computer that meets the minimum system requirements. Note that the target computer of the restore process doesn't need to have the same IP address as the original computer.

31.1 On Windows

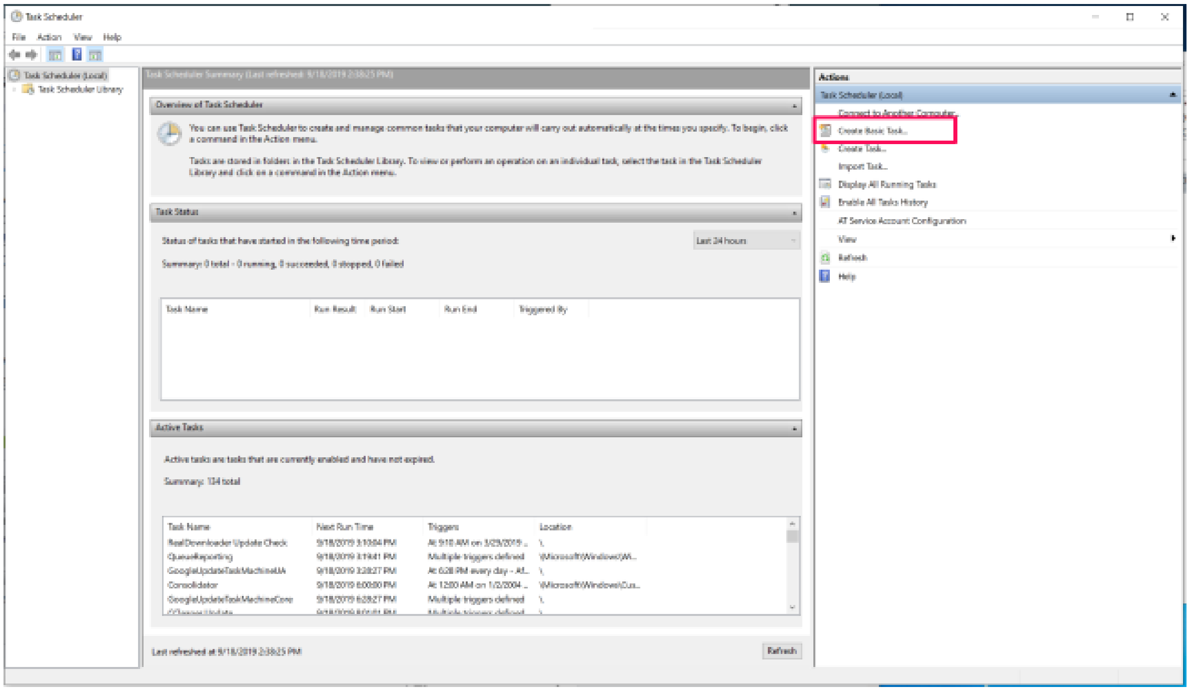
31.1.1 Windows Backup

To use the *Windows Task Scheduler* to create a daily database backup, do the following:

1. Run the *Windows Task Scheduler* as an administrator. The *Task Scheduler* window will be displayed.



2. In the **Actions** pane, click **Create Basic Task**.



Task Scheduler Summary (Last refreshed: 9/18/2019 2:38:25 PM)

Overview of Task Scheduler

You can use Task Scheduler to create and manage common tasks that your computer will carry out automatically at the times you specify. To begin, click a command in the Action menu.

Tasks are stored in folders in the Task Scheduler Library. To view or perform an operation on an individual task, select the task in the Task Scheduler Library and click on a command in the Action menu.

Task Status

Status of tasks that have started in the following time period: Last 24 hours

Summary: 0 total - 0 running, 0 succeeded, 0 stopped, 0 failed

Task Name	Run Result	Run Start	Run End	Triggered By

Active Tasks

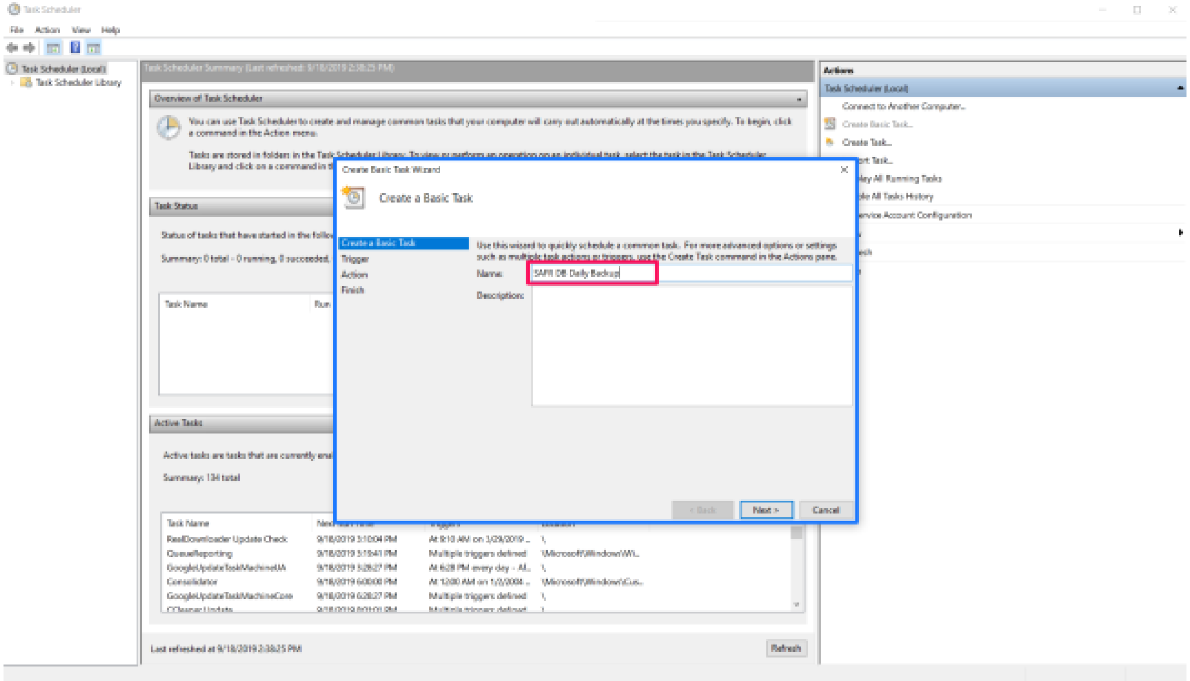
Active tasks are tasks that are currently enabled and have not expired.

Summary: 134 total

Task Name	Next Run Time	Triggers	Location
RealDownloader Update Check	9/18/2019 3:10:04 PM	At 9:10 AM on 3/29/2019 ...	\
QueueReporting	9/18/2019 3:19:41 PM	Multiple triggers defined	\Microsoft\Windows\W...
GoogleUpdateTaskMachineUA	9/18/2019 3:28:27 PM	At 6:28 PM every day - Af...	\
Consolidator	9/18/2019 6:00:00 PM	At 12:00 AM on 1/2/2004 ...	\Microsoft\Windows\Cus...
GoogleUpdateTaskMachineCore	9/18/2019 6:28:27 PM	Multiple triggers defined	\

Last refreshed at 9/18/2019 2:38:25 PM

3. In the wizard’s **Name** field, provide a name for the task. (e.g. *SAFR DB Daily Backup*) Click **Next**.

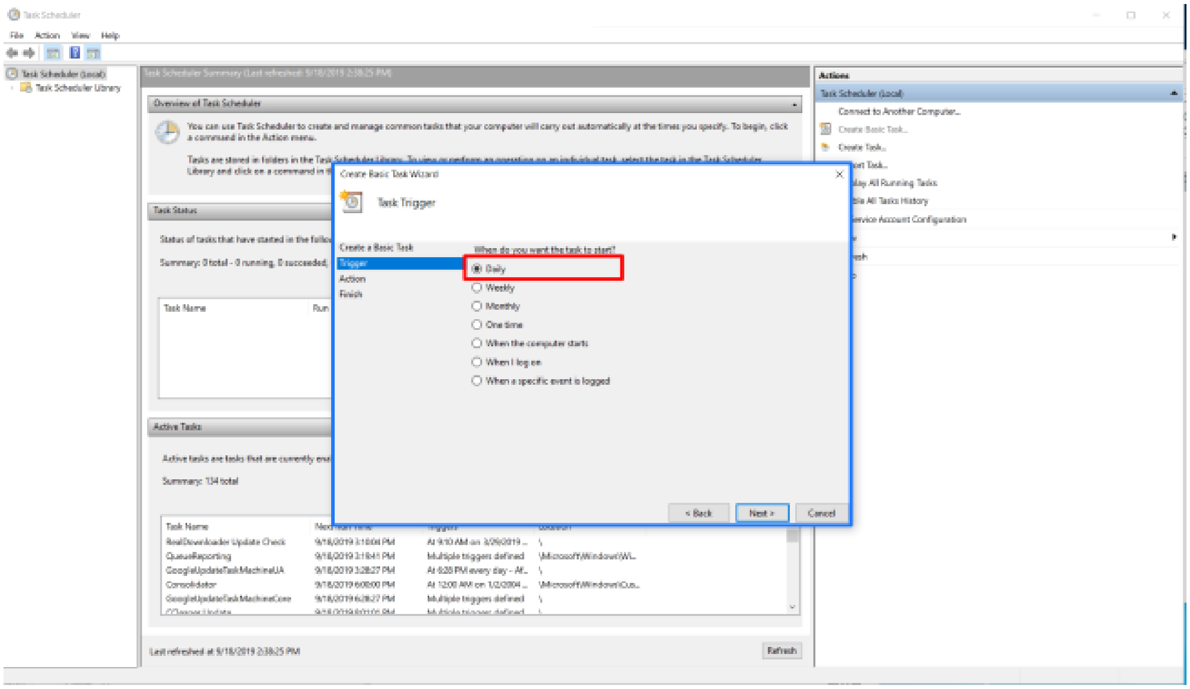


Use this wizard to quickly schedule a common task. For more advanced options or settings such as multiple task actions or triggers, use the Create Task command in the Actions pane.

Name: SAFR DB Daily Backup

Description:

4. Under **Task Trigger**, select the **Daily** option, and click **Next**.

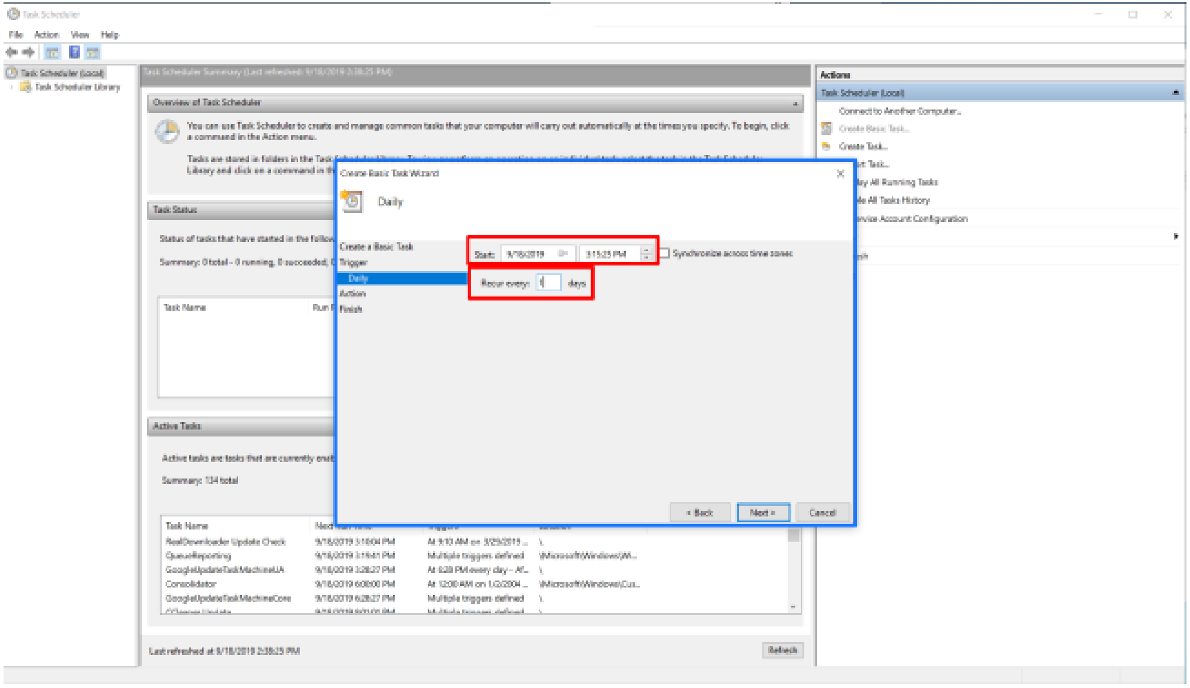

Create Basic Task Wizard

Task Trigger

When do you want the task to start?

● Daily
○ Weekly
○ Monthly
○ One time
○ When the computer starts
○ When I log on
○ When a specific event is logged

5. Set the date and time for when you want the task to run, and enter 1 in the **Recur** field. Click **Next**.


Create Basic Task Wizard

Daily

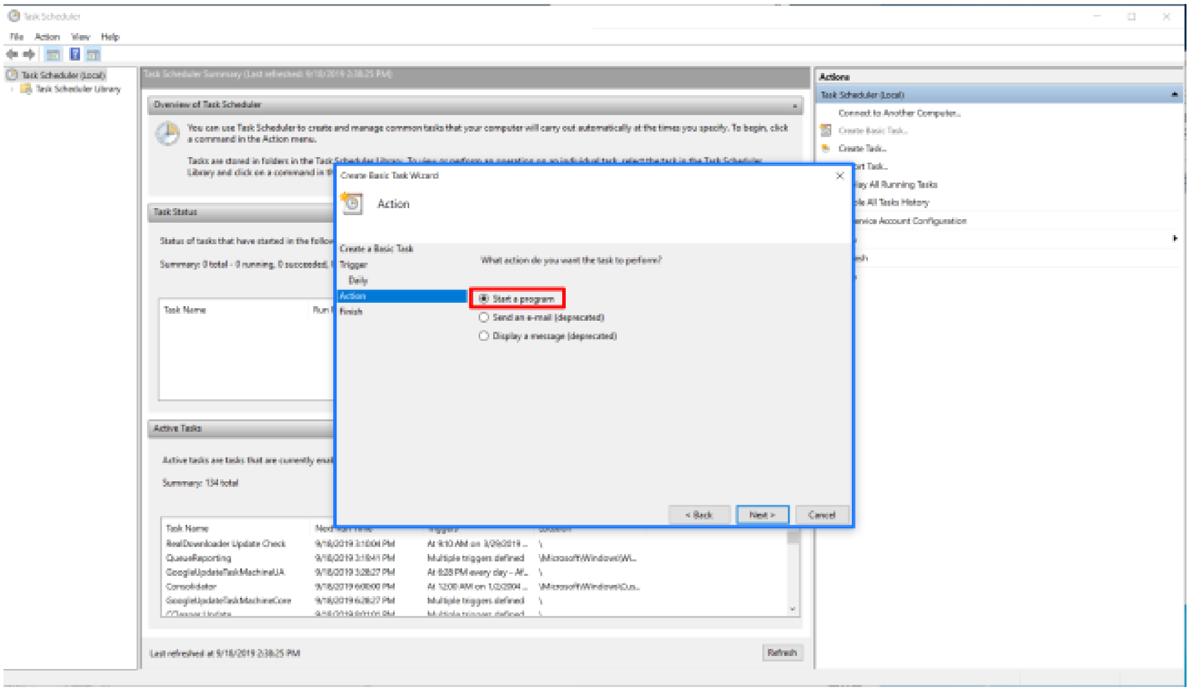
Start: 9/18/2019 3:15:25 PM □ Synchronize across time zones

Recur every: 1 days

6. Using a text editor, create a .bat file called *SAFR DB Daily Backup*. Edit the .bat file, add the following commands, and then save the file.

```
@echo off
cd C:\Program Files\RealNetworks\SAFR\bin
start python backup.py
```

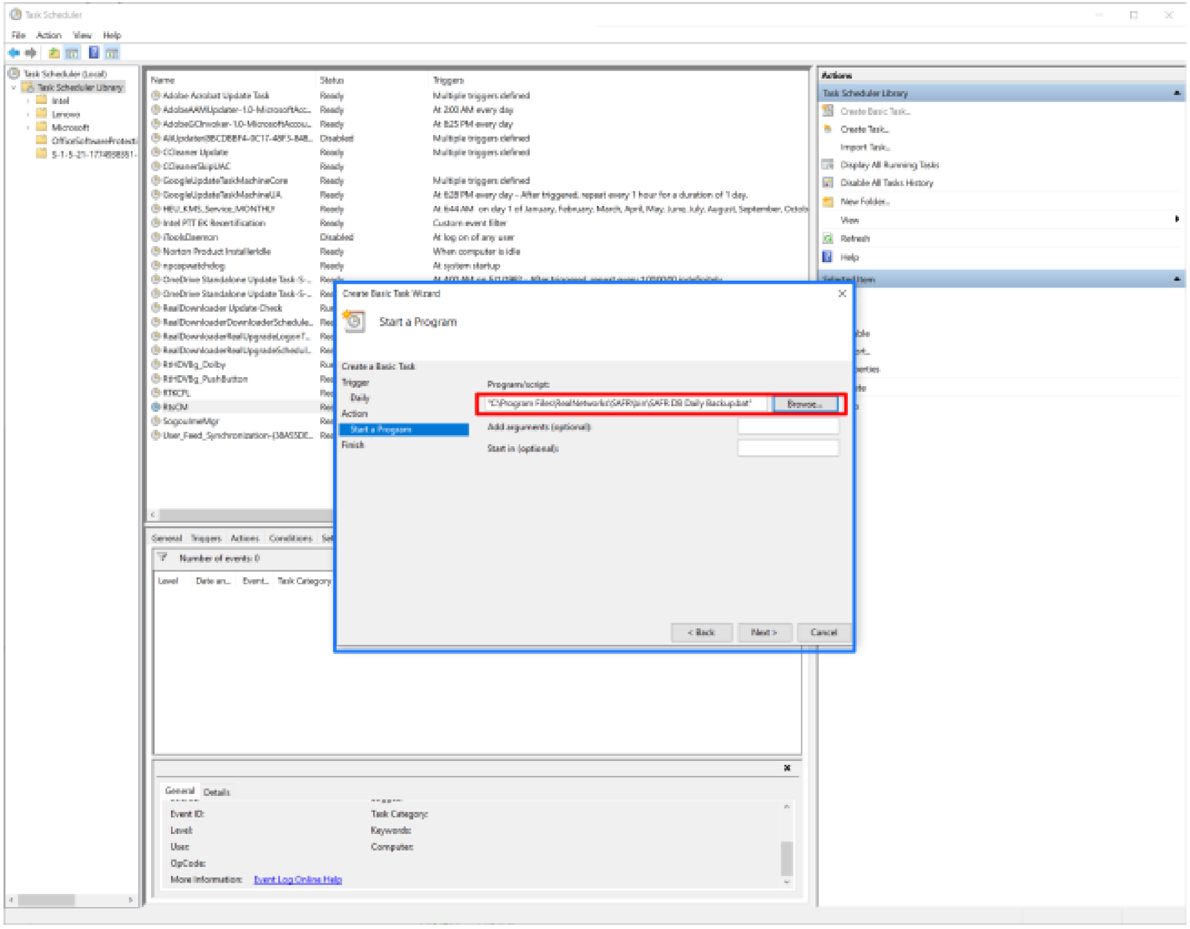
7. Under **Action**, select the **Start a Program** option, and click **Next**.



What action do you want the task to perform?

● Start a program
○ Send an e-mail (deprecated)
○ Display a message (deprecated)

8. Click **Browse** and select the *SAFR DB Daily Backup.bat* file you created in Step 6. Click **Next**.



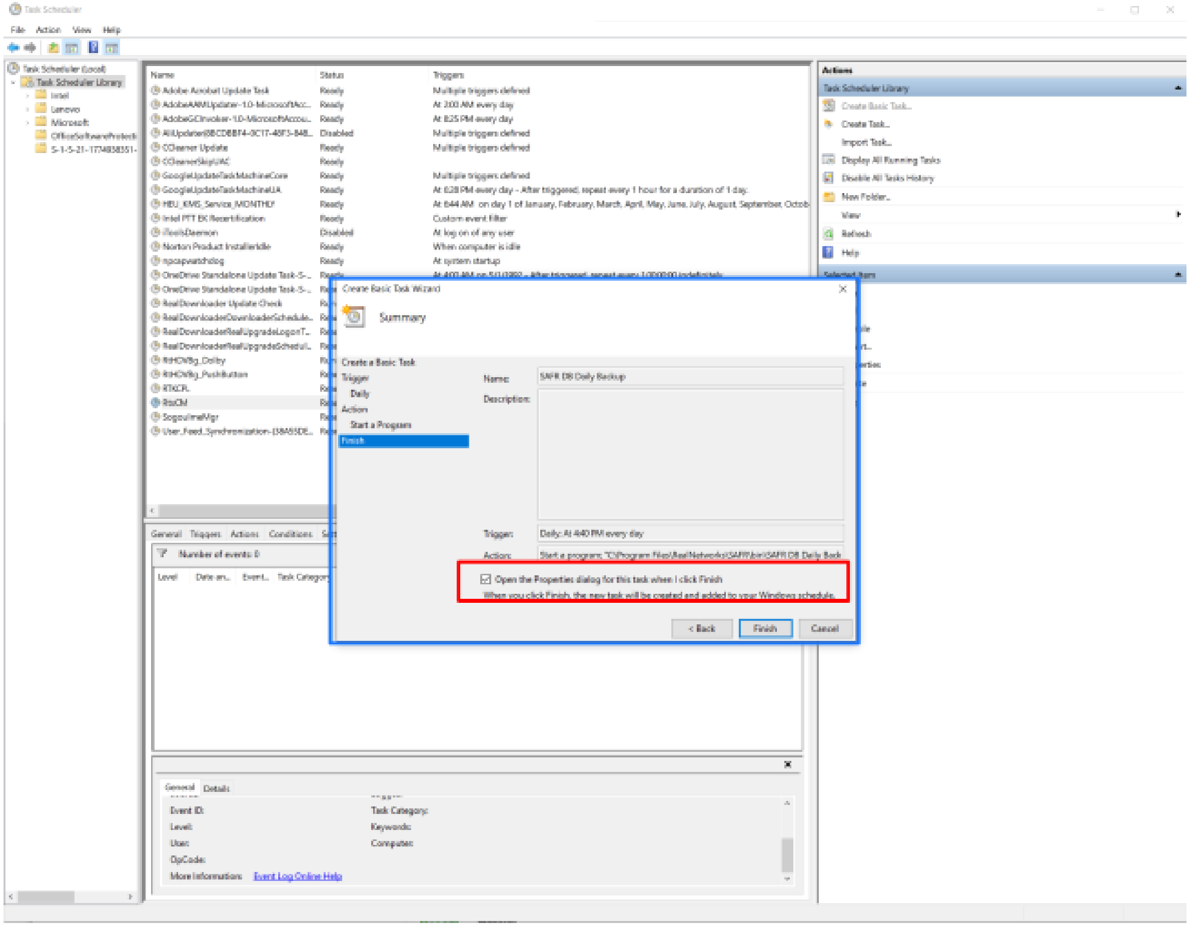
Program/script: "C:\Program Files\RealNetworks\SAFR\bin\SAFR DB Daily Backup.bat"

Add arguments (optional):

Start in (optional):

9. Under **Summary**, select **Open the Properties dialog for this task when I click Finish**, and

click **Finish**.



Create Basic Task Wizard

Summary

Name: SAFR DB Daily Backup

Description:

Trigger: Daily; At 4:40 PM every day

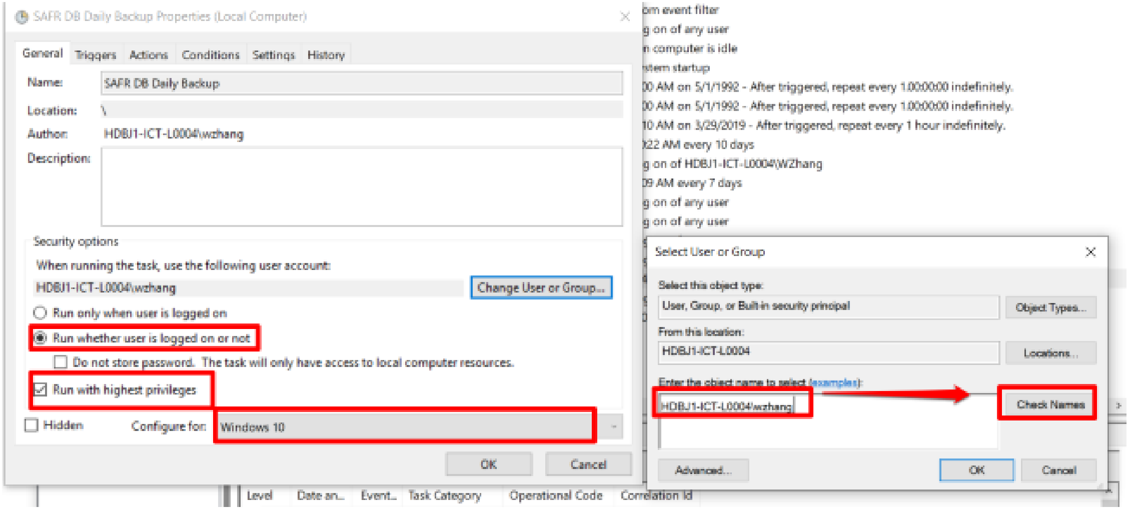
Action: Start a program; "C:\Program Files\RealNetworks\SAFR\bin\SAFR DB Daily Back

☑ Open the Properties dialog for this task when I click Finish

When you click Finish, the new task will be created and added to your Windows schedule.

10. In **Properties**, do the following:

    1. Select the **Run whether user is logged on or not** option.
    2. Select the **Run with highest privileges** option.
    3. Click **Change User or Groups**.
    4. Enter the object name to select in the field. (e.g. DB-USERMACHINE-T/username) Click **Check Name** and click **OK**.



SAFR DB Daily Backup Properties (Local Computer)

Name: SAFR DB Daily Backup

Location: \

Author: HDBJ1-ICT-L0004\wzhang

Description:

Security options

When running the task, use the following user account:

HDBJ1-ICT-L0004\wzhang

○ Run only when user is logged on

● Run whether user is logged on or not

□ Do not store password. The task will only have access to local computer resources.

☑ Run with highest privileges

□ Hidden Configure for: Windows 10

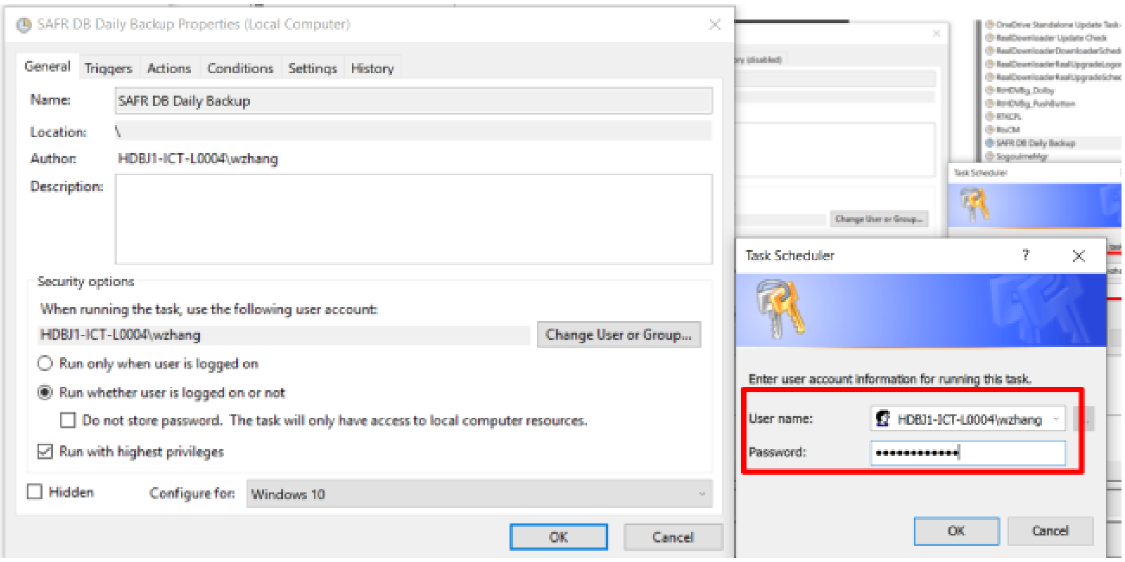
Select User or Group

Select this object type: User, Group, or Built-in security principal

From this location: HDBJ1-ICT-L0004

Enter the object name to select (examples): HDBJ1-ICT-L0004/wzhang

    5. Enter your username and password and click **OK**.



SAFR DB Daily Backup Properties (Local Computer)

General | Triggers | Actions | Conditions | Settings | History

Name: SAFR DB Daily Backup

Location: \

Author: HDBJ1-ICT-L0004\wzhang

Description:

Security options

When running the task, use the following user account:

HDBJ1-ICT-L0004\wzhang

Change User or Group...

○ Run only when user is logged on

● Run whether user is logged on or not

□ Do not store password. The task will only have access to local computer resources.

☑ Run with highest privileges

□ Hidden Configure for: Windows 10

OK Cancel

Task Scheduler

Enter user account information for running this task.

User name: HDBJ1-ICT-L0004\wzhang

Password: •••••••••••••

OK Cancel

Once this procedure is completed, you can find the task in the *Task Scheduler Library*. Check **History** to view the task events. You can find the daily backup file in the path shown in the *SAFR DB Daily Backup.bat* file.

31.1.2 Windows Restore

command path: `C:\Program Files\RealNetworks\SAFR\bin`

run command: `python restore.py BACKUPFILENAME`

- Example: `python restore.py "C:\Program Files\RealNetworks\SAFR\backups\SAFR-backup-20190814-003342.`

Press Y when asked, “Are you sure? (Yy/Nn)”

You’ll receive the following message when the restore is complete:

- `SAFR Restore Complete.`

31.2 On Linux

31.2.1 Linux Backup

command path: `/opt/RealNetworks/SAFR/bin`

run command: `sudo python backup.py`

The backup command generates a backup file at the path `/opt/RealNetworks/SAFR/backups/SAFR-backup-YYYYMMDD-HHMMSS`

You’ll receive the following message when the backup is complete:

- `Backup File: /opt/RealNetworks/SAFR/backups/SAFR-backup-20190814-003342.tgz SAFR Backup Complete.`

31.2.2 Linux Restore

command path: `/opt/RealNetworks/SAFR/bin`

run command: `sudo python restore.py BACKUPFILENAME`

- Example: `sudo python restore.py /opt/RealNetworks/SAFR/backups/SAFR-backup-20190814-083700.tgz`

Press Y when asked, “Are you sure? (Yy/Nn)”

31.2.3 Auto Daily Backup

Script:

```
#backup at 1 a.m every day
0 1 * * * /bin/sh /opt/RealNetworks/SAFR/bin/backup
# remove 7 days before backup files at each sunday 0:30 a.m
30 0 * * 0 find /opt/RealNetworks/SAFR/backups/ -mtime +3 -name "*.tgz"
   -exec rm -rf {} \;
```

Result:

```
root@SAFRDemo:/opt/RealNetworks/SAFR/backups# ls -l
total 6547396
-rw-r----- 1 safr safr 837380012 Sep 11 01:00
   SAFR-backup-20190911-010001.tgz
-rw-r----- 1 safr safr 837423761 Sep 12 01:00
   SAFR-backup-20190912-010001.tgz
-rw-r----- 1 safr safr 837443430 Sep 13 01:00
   SAFR-backup-20190913-010001.tgz
-rw-r----- 1 safr safr 837450675 Sep 14 01:00
   SAFR-backup-20190914-010001.tgz
-rw-r----- 1 safr safr 837588424 Sep 15 01:00
   SAFR-backup-20190915-010001.tgz
-rw-r----- 1 safr safr 837587472 Sep 16 01:00
   SAFR-backup-20190916-010001.tgz
-rw-r----- 1 safr safr 839439035 Sep 17 01:00
   SAFR-backup-20190917-010001.tgz
```

31.3 On macOS

31.3.1 macOS Backup

command path: `/Library/RealNetworks/SAFR/bin`

run command: `sudo python backup.py`

The backup command generates a backup file at the path `/opt/RealNetworks/SAFR/backups/SAFR-backup-YYYYMMDD-HHMMSS`

You'll receive the following message when the backup is complete:

- `Backup File: /Library/RealNetworks/SAFR/backups/SAFR-backup-20190814-003342.tgz`
  `SAFR Backup Complete.`

31.3.2 macOS Restore

command path: `/Library/RealNetworks/SAFR/bin`

run command: `sudo python restore.py BACKUPFILENAME`

- Example: `sudo python restore.py /opt/RealNetworks/SAFR/backups/SAFR-backup-20190814-083700.tgz`

Press Y when asked, "Are you sure? (Yy/Nn)"

You'll receive the following message when the restore is complete:

- `SAFR Restore Complete.`

32 SAFR Server Logging

SAFR Server offers customizable logging at each level for each of the core systems. (i.e. Computer Vision Service (COVI), Event Server, Video Recognition Gateway Admin Service (VIRGA), Reports, and Computer Object Service (CVOS))

32.1 Currently Available Logging Levels

- **ERROR** - Logs exceptions in the code, access issues, incorrect requests, and error conditions that the system cannot recover from.
- **WARN** - Logs when there is a recoverable condition encountered.
- **INFO** - Logs normal execution flow. For example, **INFO** level logs might record that a person was added and removed, a face was added and removed, and a recognition was made.
- **DEBUG** - Logs more low level debug information, such as REST calls to external services. Can also log database access in some cases.
- **TRACE** - Detailed log level for debug only purposes. Contains in some cases the actual data for requests. This logging level is normally not turned on because it can potentially effect performance.

32.2 Currently Available Logging Domains

- **Application** - General logging. Includes handled REST requests as well as other operations.
- **Performance** - Logging that measures the performance of specific areas, such as database access.
- **Access** - Access logging. For example, what REST calls were made or what HTTP response codes were received might be logged..
- **Container** - Specific Tomcat or Jetty container logging. Helps to figure out configuration issues, such as missing Jars etc.
- **Metrics** - Metrics logging such as how much memory was used.
- **STDERR** - Redirected stderr.
- **Localhost** - Platform-specific, starting service.
- **Feature** - Logs dedicated to a specific feature such as sync.log or reaper.log. These logs are variations of the application logs, but they're specific to particular area of functionality.
- **Audit** - Logs specific for auditing of changes to storage of people and faces.

32.3 Configuring Logging Levels

Logging configurations are stored in *logback-spring.xml* files. The *logback-spring.xml* files are in the config directories of the services they pertain to. (e.g. COVI, Events, CVOS, etc.)

There are multiple loggers configured in each *logback-spring.xml*. To set the logging level for an application log you would modify the following section in XML, changing the level to one of the supported log levels.

```
<root level="INFO">
    <appender-ref ref="APP" />
</root>
```

For example, to configure the performance logging, you would change the logging level in its section.

```
<logger name="Performance" level="INFO" additivity="false">

<appender-ref ref="PERF"/>

</logger>
```

In addition, you can selectively change the logging level for a specific code module:

```
<logger name="com.real.cv.event.filter.RequestLoggingFilter"
   level="DEBUG"/>
```

This can be useful when you want to zero in on a specific issue, or when you want to suppress specific logs, or similar situations.

32.4 Default Logging Levels, Locations, and Policies

Core sub-system	Logging domain	Default logging level	Location of log files*	Retention policy	Rotation policy
COVI	Application	WARN	covi-ws.log	14 days	Daily and/or 100MB size
	Performance	OFF	covi.log	14 days	Daily/ or 100MB
	Access	N/A	covi.log	14 days	Daily
	Container	N/A	covi.log	14 days	Daily
	Localhost	N/A	covi.log	14 days	Daily
	STDERR	N/A	covi-stderr.log	14 days	Daily
	Service	N/A	covi-deamon.log	14 days	Daily
	Audit	N/A	covi.log	14 days	Daily
Event Server	Application	WARN	cv-event.log	14 days	Daily and/or 512MB size
	Performance	OFF	cv-event.log	14 days	Daily
	Access	N/A	cv-event.log	14 days	Daily
	Metrics	N/A	cv-event.log	14 days	Daily
	Feature	INF0, DEBUG	cv-event.log, cv-event.log, cv-event.log	14 days	Daily
VIRGA	Application	WARN	virga.log	14 days	Daily
	Performance	OFF	virga.log	14 days	Daily
	Access	N/A	virga.log	14 days	Daily
	Container	INFO	virga.out	N/A	N/A
	Feature	WARN	virga.log	14 days	Daily
Reports	Application	INFO	cv-reports.log	14 days	Daily
	Performance	N/A	cv-reports.log	14 days	Daily
	Access	N/A	cv-reports.log	14 days	Daily
	Metrics	N/A	cv-reports.log	14 days	Daily
	Container	INFO	cv-reports-reports.out	N/A	N/A
CVOS	Application	WARN	cv-object-storage.log	14 days	Daily
	Performance	OFF	cv-object-storage.log	14 days	Daily
	Access	N/A	cv-object-storage.log	14 days	Daily
	Metrics	N/A	cv-object-storage.log	14 days	Daily
	Container	WARN	cv-object-storage-object-storage.out	N/A	N/A

Core sub-system	Logging domain	Default logging level	Location of log files*	Retention policy	Rotation policy
	STDERR	N/A	cv-object-storage-object-storage.err	N/A	N/A

*Relative path based on install location.

33 Database Memory Configuration

Note: This page describes an advanced configuration option for SAFR Server; most users don't need to worry about configuring MongoDB's memory cache.

SAFR Server uses MongoDB to store event data and identity data. By default, MongoDB caches about 9 GB RAM for operation. MongoDB can fault if at any time the full amount of the cache isn't available for use.

However, MongoDB frequently doesn't really need caches that large. You can manually configure MongoDB's cache by editing the `cacheSizeGB` setting in MongoDB's configuration file, mongod.conf. See the sections below for information about how to do so.

33.1 Calculate Required Cache Size

The cache size that MongoDB requires can be calculated according to the following formula:

cacheSizeGB: ((5 * **P**)/1000 + (.5 * **EPD** * **ND**)/1000 + (**NF** * 90/1024))/1000

where

- **P** - Number of people enrolled in the Identity Database.
- **EPD** - Events recorded per day.
- **ND** - Number of recent days for which event data is kept in memory for fast queries. This value is normally set to 2 days.
- **NF** - Number of video feeds being viewed across your SAFR system at the same time. Video feeds can be wiewed via Video Recognition Gateway (VIRGO) feeds, via Desktop Client windows, and via Mobile Client windows.

33.2 Edit the Configuration File

Linux Machines:

The default location for the configuration file is at `/etc/mongod.conf`. The cacheSizeDB setting is located in the storage section of the file:

```
storage:
  dbPath: /var/lib/mongodb
  journal:
    enabled: true
  wiredTiger:
      engineConfig:
         cacheSizeGB: 9
```

Windows Machines:

The default location for the configuration file is at `C:\Program Files\RealNetworks\SAFR\mongo\mongod.conf`. The cacheSizeDB setting is located in the storage section of the file:

```
storage:
  dbPath: C:\ProgramData\RealNetworks\SAFR\mongo\data
  journal:
    enabled: true
  engine: "wiredTiger"
  wiredTiger:
      engineConfig:
         cacheSizeGB: 9
```

34 SAFR Platform Command Line Install Options

34.1 Windows Options

Silent installation of the SAFR Platform on Windows can be achieved by invoking the SAFR Platform installer via the command line and using the `/S` switch.

Example:

```
SAFRPlatform_win_1_8_302_08_13_19.exe /S
```

The Windows SAFR Platform installer provides several options for configuring the component selection during install. Each component can be disabled or enabled by using the following syntax:

- `SAFRPlatform_win_1_8_302_08_13_19.exe /S /COMPONENT=YES`
- `SAFRPlatform_win_1_8_302_08_13_19.exe /S /COMPONENT=NO`

Examples:

```
SAFRPlatform_win_1_8_302_08_13_19.exe /S /VIRGO=YES /Actions=NO

SAFRPlatform_win_1_8_302_08_13_19.exe /Age=YES /Gender=YES /Sentiment=YES
```

Windows Command Line Install Options:

Feature Type	Component	Flag	Default	Notes
Silent Install	Silent Install	/S	Disabled	
Component	SAFR Actions	/Actions	Enabled	
Component	Desktop Client	/Application	Enabled	
Component	Web Console	/Console	Enabled	
Component	SAFR Reports	/Reports	Enabled	
Component	SAFR Logs	/Logs	Enabled	
Component	VIRGA	/VIRGA	Enabled	
Component	VIRGO	/VIRGO	Enabled	
Component	GPU Accelerated Recognition (HTFS)	/GPUFaceService	Enabled	Only enabled by default if NVIDIA Drivers greater than 418.67 are detected.
SAFR Client Component	GPU Accelerated Detection	/CUDA	Enabled	OK to install even if NVIDIA drivers aren’t installed.
Face Service Model	Age Model	/Age	Enabled	
Face Service Model	Gender Model	/Gender	Enabled	
Face Service Model	Mask Model	/Mask	Enabled	
Face Service Model	Masked Identity Model	/MaskIdentity	Enabled	
Face Service Model	Sentiment Model	/Sentiment	Enabled	
Face Service Model	Occlusion Model	/Occlusion	Enabled	
Face Service Model	Optimize GPU Models during installation	/OptimizeModels	Enabled	

Feature Type	Component	Flag	Default	Notes
VMS Integration Plugin	Avigilon Plugin	/Avigilon	Disabled	Only one VMS Plugin allowed. The first specified plugin will be used.
VMS Integration Plugin	Digifort Plugin	/Digifort	Disabled	Only one VMS Plugin allowed. The first specified plugin will be used.
VMS Integration Plugin	Genetec Plugin	/Genetec	Disabled	Only one VMS Plugin allowed. The first specified plugin will be used.
VMS Integration Plugin	GenetecFR Plugin	/GenetecFR	Disabled	Only one VMS Plugin allowed. The first specified plugin will be used.
VMS Integration Plugin	Geutebrueck Plugin	/Geutebrueck	Disabled	Only one VMS Plugin allowed. The first specified plugin will be used.
VMS Integration Plugin	Milestone Plugin	/Milestone	Disabled	Only one VMS Plugin allowed. The first specified plugin will be used.
VMS Integration Plugin	Video Insight Plugin	/VideoInsight	Disabled	Only one VMS Plugin allowed. The first specified plugin will be used.
Camera Extension	Ximea Camera Extension	/Ximea	Disabled	
Installation Location	Installation Location	/D	`C:\Program Files\RealNetworks\SAFR`	Must be the last command line argument. Do not use quotes.

34.2 Linux Options

The Linux SAFR Platform installer currently has command line options to disable the Age, Gender, Occlusion, and Sentiment models by using the following arguments:

Feature Type	Component	Flag	Default
Face Service Model	Age Model	-a, –age	Enabled
Face Service Model	Gender Model	-g, –gender	Enabled
Face Service Model	Mask Model	-m, –mask	Enabled

Feature Type	Component	Flag	Default
Face Service Model	Masked Identity Model	-n, –maskedface	Enabled
Face Service Model	Sentiment Model	-s, –sentiment	Enabled
Face Service Model	Occlusion Model	-o, –occlusion	Enabled
Face Service Model	Face Recognition Identity Model	-i, –identity	Enabled
SAFR Configuration	VIRGO Path	-v, –virgopath	N/A
SAFR Configuration	SAFR User	-u, –user	N/A

All the models are enabled by default, but they can be disabled by setting these arguments to: `off`, `false`, `disabled`, `0`, or `no`. Arguments and their values are case-insensitive. For example, the following commands are all equivalent:

```
sudo bash SAFRPlatform_linux-ubuntu_2_0_022_03_03_20.sh -a=OFF
sudo bash SAFRPlatform_linux-ubuntu_2_0_022_03_03_20.sh -A=false
sudo bash SAFRPlatform_linux-ubuntu_2_0_022_03_03_20.sh --AGE=disabled
sudo bash SAFRPlatform_linux-ubuntu_2_0_022_03_03_20.sh --age=0
sudo bash SAFRPlatform_linux-ubuntu_2_0_022_03_03_20.sh --age=no
```

Multiple models can be disabled simply by specifying multiple arguments:

```
sudo bash SAFRPlatform_linux-ubuntu_2_0_022_03_03_20.sh --age=OFF
   --gender=OFF --sentiment=OFF --occlusion=OFF
sudo bash SAFRPlatform_linux-ubuntu_2_0_022_03_03_20.sh -a=0 -g=0 -s=0
   -o=0
```

35 Video Recognition Gateway (VIRGO)

Video Recognition Gateway (VIRGO) is a daemon system which runs on a Portable Operating System Interface (POSIX)-compatible system. It receives video feeds from one or more cameras and it recognizes and tracks faces in those video streams in real time. It generates tracking events and sends those events to an event server. VIRGO daemons can be managed either through the VIRGO command line interface or through the video feeds window of the Desktop Client or the Web Console.

36 VIRGO System Requirements

36.1 Linux Requirements

Video Recognition Gateway (VIRGO) requires at least the following x86_64 CPU features:

- Ivy Bridge or better CPU architecture
- SSE4
- AVX

A Linux distribution must implement at least the following components:

- LSB support
- systemd

VIRGO on Linux is able to take advantage of GPUs to accelerate video decoding, image processing, face detection, and object detection. The GPU requirements are:

- Nvidia CUDA 10.1 compatible or newer

36.1.1 Ubuntu 16.04(.5+)

The following additional software components must be installed to allow VIRGO to run successfully:

- libcurl3
- libgomp1
- libatomic1
- libbsd0
- libv4l-0

To install the software components listed above, execute the following commands in a shell:

```
sudo apt-get update
sudo apt-get install libcurl3 libatomic1 libgomp1 libv4l-0 libbsd0
```

36.1.2 Ubuntu 18.04(.2+)

The following additional software components must be installed to allow VIRGO to run successfully:

- libcurl4
- libgomp1
- libatomic1
- libbsd0
- libv4l-0

To install the software components listed above, execute the following commands in a shell:

```
sudo apt-get update
sudo apt-get install libcurl4 libatomic1 libgomp1 libv4l-0 libbsd0
```

36.2 macOS Requirements

- macOS 10.11 or newer is required.
- Swift 5 Runtime Support for Command Line Tools must be installed. See this Apple support article for more information.

VIRGO requires the following x86_64 CPU features:

- Ivy Bridge or better CPU architecture
- SSE 4.2
- AVX

VIRGO on macOS is able to take advantage of GPUs to accelerate video decoding, image processing, face detection, and object detection. The GPU requirements are:

- Metal version 1

37 VIRGO Installation Guide

This page describes how to install the standalone Video Recognition Gateway (VIRGO).

Note: If you installed VIRGO when you installed SAFR Platform or SAFR Desktop, there's no need to install standalone VIRGO.

37.1 System Requirements

See the VIRGO System Requirements page before you start the VIRGO installation process. Note that VIRGO depends on certain 4r party libraries which must be installed before installing VIRGO.

37.2 Download the VIRGO Installer

The macOS and Linux standalone VIRGO installers can be downloaded from the SAFR Download Portal here: https://safr.real.com/developers

37.3 VIRGO Installer Package

This package installs VIRGO as a system or user daemon. The system daemon installation ensures that VIRGO will be able to run independently of any logged in user and it will start running as soon as the computer is booted up. Administrator privileges are required to complete the installation. VIRGO will look for factory default settings in the /etc/virgo-factory.conf file. The user installation, on the other hand, links VIRGO to the user who installed it. The VIRGO daemon will only be accessible to this user and it will only run while this user is logged in. However no administrator privileges are required to install and operate VIRGO in this mode. VIRGO will look for factory default settings in the ~/virgo-factory.conf file.

The following sections describe how to use the platform-specific version of the VIRGO installer package.

37.3.1 macOS

Installer name: `Virgo.pkg`

Follow these steps to install VIRGO on your macOS machine:

1. Download the macOS VIRGO installer from the SAFR Download Portal here: https://safr.real.com/developers
2. Double click it.
3. The installer will guide you through the necessary steps to complete the installation. Note that you can choose between a system and user installation by selecting the appropriate install location option.

VIRGO will be installed into the following location:

- For a user installation: `~/Library/RealNetworks`
- For a system installation: `~/Library/RealNetworks`

37.3.2 Linux

Installer name: `virgo_installer.tar.gz`

Follow these steps to install VIRGO on your Linux machine:

1. Download the Linux VIRGO installer from the SAFR Download Portal here: https://safr.real.com/developers
2. Decompress the package: `tar -xzf virgo_installer.tar.gz`
3. Make sure that the necessary third-party library dependencies are installed. For a list of required libraries see the VIRGO system requirements documentation.

4. Run the installer script. The installer script will by default install VIRGO as a system daemon. Although we strongly recommend that you install VIRGO as a system daemon, we do support user daemon installations. You can explicitly specify the desired type of installation by passing the `âĂŞ-user` or `âĂŞ-system` option to the script:
   - `virgo_installer/install.sh --user` installs VIRGO as a user daemon.
   - `virgo_installer/install.sh --system` installs VIRGO as a system daemon.

VIRGO will be installed into the following location:

- System daemon installation: `/opt/RealNetworks`
- User installation: `~/RealNetworks`

The installer script will ask you for all necessary information and guide you through the installation process.

The final VIRGO configuration information is written to a factory configuration file which is stored in the required file system location from where VIRGO is able to read it. Note that for security reasons the factory configuration file is only readable and writeable by the user who owns the VIRGO daemon. The following code block shows an example of how to install VIRGO as a system daemon:

```
> sudo virgo_installer/install.sh
```

37.4 FAQ for macOS Installations

1. I've installed VIRGO as a system daemon. How do I change the factory configuration?

   Place your custom factory configuration file in the `/etc` directory and then reset the VIRGO service like this:

   ```
   Assuming that the factory configuration file is at:

   /etc/virgo-factory.conf

   > virgo service reset
   ```

2. I've installed the VIRGO Package. How do I uninstall VIRGO?

   For system daemon installations, execute the following command from the Terminal:

   ```
   > sudo /Library/RealNetworks/virgo/uninstall.sh
   ```

   For user daemon installations, execute the following command from the Terminal:

   ```
   > ~/Library/RealNetworks/virgo/uninstall.sh
   ```

3. I've installed the VIRGO as a user daemon. How do I stop *virgod*?

   Execute the following command from the Terminal:

   ```
   > launchctl bootout gui/$(id -u)
      ~/Library/LaunchAgents/com.real.virgod.plist
   ```

   This command terminates the *virgod* daemon. Keep in mind that the VIRGO command line tool will automatically restart *virgod* when you use it again.

37.5 FAQ for Linux Installations

1. I've installed VIRGO as a system daemon. How do I change the factory configuration?

   Place your custom factory configuration file in the `/etc` directory and then reset the VIRGO service like this:

```
Assuming that the factory configuration file is at:

/etc/virgo-factory.conf

> virgo service reset
```

2. I’ve installed the VIRGO Package. How do I uninstall VIRGO?

   For system daemon installations, execute the following command from a shell:

   ```
   > sudo /opt/RealNetworks/virgo/uninstall.sh
   ```

   For user daemon installations, execute the following command from the Terminal:

   ```
   > ~/RealNetworks/virgo/uninstall.sh
   ```

3. I’ve installed VIRGO as a user daemon. How do I stop *virgod*?

   Execute the following command in a shell:

   ```
   > systemctl stop --user com.real.virgod.service
   ```

   This command terminates the *virgod* daemon. Keep in mind that the VIRGO command line tool will automatically restart *virgod* when you use it again.

38 VIRGO in the Video Feeds Window

One of the primary ways to manage Video Recognition Gateway (VIRGO) feeds is via the Video Feeds window of the Desktop Client or the Web Console. The sections below describe how to do so.

38.1 Create a VIRGO Feed

Newly created VIRGO feeds have the following properties:

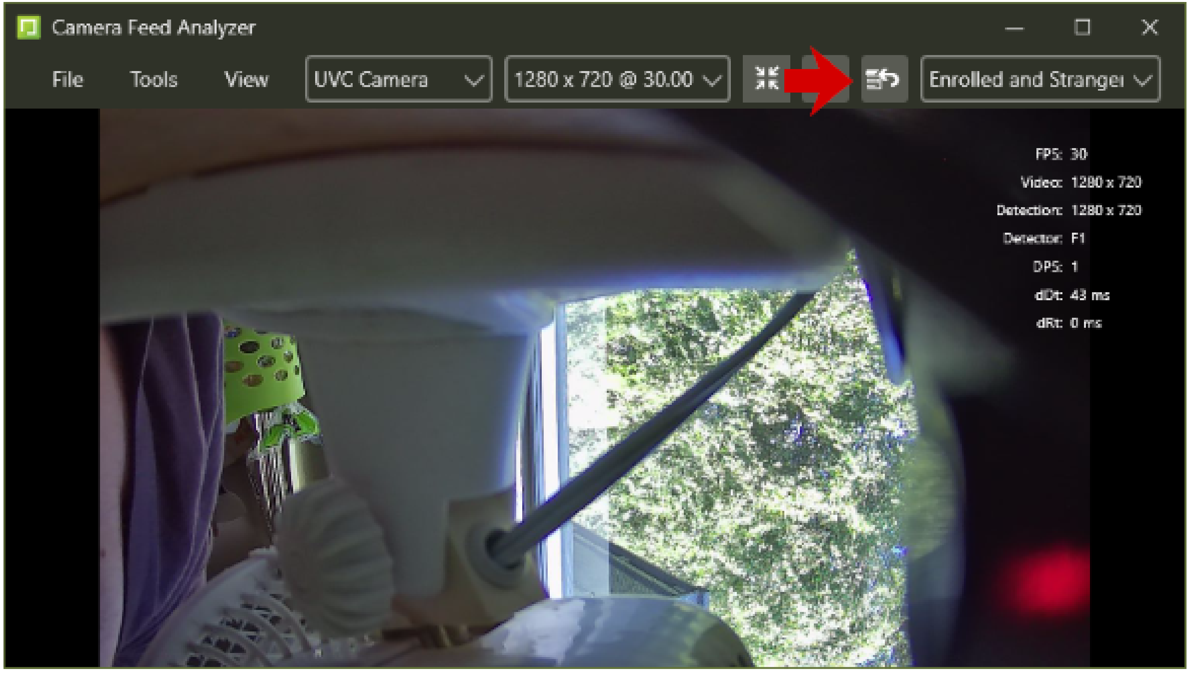
- The feed is not affected by subsequent changes to preferences made within the Desktop Client. If you change a preference within your Desktop Client, that change is not cloned to the existing VIRGO video feed(s).
- Feeds continue running and processing their video streams regardless of whether or not the Desktop Client is running or not.
- If you shut down your machine, the video feed will try to restart itself whenever your machine is turned on again.

38.1.1 Camera Feed Analyzer Method

Windows only.

The easiest way to create a VIRGO feed is to do the following:

1. Open the Desktop Client.
2. Connect a camera to the client in the *Camera Feed Analyzer* window. (i.e. the default window)
3. Press the **Add to Video Feeds for continuous processing in the background** button as shown below highlighted by the red arrow.



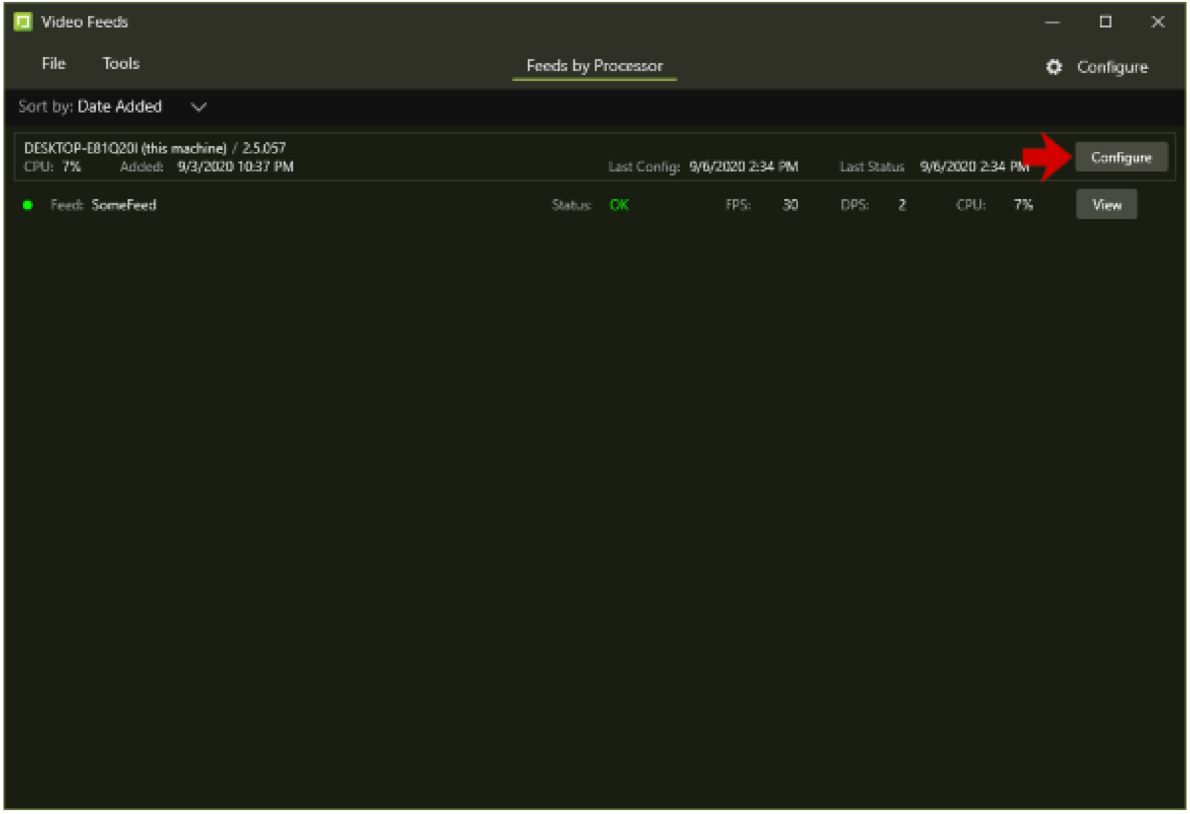
FPS: 30
Video: 1280 x 720
Detection: 1280 x 720
Detector: F1
DPS: 1
dDt: 43 ms
dRt: 0 ms

4. You'll be prompted for the following information:
   - **Feed Name**: Enter any name for the video feed you wish.
   - **Mode**: Select the video processing mode from the drop-down menu that you want the VIRGO daemon to operate in. For a description of the video processing modes, see here.
   - **Processor**: The machine where you want the VIRGO feed to run. The current machine is selected by default.
   - **Apply Mode Customizations from Preferences**: Enable if you want the Desktop Client preferences applied to the new VIRGO feed.

38.1.2 Manual Method

The manual way to create a VIRGO feed is as follows:

1. Connect a camera to SAFR.
2. Open the *Video Feeds* window and click the **Configure** button for your current machine.



Sort by: Date Added

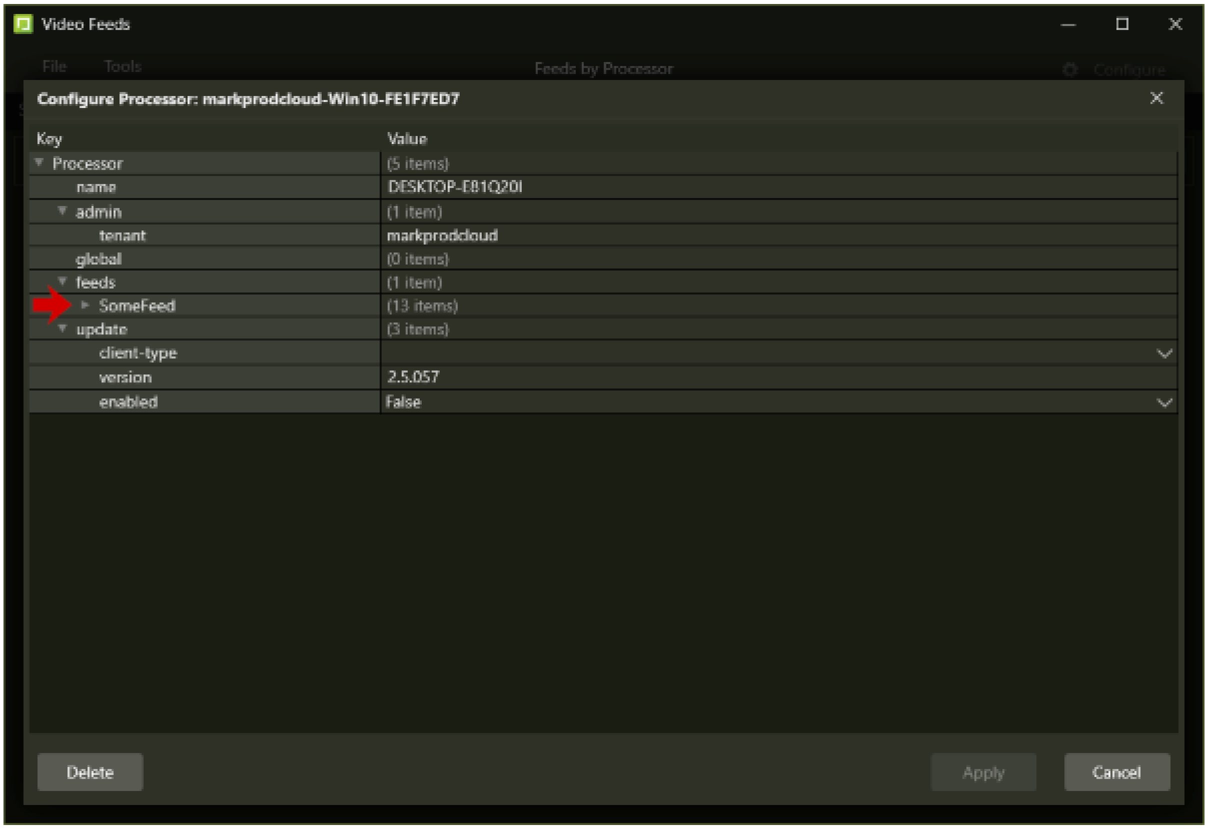
DESKTOP-E81Q20I (this machine) / 2.5.057
CPU: 7% Added: 9/3/2020 10:37 PM Last Config: 9/6/2020 2:34 PM Last Status 9/6/2020 2:34 PM

Feed: SomeFeed Status: OK FPS: 30 DPS: 2 CPU: 7%

3. Hover your mouse over **Feeds**, then click the + button.
4. You'll be prompted for the following information:
   - **Feed Name**: Enter any name for the video feed you wish.
   - **Camera**: Select the camera feed you connected in Step #2 from the drop-down menu.
   - **Format**: Select the resolution and frame rate for the new video feed.
   - **Mode**: Select the video processing mode from the drop-down menu that you want the VIRGO feed to operate in. For a description of the video processing modes, see here.
   - **Apply Mode Customizations from Preferences**: Enable if you want the Desktop Client preferences applied to the new VIRGO feed.
5. Click the **Add** button.
6. Click **Apply** in the bottom right corner of the *Video Feeds* window.
   Note: If you close the *Video Feeds* window without first clicking **Apply**, the video feed won't be created.

38.2 Manage VIRGO Feeds

To manage VIRGO feeds, open the *Video Feeds Window* within either the Desktop Client or the Web Console, then click the **Configure** button for your current machine. You'll see a screen similar to the following: (**Note**: You can often expose additional properties by clicking on the arrow next to entries, as shown by the arrow next to the *SomeFeed* entry below.)



Configure Processor: markprodcloud-Win10-FE1F7ED7

Key	Value
Processor	(5 items)
name	DESKTOP-E81Q20I
admin	(1 item)
tenant	markprodcloud
global	(0 items)
feeds	(1 item)
SomeFeed	(13 items)
update	(3 items)
client-type	
version	2.5.057
enabled	False

- **name**: The name of the machine being managed.
- **admin**: Contains admin properties.

Property	Description
resource-type	
tenant	The tenant being managed.

- **global**: Contains global properties. See Global Properties for more information.
- **monitoring**: Monitoring properties allow you to monitor a video feed's health and send a notification email if one of the health metrics degrades to a certain level.
  To set up notification emails, do the following:
  1. Set up an SMTP Email Service on the Status Page of the Web Console.
  2. Enable one or more of the alarm conditions of the video feed's monitoring properties. There are 7 conditions available:
     1. **delinquent**: The video feed has stopped responding/sending status updates.
     2. **feed.error**: The video feed has encountered an error.
     3. **lowRAM**: The host machine has low RAM memory.
     4. **lowDisk**: The host machine has low hard drive storage space.
     5. **lowGPUMemory**: The host machine has low GPU memory.
     6. **lowCPU**: The host machine has low CPU processing power.
     7. **lowGPU**: The host machine has low GPU processing power.
  3. Set the subject and message properties for your enabled conditions.
  4. Set the threshold property for enabled conditions. (**Note**: The *delinquent* and *feed.error* conditions don't have threshold properties),
  5. Set the *alarm.mail.username* and *alarm.mail.password* properties to your email credentials.

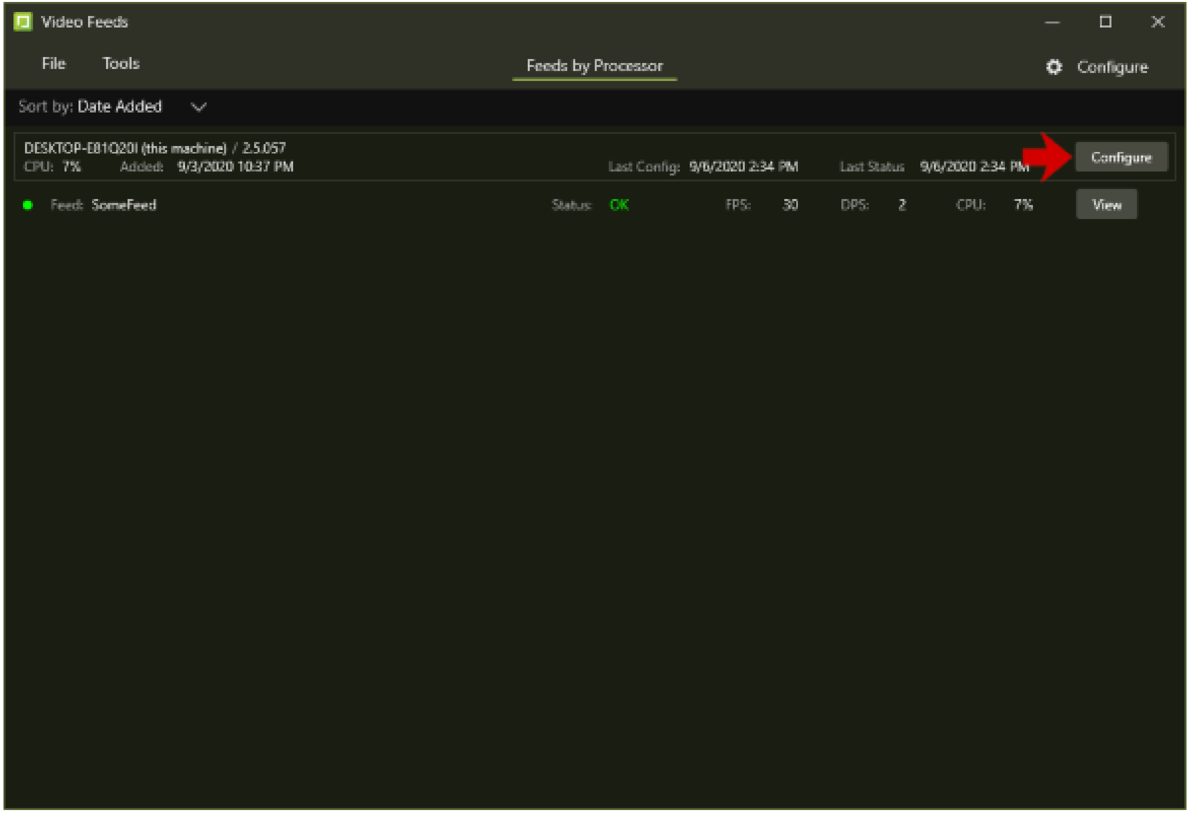
6. Set *alarm.mail.enabled* to TRUE to enable the notification emails.
   See Monitoring Properties for a list of all the available monitoring properties.

- **feeds**: Specifies the feed properties. See Feeds Properties for more information.
- **update**: The update properties are not intended for public consumption at this time.

38.3 Terminate a VIRGO Feed

VIRGO feeds that haven't been terminated will continuously run in the background, and will automatically restart themselves after system shutdowns and reboots. Because each video feed consumes a significant amount of CPU resources, you'll want to terminate video feeds that are no longer of interest to you. Do the following to terminate a video feed:

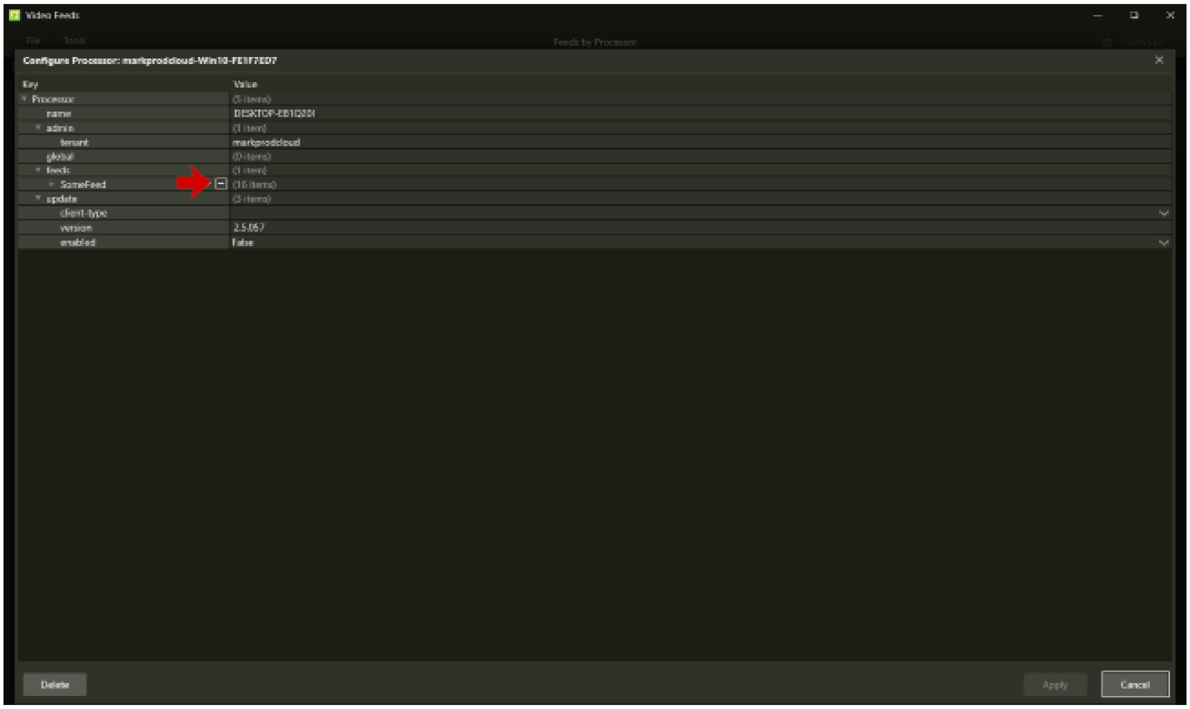
1. Open the *Video Feeds Window* within either the Desktop Client or the Web Console.
2. Click the **Configure** button.



DESKTOP-E81Q20I (this machine) / 2.5.057
CPU: 7% Added: 9/3/2020 10:37 PM Last Config: 9/6/2020 2:34 PM Last Status 9/6/2020 2:34 PM

● Feed: SomeFeed Status: OK FPS: 30 DPS: 2 CPU: 7%

3. When you hover your mouse over name of the feed that you want to terminate, you'll see a + button and a **-** button. Click the **-** button.



Configure Processor: markprodcloud-Win10-FE1F7ED7

Key	Value
Processor	(5 items)
name	DESKTOP-EB1Q20I
admin	(1 item)
tenant	markprodcloud
global	(0 items)
feeds	(1 item)
SomeFeed	(16 items)
update	(3 items)
client-type	
version	2.5.057
enabled	False

4. Click the **Apply** button in the bottom right of the window. At this point, the feed will be terminated. **Note**: If you click the **Cancel** button, the feed termination is undone and the feed will continue operating.

39 VIRGO Command Line Interface

The command line interface is designed based on an object - verb structure.

- Video Recognition Gateway (VIRGO) is conceptually organized into sub-systems which are represented by "objects".
- "Verbs" are commands that can be issued on an object.
- Some verbs may require additional parameters.

VIRGO currently defines the following types of objects (subsystems):

- **Service**: The VIRGO daemon itself.
- **Feed**: A video stream. (e.g. from a camera)
- **Environment**: The environment to which *virgod* connects.

The sections below describe the VIRGO command line syntax. Note that VIRGO command line options follow the standard Portable Operating System Interface (POSIX) convention. This means that many of those options come in a short (single dash prefix) and a long (double dash prefix) form.

39.1 Command Line Options

Help

```
> virgo --help
> virgo -h
<help text>
```

Shows all available VIRGO command line options.

39.1.1 Administrator

Get the current administrator configuration

```
> virgo administrator get
```

This command causes VIRGO to print the current administrator configuration. VIRGO may either be administrated by a cloud server (aka VIRGA) or it may be self-administrated. 'Virgo' is printed in the former case 'Virga' in the later.

Setting the administrator configuration

```
> virgo administrator set <name>   // <name> is either 'virga' or 'virgo'
Administrator: <name>
```

This command causes VIRGO to switch to the specified administrator. Pass 'virga' if VIRGO should be administrated via the VIRGA server. Note that the environment definition must contain an admin-server-url entry in this case. Pass 'virgo' if VIRGO should be used standalone without a cloud command & control server. Standalone mode allows you to freely add, remove, and change feeds whereas the VIRGA administration mode requires that feeds are added, removed, and changed via VIRGA.

39.1.2 Service

Get information about the VIRGO service

```
> virgo service info
Version:       1.0.0
Target:        x86_64-macos
Domain:        System
Administrator: Virga
Environment:   PROD
```

```
Client ID:      <client-id>
Client Type:    <client type>
```

This command prints the following information about the installed VIRGO daemon build and its fundamental configuration.

- **Version**: The VIRGO build version.
- **Target**: Specifies for which operating system and CPU architecture the VIRGO daemon was built.
- **Domain**: Specifihies whether the VIRGO daemon is running as a system-wide daemon (system) or a daemon which is only available to the currently logged in user (user). Note that user-wide VIRGO daemons will terminate when the user logs out.
- **Environment**: The environment to which the VIRGO daemon connects in order to receive commands from the command & control server.
- **Client ID**: The client ID that the VIRGO daemon sends to the command & control server to identify itself.

Get the current service status

```
> virgo service status
camera_1: ok
camera_2: ok
camera_3: inactive
```

This command tells VIRGO to print the current status of all registered feeds.

Monitor the current status of all feeds

```
> virgo service monitor
```

This command enables the service monitor. See Service Monitoring for more information.

Logging

```
> virgo service log <log specification>
```

This command displays the current service log. See Service Logging for more information.

Resetting the VIRGO daemon state

```
> virgo service reset
```

This command tells VIRGO that it should delete its current state and reinitialize it from the contents of the factory configuration file. This effectively resets the daemon back to the factory state.

Updating VIRGO

```
> virgo service update <version> <url> [--verbose]    // download an
   install a new version.
> virgo service update <version> [--verbose]          // switch virgo to
   a previously installed version. E.g. downgrade to an old version.
```

This command causes VIRGO to upgrade or downgrade to the specified version. <version> is the version to upgrade or downgrade to and <url> is a file or HTTP/HTTPS URL that points to VIRGO update archive. Specifying the update archive URL is only necessary if the version you are trying to switch to isn't already installed on the machine. By default VIRGO shows the current update status and progress. Specify the "–verbose" switch to cause VIRGO to show the full update log instead.

Get information about the installed VIRGO versions

```
> virgo service versions
Installed:
   1.0.0
   1.1.0
-> 1.2.0

Current:
   1.2.0
```

This command causes VIRGO to print the version numbers of all installed VIRGO packages plus the version number of the currently active and running VIRGO daemon.

39.2 Environment

A VIRGO daemon has a built-in list of supported environments. Only one of those environments can be active at a given time. The active environment determines to which VIRGA, face recognition, and event servers *virgod* and its *virgafeedd* child processes will talk.

List supported environments

```
> virgo environment list
DEV
INT2
LOCAL
PROD
```

Lists all environments supported by VIRGO.

Get the active environment

```
> virgo environment get [--verbose]
PROD
```

Returns the currently active environment. This is the environment to which *virgod* and all of its *virgofeedd* daemons connect. Additionally VIRGO will show the URLs of the individual servers in the environment if you pass the `--verbose` flag.

Set the active environment

```
> virgo environment set <environment name> [--verbose]
OK
```

Sets the environment which VIRGO and its feeds will use. Note that <environment name> must be one of the supported environments or one of the custom environments defined in the factory configuration file. Note that changing the environment also resets the VIRGO daemon back to the factory configuration.

By default the command prints “OK” if the switch to the new environment succeeds, while it prints an error if one or more services can not be contacted. You can pass the `--verbose` flag to get a detailed status for each service.

39.3 Cloud User

Get cloud account details

```
> virgo user get
User ID: <user id>
Password: ***
```

Prints the *User ID* and and an indication whether a password was provided. Three asterisk characters indicate that VIRGO has a password on file, while an empty password line indicates that VIRGO doesn't have a password for the user on file.

Set the cloud account

```
> virgo user set
User ID: <user id>
Password: ***
```

Replaces the current cloud account's credentials with the provided *User ID* and *Password*. All currently enabled feeds are automatically restarted with the new account information.

39.4 Feeds

A single *virgod* daemon instance is capable of managing a set of video feeds. *Virgod* spawns one *virgofeedd* instance per feed and this *virgofeedd* instance is exclusively responsible for tracking its assigned feed. *Virgod* automatically respawns a *virgofeedd* instance if it dies unexpectedly.

A feed has:

- A name which is used to identify a particular feed.
- An RTSP URL which provides access to the video stream.
- Default face detection, recognition, and tracking parameters.
- Additional information to control features like lens correction.

Virgod stores the configuration information for a feed persistently. A feed can be added, removed, started, and stopped at any time. A VIRGO instance may come prepackaged with the configuration information for one or more feeds. New feeds may be added dynamically any time as long as *virgod* is running.

List feeds

```
> virgo feed list
camera_1
camera_2
```

Lists all enabled and disabled feeds that have been registered with VIRGO.

Get the configuration information for a feed

```
> virgo feed get <feed name>
{
   "active":true
   "url":"rtp://camera.is.here/with/stream:8789"
   ...
}
```

Prints the feed configuration JSON dictionary.

Update/set the configuration information for a feed

```
> virgo feed set <feed name> <feed config file path>
```

Updates the current configuration of the feed with name <feed name>. The feed configuration file is read and the properties in the configuration file are applied to the current feed configuration stored in VIRGO. The feed configuration file is a JSON file with a single dictionary which contains the feed properties that you want to change. Note that the feed configuration file only needs to contain those properties that you want to change. See Video Feeds Properties for a list of supported feed properties.

Get the PID of a feed

```
> virgo feed get-pid <feed name>
53280
```

Prints the *PID* of the feed. -1 is printed if the feed is currently not active and thus no feed daemon is running to process the feed video stream.

Get the status of a feed

```
> virgo feed status <feed name>
ok
```

Prints the current status of a feed.

Add a new feed

```
> virgo feed add <feed name> <feed config file path>
```

Adds a new feed configuration to the persistent list of feeds. The feed name must be unique with respect to the VIRGO instance. The feed configuration is read from the supplied feed configuration file. The feed will immediately start processing if it is marked as “enabled” in the configuration file; otherwise the feed will be added to the persistent list of feeds but a separate “virgo feed start <feed name>” command will have to be issued to cause the feed to start running.

Remove an existing feed

```
> virgo feed remove <feed name>
```

VIRGO will stop the feed and then remove the feed configuration information from its persistent feed table.

Starting a feed

```
> virgo feed start <feed name>
```

VIRGO will mark the feed as active and start processing it. A video file feed starts processing from the beginning of the video while a camera feed starts processing from the current time code of the video stream. If the feed is already active and running this command instead does nothing.

Stopping a feed

```
> virgo feed stop <feed name>
```

Marks the feed as inactive and stops processing the video stream. If the feed is already marked as inactive, then this command instead does nothing.

Capturing an image from a feed

```
> virgo feed capture-image <feed name> <url or path> [--size
   <image_size>] [--max-frames <max_number_of_frames>] [--frame-delay
   <delay_in_milliseconds>]
```

Enables capturing of a single image or a series of consecutive images from the specified feed. <url or path> is a file or HTTP URL or a file system path. The URL/path is expected to point to a directory. VIRGO will create the directory if necessary and it will write all captured images to this directory. The size of the larger side of the capture image can be specified with the `--size` option. The default capture image size is 720 pixels. The maximum number of consecutive frames that should be captured can be specified with the `--max-frames` option. The default is to capture a single image. The `--frame-delay` option allows you to specify the delay between consecutive frames in milliseconds.

40 Video Feeds Properties

40.1 Global Properties

Below is a list of global processor properties.

Property	Default	Description
status-interval	500	Status reporting time interval in milliseconds.
avigilon	N/A	Exposes the properties related to the SAFR Avigilon integration.
genetec	N/A	Exposes the properties related to the SAFR Genetec and SAFR Genetec FaceRec integrations.
digifort	N/A	Exposes the properties related to the SAFR Digifort integration.
milestone	N/A	Exposes the properties related to the SAFR Milestone integration.
geutebrueck	N/A	Exposes the properties related to the SAFR Geutebrueck integration.
videoInsight	N/A	Exposes the properties related to the SAFR Panasonic Video Insight integration.

40.2 Monitoring Properties

Below is a list of all the monitoring properties.

Property	Default Value	Description
alarm.condition.delinquent.enabled	TRUE	Enables the deliquent alarm condition.
alarm.condition.delinquent.subject	“SAFR Feed Processor Unresponsive”	Sets the text of the subject line of the delinquent notification mail.
alarm.condition.delinquent.message	“SAFR Feed Processor %s is not responding.”	Sets the text of the email body of the delinquent notification mail.
alarm.condition.feed.error.enabled	TRUE	Enables the feed.error alarm condition.
alarm.condition.feed.error.subject	“SAFR Feed Error”	Sets the text of the subject line of the feed.error notification mail.
alarm.condition.feed.error.message	“SAFR Feed Processor %s feed %s encountered an error %d: %s.”	Sets the text of the email body of the feed.error notification mail.
alarm.condition.lowRAM.enabled	TRUE	Enables the lowRAM alarm condition.
alarm.condition.lowRAM.subject	“SAFR Feed Processor low on RAM”	Sets the text of the subject line of the lowRAM notification mail.
alarm.condition.lowRAM.thresholdGB	0.5	The threshold, in GB, below which the lowRAM alarm condition is triggered.

Property	Default Value	Description
alarm.condition.lowRAM.message	“SAFR Feed Processor %s RAM remaining is at %f.1GB which is below healthy threshold of %f.1GB.”	Sets the text of the email body of the lowRAM notification mail.
alarm.condition.lowDisk.enabled	TRUE	Enables the lowDisk alarm condition.
alarm.condition.lowDisk.subject	“SAFR Feed Processor low on disk space”	Sets the text of the subject line of the lowDisk notification mail.
alarm.condition.lowDisk.thresholdGB	1.0	The threshold, in GB, below which the lowDisk alarm condition is triggered.
alarm.condition.lowDisk.message	“SAFR Feed Processor %s disk space remaining is at %f.1GB which is below healthy threshold of %f.1GB.”	Sets the text of the email body of the lowDisk notification mail.
alarm.condition.lowGPUMemory.enabled	TRUE	Enables the lowGPUMemory alarm condition.
alarm.condition.lowGPUMemory.subject	“SAFR Feed Processor low on GPU memory”	Sets the text of the subject line of the lowGPUMemory notification mail.
alarm.condition.lowGPUMemory.thresholdGB	0.3	The threshold, in GB, below which the lowGPUMemory alarm condition is triggered.
alarm.condition.lowGPUMemory.message	“SAFR Feed Processor %s GPU memory remaining is at %f.1GB which is below healthy threshold of %f.1GB.”	Sets the text of the email body of the lowGPUMemory notification mail.
alarm.condition.lowCPU.enabled	TRUE	Enables the lowCPU alarm condition.
alarm.condition.lowCPU.subject	“SAFR Feed Processor low on CPU”	Sets the text of the subject line of the lowCPU notification mail.
alarm.condition.lowCPU.thresholdPercent	15.0	The threshold, as a percentage, below which the lowCPU alarm condition is triggered.
alarm.condition.lowCPU.message	“SAFR Feed Processor %s CPU capacity remaining is at %f.1% which is below healthy threshold of %f.1%.”	Sets the text of the email body of the lowCPU notification mail.
alarm.condition.lowGPU.enabled	TRUE	Enables the lowGPU alarm condition.
alarm.condition.lowGPU.subject	“SAFR Feed Processor low on GPU”	The threshold, as a percentage, below which the lowGPU alarm condition is triggered.
alarm.condition.lowGPU.thresholdPercent	15.0	The threshold, as a percentage, below which the lowGPU alarm condition is triggered.
alarm.condition.lowGPU.message	“SAFR Feed Processor %s GPU capacity remaining is at %f.1% which is below healthy threshold of %f.1%.”	Sets the text of the email body of the lowGPU notification mail.
alarm.mail.enabled	TRUE	Enables alarm notification mails.

Property	Default Value	Description
alarm.mail.recipients	N/A	A comma-separated list of emails that specifies who should receive email notifications.
alarm.mail.username	N/A	The username for your email service.
alarm.mail.password	N/A	The password for your email service.

40.3 Feeds Properties

Property	Default Value	Description
accelerator	"auto"	The type of acceleration that a feed should use. There are three possible values: **auto** - VIRGO will automatically pick the best available acceleration type. For example, VIRGO will assign the feed to one of the available GPUs if there is still processing capacity available. Otherwise VIRGO will assign the feed to the CPU. **cpu** - The feed will exclusively run on the CPU and not use any GPU even if a GPU is available. **gpu** - The feed will exclusively run on a GPU and not use the CPU for video decoding, graphics processing, or detection. The feed will fail if no GPU is available. This property is not supported by SAFR Inside.
accelerator.gpu-id	0	The GPU identifier to use when GPU acceleration is in use. This property is only used when the **accelerator** property is set to *gpu* or *auto*, and a GPU is being used. If the **accelerator.gpu-id** property is specified it will force the specific GPU to be used, and if failure occurs SAFR will fall back to the CPU. **accelerator.gpu-id** is an advanced setting that should only be used in very specific cases. This property is not supported by SAFR Inside.

Property	Default Value	Description
capture.frame-delay	30	Wall-clock time between consecutive frame captures. If this value is set to 0 then VIRGO will capture frames as fast as the native frame rate is playing the video. This property is not supported by SAFR Inside.
capture.maximum-frames	3600	If > 0, enables the capture of “max-frames” frames; if 0, disables capture. This property is not supported by SAFR Inside.
capture.size	480	Specifies size of the smaller dimension of the image that will be sent. This property is not supported by SAFR Inside.
detector.detect-badges	FALSE	Whether detection of badges should be enabled for this feed. This property is not supported by SAFR Inside.
detector.detect-faces	TRUE	Whether detection of faces should be enabled for this feed.
detector.detect-faces-input-size	“normal”	Sets the face detector input size. This property allows you to manage the trade-off between accuracy vs. speed. There are 3 possible values: **normal** - This is the standard against which the other 2 possible values are measured. **small** - Decreased accuracy but increased speed. **large** - Increased accuracy but decreased speed.

Property	Default Value	Description
detector.detect-faces-service	“auto”	Specifies which face detection service will be used: **standard** - The standard facial detection service that SAFR uses. **high-sensitivity** - A high sensitivity facial detection service which has a lower latency and whose performance doesn’t degrade when multiple faces are being analyzed simultaneously. The high sensitivity service consumes many more GPU resourcs than the standard service. **auto** - This value will automatically select the high sensitivity service if sufficient GPU resources are available to run it. If there are insufficient GPU resources, then the standard service is used instead. This property is not supported by SAFR Inside.
detector.detect-people	FALSE	Whether detection of people should be enabled for this feed. This detects any part of a person’s body and not just the face. This property is not supported by SAFR Inside.
detector.detect-people-every-n-frames	1	This can be used to avoid running person detection on every frame. Since person detection requires a lot of GPU processing if the hardware is not powerful enough this value can be changed so that we only attempt to detect people every Nth frame to save processing power to keep up with realtime detection. This property is not supported by SAFR Inside.

Property	Default Value	Description
detector.detect-people-input-size	"normal"	Valid values: **normal** - 416 pixel input. People detection balances speed and accuracy for best results. **small** - 320 pixel input. People detection will be the fastest with this input, but least accurate. **large** - 608 pixel input. People detection will be the slowest with this input, but most accurate. This property is not supported by SAFR Inside.
detector.detect-people-model	"balanced"	Valid values: **max-accuracy** - Use a larger model for better accuracy, but the speed will be slower. **max-speed** - Use a smaller model for faster speed, but the accuracy will be lower. **balanced** - Use a larger model for better accuracy, but the precision will be slightly lower resulting in faster speeds than the *max-accuracy* model without sacrificing too much accuracy. This property is not supported by SAFR Inside.
detector.detect-rgb-liveness	FALSE	Enables RGB liveness detection. This property is not supported by SAFR Inside.
detector.detect-vehicle	N/A	Internal use only.
detector.detect-vehicle-every-n-frames	N/A	Internal use only.
detector.detect-vehicle-input-size	N/A	Internal use only.
detector.detect-vehicle-model	N/A	Internal use only.
detector.face-sensitivity-threshold	0	The sensitivity threshold when using the *High Sensitivity* facial detection service. The lower this value is, the more lenient the facial detection service will be when attempting to recognize a face, which can result in additional false positives. This setting is only available if you selected *High Sensitivity* for the **Detection service** setting above. This property is not supported by SAFR Inside.

Property	Default Value	Description
detector.final-face-selection-threshold	0.9	The final face candidate threshold that is used during face detection. This property is not supported by SAFR Inside.
detector.initial-face-selection-threshold	0.8	The initial face candidate threshold that is used during face detection. This property is not supported by SAFR Inside.
detector.maximum-concurrent-detections	1	The maximum number of concurrent detections to allow. 0 means to automatically set this. This property is not supported by SAFR Inside.
detector.maximum-input-resolution	720	Maximum resolution of the Input image. Bigger images are scaled down (aspect-ratio preserving) to this resolution before detection.
detector.maximum-input-resolution-badges	4320	Maximum resolution of the Input image. Bigger images are scaled down (aspect-ratio preserving) to this resolution before detection. This property is not supported by SAFR Inside.
detector.middle-face-selection-threshold	0.85	The middle face candidate threshold that is used during face detection. This property is not supported by SAFR Inside.
detector.minimum-consecutive-detections-required-person	0	This is the number of consecutive detections that are required before reporting that the person (based on object id) was actually detected and can be used to filter out false positives. This property is not supported by SAFR Inside.
detector.minimum-consecutive-detections-required-vehicle	N/A	Internal use only.
detector.minimum-required-badge-size	0	The minimum size of badges to accept from the detector. Only badges with at least this size are eligible for recognition. This property is not supported by SAFR Inside.
detector.minimum-required-face-size	0	minimum size of faces to accept from the detector. Only faces with at least this size are eligible for recognition.

Property	Default Value	Description
detector.minimum-required-person-to-screen-height-proportion	0	Specifies the ratio of the person to the screen height. This can be between 0 - 1 and allows for decimal precision. For example, if you don't want the person to show up unless they are greater than 25% of the image height then specify a value of 0.25. This property is not supported by SAFR Inside.
detector.minimum-required-vehicle-to-screen-proportion	N/A	Internal use only.
detector.minimum-searched-badge-size	20	The badge detector is advised to search for badges of at least this size. This value is applied while searching the image. This property is not supported by SAFR Inside.
detector.minimum-searched-face-size	80	The face detector is advised to search for faces of at least this size. This value is applied while searching the image. This property is not supported by SAFR Inside.
detector.person-detection-threshold	0.4	This is the detection threshold to use when matching objects. The higher the threshold the more strict the matching will be and the higher the confidence will be that the actual object matches. This property is not supported by SAFR Inside.
detector.person-separation-threshold	0.45	This threshold controls the object separation when the objects are overlapping. This determine how much overlap is needed before no longer detecting the object with the weaker footprint. This property is not supported by SAFR Inside.

Property	Default Value	Description
detector.rgb-liveness-detection-scheme	“strict”	Specifies which RGB liveness model(s) should be used: **Fast Unimodal**: Only the Texture model will be used. **Normal Unimodal**: Only the Context model will be used. **Strict Multimodal**: Both the Texture and Context models will be used. For a subject to pass the RGB liveness test, both of the results of the models must meet or exceed the *Liveness detection threshold* value. **Normal Multimodal**: Both the Texture and Context models will be used. For a subject to pass the RGB liveness test, the average of the results of the two models must meet or exceed the *Liveness detection threshold* value. **Tolerant Multimodal**: Both the Texture and Context models will be used. Subjects pass the RGB liveness test when the result of either model meets or exceeds the *Liveness detection threshold* value. This property is not supported by SAFR Inside.
detector.rgb-liveness-evaluate-fake-over-n-frames	20	The number of frames over which fakeness should be evaluated. This property is not supported by SAFR Inside.
detector.rgb-liveness-evaluate-over-n-frames	10	The number of frames over which RGB liveness detection should be evaluated. This property is not supported by SAFR Inside.
detector.rgb-liveness-fake-threshold	0.33	Specifies how difficult it will be for a subject to be verified as NOTLIVE_CONFIRMED. This property is not supported by SAFR Inside.
detector.rgb-liveness-minimum-center-pose-quality	0.3	The minimum face center pose quality for RGB liveness detection to be used. This property is not supported by SAFR Inside.

Property	Default Value	Description
detector.rgb-liveness-minimum-confirmed-percent	0.9	The percentage of frames that must meet the liveness or fake threshold for the subject to be declared either LIVENESS_CONFIRMED or NOTLIVE_CONFIRMED. This property is not supported by SAFR Inside.
detector.rgb-liveness-minimum-face-context-percent	1.0	The minimum required extra context around faces for the *Context Model* of RGB liveness detection to be used. This property is not supported by SAFR Inside.
detector.rgb-liveness-minimum-face-contrast-quality	0.45	The minimum face contrast quality for RGB liveness detection to be used. This property is not supported by SAFR Inside.
detector.rgb-liveness-minimum-face-sharpness-quality	0.45	The minimum face sharpness quality for RGB liveness detection to be used. This property is not supported by SAFR Inside.
detector.rgb-liveness-minimum-face-size	150	The minimum required height and width of a face, in number of pixels, for the *Texture Model* of RGB liveness detection to be used. This property is not supported by SAFR Inside.
detector.rgb-liveness-minimum-preliminary-threshold	0.5	For multimodal *detection schemes*, this is the liveness threshold which the first evaluated model (the *Texture Model*) must exceed before SAFR will bother evaluating the second model. If this threshold is not met, SAFR immediately returns NOTLIVE_CONFIRMED for the subject. This property is not supported by SAFR Inside.
detector.rgb-liveness-threshold	0.6	Specifies how difficult it will be for a subject to be verified as LIVENESS_CONFIRMED. This property is not supported by SAFR Inside.
directory	N/A	Directory name.
enabled	FALSE	Enables or disables the feed.

Property	Default Value	Description
input.back-channel.mobotix.cash-point	“None”	When the connected camera is a Mobotix camera, this property must be set to the configured cash point within the Mobotix app for the back-channel to work.
input.back-channel.type	“None”	When the connected camera is a Mobotix camera, you can set this property to “Mobotix MX” in order to have SAFR report STRANGER and RECOGNIZED event types to the camera. This feature is necessary if you want to make use of the Mobotix app. If the connected camera isn’t a Mobotix camera, this property doesn’t have any effect.
input.contrast-enhancement.detection-only	FALSE	If true then contrast enhancement is applied to the image which is handed off to the face detector only. If false then contrast enhancement is applied to the video frame as delivered by the camera. Consequently the contrast enhancement effect is visible in the video preview if this option is off but not if it is on. This property is not supported by SAFR Inside.
input.contrast-enhancement.enabled	FALSE	Enables contrast enhancement of the input video frame. This property is not supported by SAFR Inside.
input.contrast-enhancement.exposure-boost	0	The boost for contrast enhancement. This property is not supported by SAFR Inside.
input.contrast-enhancement.low-light-threshold	0.02	Low-light-threshold for contrast enhancement. This property is not supported by SAFR Inside.
input.crop-rectangle.enabled	FALSE	When this is true the defined crop rectangle is used for the camera feed. The crop rectangle is specified in a normalized coordinate system, which means the rectangle is (0, 0) x (1, 1). This property is not supported by SAFR Inside.

Property	Default Value	Description
input.crop-rectangle.height	1	The normalized height value relative to the video of how big the crop rectangle size should be. This property is not supported by SAFR Inside.
input.crop-rectangle.left	0	The normalized left coordinate relative to the video of where the crop rectangle origin should be. This property is not supported by SAFR Inside.
input.crop-rectangle.top	0	The normalized top coordinate relative to the video of where the crop rectangle origin should be. This property is not supported by SAFR Inside.
input.crop-rectangle.width	1	The normalized width value relative to the video of how big the crop rectangle size should be. This property is not supported by SAFR Inside.
input.lens-correction.enabled	FALSE	Enables lens correction for the camera. This property is not supported by SAFR Inside.
input.lens-correction.k1	0	The "k1" lens correction factor. This property is not supported by SAFR Inside.
input.lens-correction.k2	0	The "k2" lens correction factor. This property is not supported by SAFR Inside.
input.loop	FALSE	Enables looping of the feed input. Only video file-based feeds support looping. Ignored when **input.type** is set to "stream". (e.g. when a camera feed is being processed). This property is not supported by SAFR Inside.
input.mirroring.enabled	FALSE	Whether the video image should be mirrored before detection and recognition operations are executed. This property is not supported by SAFR Inside.
input.password	N/A	Password of the person accessing the video stream. This property is not supported by SAFR Inside.
input.rotation.angle	0	Whether the video should be rotated before detection and recognition operations are executed. Valid values are 0, 90, 180, and 270. This property is not supported by SAFR Inside.

Property	Default Value	Description
input.stream.id	N/A	Identifier used to connect to a stream if the URL is blank. This property is not supported by SAFR Inside.
input.stream.name	N/A	A friendly name used for display purposes. This property is not supported by SAFR Inside.
input.stream.rtsp.transport	“udp”	The transport protocol that should be used while accessing the RTSP video stream. Must be one of “udp”, “tcp”, or “udp-multicast”. This property is not supported by SAFR Inside.
input.stream.url	N/A	The video stream URL. The URL must point to a RTSP, HTTP, or FILE stream. Note that if you want to point the input stream to a locally saved file on a Windows machine, VIRGO expects a Windows native path for this property. (e.g. `C:\ProgramData\RealNetworks\SAFR\bin\fo`
input.type	“stream”	The type of feed input; either “stream” or “file”.
input.user-name	N/A	Username of the person accessing the video stream. This property is not supported by SAFR Inside.
input.video-clock.enabled	TRUE	Enables enforcement of the video clock. Video files will be processed as fast as possible if the video clock is turned off. Streams are always processed as fast as possible so this value is ignored when **input.type** is set to “stream”. Live streams play at the rate the data is sent, while file streams will be processed as fast as possible. When **input.type** is set to “file”, then this value is used to determine whether the file should be processed at real time or faster than real time. This property is not supported by SAFR Inside.
mode	“Enrolled and Stranger Monitoring”	Specifies which video processing mode the feed is using.
recognizer.3d-liveness-threshold	0.6	Windows only. Specifies the 3d liveness threshold. This property is not supported by SAFR Inside.

Property	Default Value	Description
recognizer.detect-3d-liveness	FALSE	Windows only. Enables 3D liveness. 3D liveness is a special feature of certain Intel RealSense camera models that allows them to distinguish flat images from 3 dimensional ones, thus allowing SAFR to tell the difference between a real face and a photo. This feature only works with Intel RealSense D415 and D435 cameras; if you don't have any cameras of those types connected to SAFR, then this feature will not work. This property is not supported by SAFR Inside.
recognizer.detect-age	FALSE	Enables the detection of age information.
recognizer.detect-gender	FALSE	Enables the detection of gender information.
recognizer.detect-identity	TRUE	Enables detection of an identity, which matches against the existing database of people (identities).
recognizer.detect-mask	FALSE	When enabled, SAFR will evaluate all occluded faces to see if they're covered by a mask. If they are, then SAFR will use the mask enhanced model to attempt to recognize the face behind the mask. If the occluded face isn't covered by a mask, then the normal occluded model will be used instead.
recognizer.detect-mask-model	"precise"	Specifies the model to be used for mask detection. There are 3 possible values: **Precise**: This model produces the least number of false positives (i.e. detecting that a person is wearing a mask but there is no mask), but it suffers from the lowest true positive rate. (i.e. detecting masks that are actually there) **Sensitive**: This model produces the highest true positive rate, but it suffers from the highest number of false positives. **Normal**: This model produces a moderate amount of both false positives and true positives.

Property	Default Value	Description
recognizer.detect-mask-threshold	0.5	Specifies the threshold at and above which mask detection will conclude that *mask=true*.
recognizer.detect-occlusion	FALSE	Enables occlusion detection during recognition.
recognizer.detect-pose-action	FALSE	Enables the pose liveness action recognizer. This property is not supported by SAFR Inside.
recognizer.detect-rgb-action	FALSE	Enables the RGB liveness recognizer. When enabled, this recognizer generates events based on the RGB liveness feature when cameras view somebody enrolled in your SAFR Identity Database. This property is not supported by SAFR Inside.
recognizer.detect-sentiment	FALSE	Enables the detection of sentiment information.
recognizer.detect-smile-action	FALSE	Enables the smile action recognizer. This property is not supported by SAFR Inside.
recognizer.identity-masked-threshold-offset	0	Sets the identity threshold when detecting masks.
recognizer.identity-proximity-threshold-allowance	0.13	A boost value that is added to the Identity Recognition Threshold.
recognizer.identity-recognition-threshold	0.54	Identity recognition threshold.
recognizer.learning-enabled	FALSE	Enables the feed to learn new identities.
recognizer.learn-occluded-faces	FALSE	Enables learning of occluded faces regardless of the maximum occlusion setting. If this is true then the server configuration will be used, which by default doesn't do any occlusion detection.
recognizer.mask-check-detection-edge-threshold	0.03	How far a face must be from the edge of the screen before a mask event detection is attempted. For example, if a face is 100 pixels, and **recognizer.mask-check-detection-edge-threshold** is set to .03 (i.e. 3%), then the face must be 3 pixels from the edge of the screen before SAFR will attempt a mask event detection. This property is not supported by SAFR Inside.

Property	Default Value	Description
recognizer.mask-check-enabled	FALSE	Enables the detection of mask event types. Mask event detection attempts can return 3 potential results: *mask=false*, *mask=indeterminate*, or *mask=true*. After the configured number of consecutive mask event detection results, the mask event state is set to the appropriate value. The mask event state can only progress from false towards true; the state never regresses back towards false. For example, once the mask event state for a viewed person becomes set to *mask=true*, then that person's mask event state won't ever regress to mask=indeterminate or *mask=false*. Events are generated when the mask event state is set to either mask=false or *mask=true*.
recognizer.mask-check-min-consecutive-mask-detections	1	Specifies the minimum number of consecutive *mask=true* mask detection results that must occur before SAFR will generate a *mask=true* event. See the **recognizer.mask-check-enabled** property for more information.
recognizer.mask-check-min-consecutive-no-mask-detections	2	Specifies the minimum number of consecutive *mask=false* mask detection results that must occur before SAFR will generate a *mask=false* event. See the **recognizer.mask-check-enabled** property for more information.
recognizer.mask-check-min-consecutive-occluded-no-mask-detections	2	Specifies the minimum number of consecutive *mask=indeterminate* mask detection results that must occur before SAFR will set the mask event state to *mask=indeterminate*. See the **recognizer.mask-check-enabled** property for more information.
recognizer.mask-check-min-mask-detection-size	70	The smallest face size, in pixels, upon which SAFR will attempt to detect a mask event.

Property	Default Value	Description
recognizer.maximum-clip-ratio	0.1	The maximum clip ratio on either side the recognition candidate might have.
recognizer.maximum-clip-ratio-identification	0	The maximum clip ratio on either side the insertion candidate might have.
recognizer.maximum-concurrent-recognitions	5	The maximum number of concurrent recognitions to allow. 0 means to automatically set this. This property is not supported by SAFR Inside.
recognizer.maximum-occlusion	0	The maximum occlusion value that is allowed when adding a new candidate images into the Person Directory. If the face is occluded with a value greater than this then the face will not be added, but if it's less than or equal to this value then it will be added.
recognizer.maximum-pitch-identification	0.4	The maximum pitch value used to determine if the face is looking straight ahead. The pitch value is the forward/backward movement of the face.
recognizer.maximum-roll-identification	0.15	The maximum roll value used to determine if the face is looking straight ahead. The roll value is the side to side tilt movement of the face.
recognizer.maximum-yaw-identification	0.4	The maximum yaw value used to determine if the face is looking straight ahead. The yaw value is the side to side movement of the face.
recognizer.minimum-center-pose-quality	0.05	The minimum center pose quality that a recognition candidate must have in order to allow the addition of the candidate image into the Person Directory.

Property	Default Value	Description
recognizer.minimum-center-pose-direct-gaze	0.70	If a face's center pose quality is above this value, the face is determined to be gazing directly at the camera, but if the center pose quality is below this value, the face is determined to be turned away. The longer the face gazes directly at the camera, the longer events' directGazeDuration is. This property is not supported by SAFR Inside.
recognizer.minimum-center-pose-quality-identification	0.45	The minimum center pose quality that a recognition candidate must have in order to allow the addition of the candidate image into the Person Directory.
recognizer.minimum-center-pose-quality-merging	0.59	The minimum center pose quality that a recognition candidate must have in order to allow merging.
recognizer.minimum-face-contrast-quality	0.1	The minimum face contrast quality that a face image must have before recognition is attempted.
recognizer.minimum-face-contrast-quality-identification	0.3	The minimum face contrast quality that a recognition candidate must have in order to allow the addition of the candidate image into the Person Directory.
recognizer.minimum-face-contrast-quality-merging	0.45	The minimum face contrast quality that a recognition candidate must have in order to allow merging.
recognizer.minimum-face-sharpness-quality	0.1	The minimum face sharpness quality that a face image must have before recognition is attempted.
recognizer.minimum-face-sharpness-quality-identification	0.3	The minimum face sharpness quality that a recognition candidate must have in order to allow the addition of the candidate image into the Person Directory.
recognizer.minimum-face-sharpness-quality-merging	0.45	The minimum face sharpness quality that a recognition candidate must have in order to allow merging.

Property	Default Value	Description
recognizer.minimum-face-size	80	The minimum size of faces to detect. This value is applied after searching the image.
recognizer.minimum-face-size-identification	120	The minimum resolution that a recognition candidate image must have in order to allow the addition of the candidate image into the Person Directory.
recognizer.minimum-face-size-merging	220	The minimum resolution a recognition candidate must have in order to allow merging.
recognizer.pose-action-max-cpq-jump-after-discontinuity	0.15	The maximum change between samples while the pose is changing from center to profile if lingering. This property is not supported by SAFR Inside.
recognizer.pose-action-max-cpq-jump-in-continuity	0.18	The maximum change between samples while the pose is changing from center to profile. This property is not supported by SAFR Inside.
recognizer.pose-action-max-profile-confidence-end	0.60	The maximum profile pose confidence to allow during the final profile pose detection phase. This property is not supported by SAFR Inside.
recognizer.pose-action-max-profile-pose-quality	0.26	The maximum center pose quality to use when detecting the final profile pose. This property is not supported by SAFR Inside.
recognizer.pose-action-max-profile-pose-roll	0.3	The maximum roll threshold in either direction in which the face can rotate when determining whether the face is in profile pose. This property is not supported by SAFR Inside.
recognizer.pose-action-min-center-pose-quality	0.5	The minimum center pose quality to use when detecting the initial center pose. This property is not supported by SAFR Inside.
recognizer.pose-action-min-detections-per-second	15	The minimum number of frames per second that is required during the process. This property is not supported by SAFR Inside.
recognizer.pose-action-min-profile-confidence-start	0.35	The minimum profile pose confidence to allow during the initial center pose detection phase. This property is not supported by SAFR Inside.

Property	Default Value	Description
recognizer.pose-action-min-profile-pose-yaw	0.81	The minimum profile pose yaw value that is required during the final profile pose detection phase. This property is not supported by SAFR Inside.
recognizer.pose-action-min-profile-similarity	0.86	The minimum similarity score required when verifying the final profile pose. This property is not supported by SAFR Inside.
recognizer.pose-action-min-transition-poses	2	The minimum number of required center pose samples during the transition from center to profile pose. This property is not supported by SAFR Inside.
recognizer.pose-action-profile-pose-required-confirmations	1	The number of consecutive confirmations required to enter the final profile pose detection phase. This property is not supported by SAFR Inside.
recognizer.pose-action-required-confirmations	3	The number of consecutive confirmations required to enter the initial center pose detection phase. This property is not supported by SAFR Inside.
recognizer.pose-configuration-identification-enabled	FALSE	If this is true then pose configuration is enabled for identification. The pose configuration allows for replacing center pose quality with advanced parameters such as yaw, pitch and roll. When pose configuration is enabled, then recognizer.minimum-center-pose-quality is ignored and the following 3 properties are used instead: recognizer.maximum-yaw-identification, recognizer.maximum-pitch-identification, and recognizer.maximum-roll-identification.
recognizer.rgb-action-identity-threshold-boost	0.0	The amount to temporarily boost identity recognition attempts during RGB liveness actions. This property is not supported by SAFR Inside.
recognizer.rgb-action-min-recognitions-fake	2	The minimum number of consecutive “fake” recognitions that are required in order to consider the face fake. This property is not supported by SAFR Inside.

Property	Default Value	Description
recognizer.rgb-action-min-recognitions-live	2	The minimum number of consecutive “live” recognitions that are required in order to consider the face live. This property is not supported by SAFR Inside.
recognizer.smile-detect-rgb-liveness	FALSE	Enables RGB liveness detection for the smile action recognizer.
recognizer.smile-rgb-action-min-recognitions-fake	2	The minimum number of consecutive “fake” recognitions that are required in order to consider the face fake.
recognizer.smile-rgb-action-min-recognitions-live	2	The minimum number of consecutive “live” recognitions that are required in order to consider the face live.
recognizer.smile-duration	0	The amount of time that the smile should last. This property is not supported by SAFR Inside.
recognizer.smile-identity-threshold-boost	0.13	The smile threshold to boost temporarily during the smile action. This property is not supported by SAFR Inside.
recognizer.smile-pre-delay	100	The amount of time that there should be no smile. This property is not supported by SAFR Inside.
recognizer.smile-thresholds-enabled	FALSE	Enables the smile threshold values. This property is not supported by SAFR Inside.
recognizer.smile-threshold-neutral	-0.1	The threshold in which there is no smile. This property is not supported by SAFR Inside.
recognizer.smile-threshold-smiling	0.7	The threshold in which there is a smile. This property is not supported by SAFR Inside.
reporter.delay	0	Delay the event reporting to the server by this amount in seconds.
reporter.enabled	TRUE	Enables or disables event reporting.

Property	Default Value	Description
reporter.events-date-timestamps-enabled	TRUE	When a video file is being processed, this property is used to determine whether the events should use date timestamps relative to an initial start date or video timestamps relative to the start of the video. If this setting is `FALSE`, video timestamps will be used. In this case the timestamps are relative to the start of the video, which is typically 0. It also sets the context to “media” when posting the event so that the time will properly be interpreted as a timestamp rather than a date. If this setting is set to `TRUE`, then date timestamps will be used relative to the **reporter.events-initial-date-offset**. If **reporter.events-initial-date-offset** is null then the current system date will be used when the video processing starts. In this case, the context is set to “live” when posting events so that the time will properly be interpreted as a date. There are cases when videos are processed where users are interested in video timestamps. (e.g. a video indexing feature) In these cases, this setting should be set to `FALSE`. The more common use case now is that users are interested in processing video files in real time or processing security videos from specific times that it is necessary to use date timestamps relative to a specific time instead of video timestamps relative to the start of the video. The reporter.events-initial-date-offset option is ignored when this is set to false. This property is not supported by SAFR Inside.

Property	Default Value	Description
reporter.events-initial-date-offset	0	When processing a video file for events this value can be used to set the initial date offset to use for the events being processed. By default video events start with current system time. This option is ignored if the **reporter.events-date-timestamps-enabled** option is set to `FALSE` because video timestamps will be used instead.
reporter.minimum-event-duration-identified	0	The minimum allowed recognized person event duration in seconds. Events shorter than this duration will not be reported.
reporter.minimum-event-duration-stranger	0	The minimum allowed stranger event duration in seconds. Events shorter than this duration will not be reported.
reporter.minimum-event-duration-unidentified	1500	The minimum allowed unrecognizable person event duration in seconds. Events shorter than this duration will not be reported.
reporter.report-event-face	TRUE	Enables the inclusion of face thumbnails in event reports.
reporter.report-event-scene	FALSE	Enables the inclusion of scene images in event reports.
reporter.report-secondary-events	FALSE	Reports secondary events. Secondary events are events that are associated with a primary event via the rootEventId property in the event. It is usually preferred to only report the primary events and the secondary events need to only be reported if there is more detail needed. If this is disabled then all events with a rootEventId property set to a primary event will not be reported. Only events with rootEventId not set to anything will be reported, which are the primary events.
reporter.report-speculated-events	TRUE	Reports events for speculated faces. Speculated faces are faces that aren't a 100% match, but are close.

Property	Default Value	Description
reporter.report-stranger-events	TRUE	Reports events for people that are strangers. These are people not registered by the system after running facial recognition on the face.
reporter.report-unrecognizable-events	TRUE	Reports events for people that are not recognizable.
reporter.stranger-events.only-if-occluded	FALSE	Specifies whether only occluded stranger events should be reported. By default stranger events are only generated if the face is not occluded, if occlusion detection is enabled, otherwise they are generated when the face meets the identification image quality metrics. If this option is set to true then stranger events will be reported only if the face is occluded.
reporter.stranger-maximum-age	0	The maximum age of strangers that will trigger stranger events. If a stranger older than the specified maximum age is detected, no stranger event is generated.
reporter.stranger-minimum-age	0	The minimum age of strangers that will trigger stranger events. If a stranger younger than the specified minimum age is detected, no stranger event is generated.
reporter.update-images	FALSE	Updates the thumbnail images with higher quality images during the course of the event if possible.
reporter.update-in-progress-event-interval	1000	When **reporter.update-in-progress-event-properties** is set to `TRUE`, this property specifies the time interval in which to update event properties that change. When **reporter.update-in-progress-event-properties** is set to `FALSE`, this property has no effect.

Property	Default Value	Description
reporter.update-in-progress-event-properties	FALSE	If this is enabled, then any event properties that change will be updated at the specified **reporter.update-in-progress-event-interval**. Many properties do change periodically. (e.g. averages that are continually computed)
site	N/A	Site name, if any.
source	N/A	Source name.
statistics.enabled	FALSE	Specifies whether VIRGO should record and report statistics for this feed.
tracker.detect-direction-of-travel.person.bottom-boundary	0	The percentage of the bottom side of the camera view field to exclude from direction of travel event reporting. This property is not supported by SAFR Inside.
tracker.detect-direction-of-travel.person.down	FALSE	Enables direction of travel detection in the downward direction. This property is not supported by SAFR Inside.
tracker.detect-direction-of-travel.person.down-distance	0.1	The percentage of the camera view that a tracked person can travel in a downward direction before a direction of travel event is generated. This property doesn't have any effect if **tracker.detect-direction-of-travel.person.down** is set to FALSE. This property is not supported by SAFR Inside.
tracker.detect-direction-of-travel.person.left	FALSE	Enables direction of travel detection in the leftward direction. This property is not supported by SAFR Inside.
tracker.detect-direction-of-travel.person.left-boundary	0	The percentage of the left side of the camera view field to exclude from direction of travel event reporting. This property is not supported by SAFR Inside.
tracker.detect-direction-of-travel.person.left-distance	0.1	The percentage of the camera view that a tracked person can travel in a leftward direction before a direction of travel event is generated. This property doesn't have any effect if **tracker.detect-direction-of-travel.person.left** is set to FALSE. This property is not supported by SAFR Inside.

Property	Default Value	Description
tracker.detect-direction-of-travel.person.right	FALSE	Enables direction of travel detection in the rightward direction. This property is not supported by SAFR Inside.
tracker.detect-direction-of-travel.person.right-boundary	0	The percentage of the right side of the camera view field to exclude from direction of travel event reporting. This property is not supported by SAFR Inside.
tracker.detect-direction-of-travel.person.right-distance	0.1	The percentage of the camera view that a tracked person can travel in a rightward direction before a direction of travel event is generated. This property doesn't have any effect if **tracker.detect-direction-of-travel.person.right** is set to FALSE. This property is not supported by SAFR Inside.
tracker.detect-direction-of-travel.person.top-boundary	0	The percentage of the top side of the camera view field to exclude from direction of travel event reporting. This property is not supported by SAFR Inside.
tracker.detect-direction-of-travel.person.up	FALSE	Enables direction of travel detection in the upward direction. This property is not supported by SAFR Inside.
tracker.detect-direction-of-travel.person.up-distance	0.1	The percentage of the camera view that a tracked person can travel in an upward direction before a direction of travel event is generated. This property doesn't have any effect if **tracker.detect-direction-of-travel.person.up** is set to FALSE. This property is not supported by SAFR Inside.
tracker.enable-face-bounds-prediction	TRUE	Enables face bounds prediction, which predicts which direction the face is moving to maintain tracking.
tracker.enable-face-size-correlation	TRUE	Enables face correlation of tracked faces, which compares detected faces looking for a change in area.

Property	Default Value	Description
tracker.enable-high-precision	FALSE	Enables high precision tracking, which decreases event fragmentation and increases the stickiness of SAFR's tracking algorithm at the cost of computer processing power. This property should be enabled if you are experiencing duplicate or missing Direction of Travel events. See Camera Preferences for information about the Direction of Travel feature. This property is not supported by SAFR Inside.
tracker.failed-recognition-back-off-interval	340	After making the initial recognition attempts as quickly as possible, back up the amount specified by this setting for each subsequent recognition. This continues until the retry interval is reached.
tracker.failed-recognition-retry-interval	0	The interval in which to run recognition requests if the face has not been recognized.
tracker.identity-relearn-interval-days	0	Updates the identity only when the currently saved identity is older than the updated identity.
tracker.identity-update-better-image	FALSE	Updates the identity when the currently saved identity is of lower quality (in all aspects) than the new image.
tracker.initial-recognition-attempts	3	The number of initial recognition attempts to make on an unrecognized person as fast as possible.
tracker.maximum-linger-frames	30	Determines for how many frames more we continue to keep a tracked face around after we have failed to detect it in the most recent frame. This makes the tracker resilient against intermittent loss of face.
tracker.max-position-change-relative-to-face	115	The maximum position change, specified in percentage relative to the face size, to continue tracking.
tracker.max-size-change-relative-to-face	50	The maximum size change, specified in percentage relative to the object size, to continue tracking.

Property	Default Value	Description
tracker.min-failed-recognitions-to-stop-tracking-identity	3	When a face is being tracked recognitions are continually confirming the identity. The identity is also being verified if it is transferred from a person object. In these cases, if the recognition or verification fails this number of consecutive times then the identity will be reset and no longer associated with the face because we are no longer sure it is the same identity.
tracker.minimum-number-identical-recognitions-learn	2	The number of consecutive recognitions that need to occur before adding a new identity to the system.
tracker.minimum-number-identical-recognitions-lock	1	The number of consecutive recognition attempts that we must run and produce the same person identity before we lock onto this identity.
tracker.minimum-required-consecutive-badge-detections	0	The number of consecutive detections that are required before reporting that the object (based on object id) was actually detected. This property can be used to filter out false positives. This property is not supported by SAFR Inside.
tracker.minimum-required-consecutive-mask-detections	1	The number of consecutive detections that are required before reporting that the masked person was detected. This property can be used to filter out false positives. This property is not supported by SAFR Inside.
tracker.reconfirmation-interval	1000	Identity reconfirmation time interval in ms.
tracker.stop-tracking-on-failed-re-recognition	FALSE	If recognition fails when re-recognizing a person then delete the identity that was created.

40.4 Update Properties

The update properties are not intended for public consumption at this time.

Property	Default Value	Description
client-type	OS-defined client type	Internal use only.

Property	Default Value	Description
version	N/A	Internal use only.
enabled	FALSE	Internal use only.
progress-status	N/A	Internal use only.
progress-url	N/A	Internal use only.
download-url	N/A	Internal use only.
log-enabled	FALSE	Internal use only.
progress-interval	1000	Internal use only.

41 Processing Video Files

Both the Video Feeds Window (located in the Desktop Client and the Web Console) and the Command Line Interface can be used to process video files for person detection and/or recognition. This can be useful to recognize faces for purposes of identifying people or generating events on pre-recorded videos. For example, consider the scenario where you have video footage from cameras throughout a facility and you want to determine where and when a person of interest appears in those videos. To do this, you would register the people of interest to the Person Directory (if they weren't already registered) and you could then process the videos in Enrolled Monitoring video processing mode in order to identify when and where that individual appears.

The Video Feeds Window is the best video file processing option when you only have 1 video to process. When you want to process 2 or more video files, the Command Line Interface is the better choice.

Note When you switch control from the Video Feeds Window to the Command Line Interface, any feeds that you configured in the Video Feeds Window will be stopped and will not be visible from the Command Line Interface. The feeds still exist in the Video Feeds Window, but you cannot start or manage those feeds from the Command Line Interface. You must re-create new feeds for the Command Line Interface.

41.1 Process Files with the Video Feeds Window

To process video files, the Video Feeds Window must have access to those files on the local file system. The window can work with files on an internal hard drive, an external drive attached to the machine, or a network share mounted to the file system. In this example we will use the following directories.

```
mkdir -p /files/videos
mkdir -p /files/feeds
```

- The `videos` directory will be used to store the video files that VIRGO will process.
- The `feeds` directory will be used to store feed configuration files used by VIRGO. Feed configuration files store information necessary to process video from a file or camera.
- `videos` and `feeds` can be either aliases or directories mounted to an external file system.

41.1.1 Mount Host Filesystem to VIRGO Container

In order to process files that are on a Linux machine, you must mount a file system on the host drive to the Docker container running Video Recognition Gateway (VIRGO).

You can mount a host drive onto the Docker instance by doing the following:

1. Add the following lines to the end of the file "/opt/RealNetworks/SAFR/virgo/app/docker-compose.yml"

   ```
   vi /opt/RealNetworks/SAFR/virgo/app/docker-compose.yml

   - /files:/files
   ```

   The hyphen should be included.
   The resulting file should look something like this. (DON'T USE THE EXAMPLE BELOW - Your system will have different version info.)

   ```
   version: "3.6"
   services:
     virgo:
       image: safr_virgo:1.2.22
       container_name: safr_virgo
       restart: on-failure
       pid: "host"
       volumes:
   ```

```
      - /opt/RealNetworks/SAFR/virgo/config/:/etc/virgo
      - /files:/files
```

- The mount format is `<path on host>`:`<path on VIRGO Docker container>`
  - This is equivalent to creating an alias on the Docker container pointing to a actual directory on the host.
- `<path on VIRGO Docker container>` can be any path. It should not exist already.
- `<path on host>` should be a real directory. It can be located anywhere on the host drive.
- Using the same path for both makes it less confusing when typing filepaths because there will be no need to remember the Docker path and to append the relative host path. Just use the fully qualified host path when defining the location of a video in the VIRGO config.

2. Run the following commands to re-mount all paths on the Docker container:

```
docker-compose -f /opt/RealNetworks/SAFR/virgo/app/docker-compose.yml down
docker-compose -f /opt/RealNetworks/SAFR/virgo/app/docker-compose.yml up
   -d
```

3. Check to see if the folder is mounted correctly in the Docker container by doing the following:

   1. Run the following command to sign into the Docker container:

   ```
   docker exec -it safr_virgo bash
   ```

   2. Try to list the directory of the location mounted above:

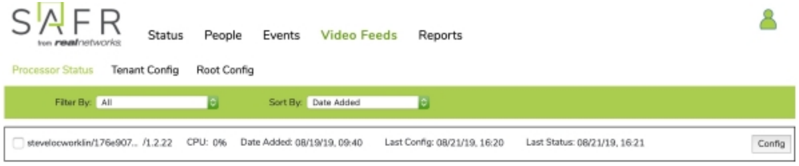
   ```
   ls /files
   ```

   You should see the two folders (feeds and videos) that you created listed under this location.

41.1.2 Use the Video Feeds Window

To use the Video Feeds Window, do the following:

1. Connect to the Web Console or open the Desktop Client. See here for details about the Web Console and how to connect to it.
2. Navigate to the **Video Feeds** tab and select **Processor Status**. Assuming you have no other video clients connected to the server (i.e. from additional connected VIRGO feeds or Desktop Clients), you will see a single entry representing VIRGO with no feeds.

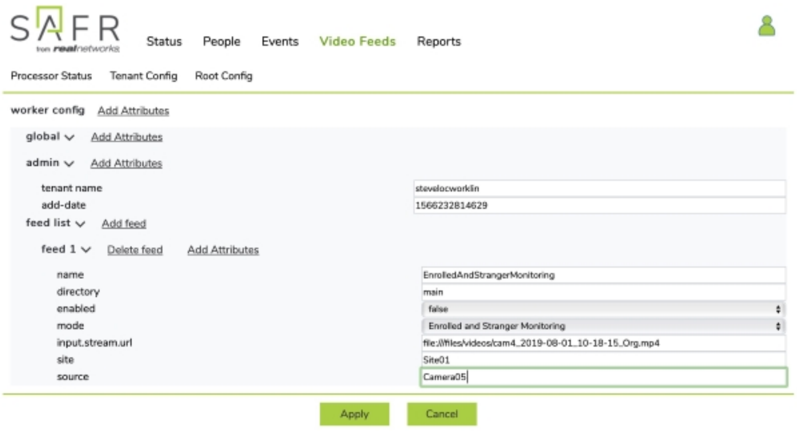


Filter By: All

Sort By: Date Added

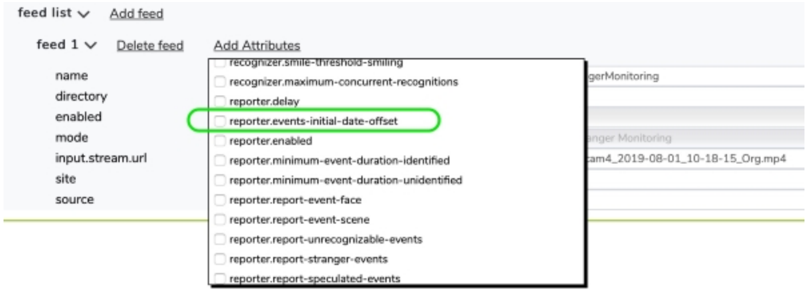
□ stevelocworklin/176e907... /1.2.22 CPU: 0% Date Added: 08/19/19, 09:40 Last Config: 08/21/19, 16:20 Last Status: 08/21/19, 16:21

3. Click **Config** to add feeds.
4. Enter values for the following fields:
   - **name** - User-defined name. We recommended that you only use ASCII letters and numbers (i.e. no spaces) since this is used to reference the feed on the command line.
   - **mode** - Defines the default settings for SAFR detection and recognition.
   - **input.stream.url** - Source camera URL or filename of the video to use as input. In this case we pre-pend the value with "file://" to indicate that we're processing a file. The path portion is the path to the video file relative to the Docker container, as explained above.
   - In the screenshot below, the `enabled` field still has its default value of "false". You should set it to "true" when you are ready to start processing the file.

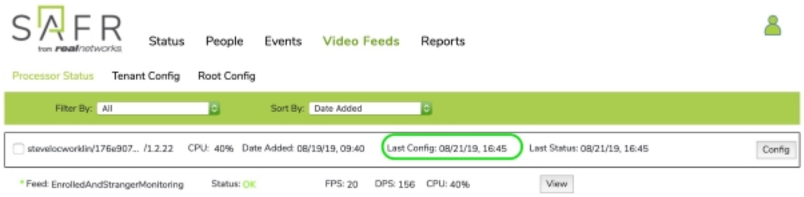


tenant name: stevelocworklin
add-date: 1566232814629
name: EnrolledAndStrangerMonitoring
directory: main
enabled: false
mode: Enrolled and Stranger Monitoring
input.stream.url: file:///files/videos/cam4_2019-08-01_10-18-15_Org.mp4
site: Site01
source: Camera05

5. Before clicking **Apply**, add an additional field to specify the start time of the video. Do this by doing the following:
    1. Click **Add Attributes** under the feed.
    2. Search for and select the “reporter.events-initial-date-offset” property.


feed list
Add feed
feed 1
Delete feed
Add Attributes
name
directory
enabled
mode
input.stream.url
site
source
recognizer.smile-threshold-smiling
recognizer.maximum-concurrent-recognitions
reporter.delay
reporter.events-initial-date-offset
reporter.enabled
reporter.minimum-event-duration-identified
reporter.minimum-event-duration-unidentified
reporter.report-event-face
reporter.report-event-scene
reporter.report-unrecognizable-events
reporter.report-stranger-events
reporter.report-speculated-events
gerMonitoring
nger Monitoring
am4_2019-08-01_10-18-15_Org.mp4

6. Once added set a date value as Epoch in milliseconds. For example, for Aug 1, 2019 at 11 AM, the Epoch value is 1564682400000.
    - This causes generated events to be recorded at the correct point in time rather than being recorded at the default Epoch value of “0” or January 1, 1970.
    - Try the Chrome Extension ‘utime’ to create Epoch times. This extension allows you to type any date/time and get Epoch values as well as natural language strings such as ‘1 hour ago’.
      Note: Be sure to set the time format to “milliseconds” or else the time will be off by many years.
7. Change the `enabled` field to “true” and click **Apply** to save the feed and start processing the video.
8. If all goes well you should see the following:



Filter By: All   Sort By: Date Added

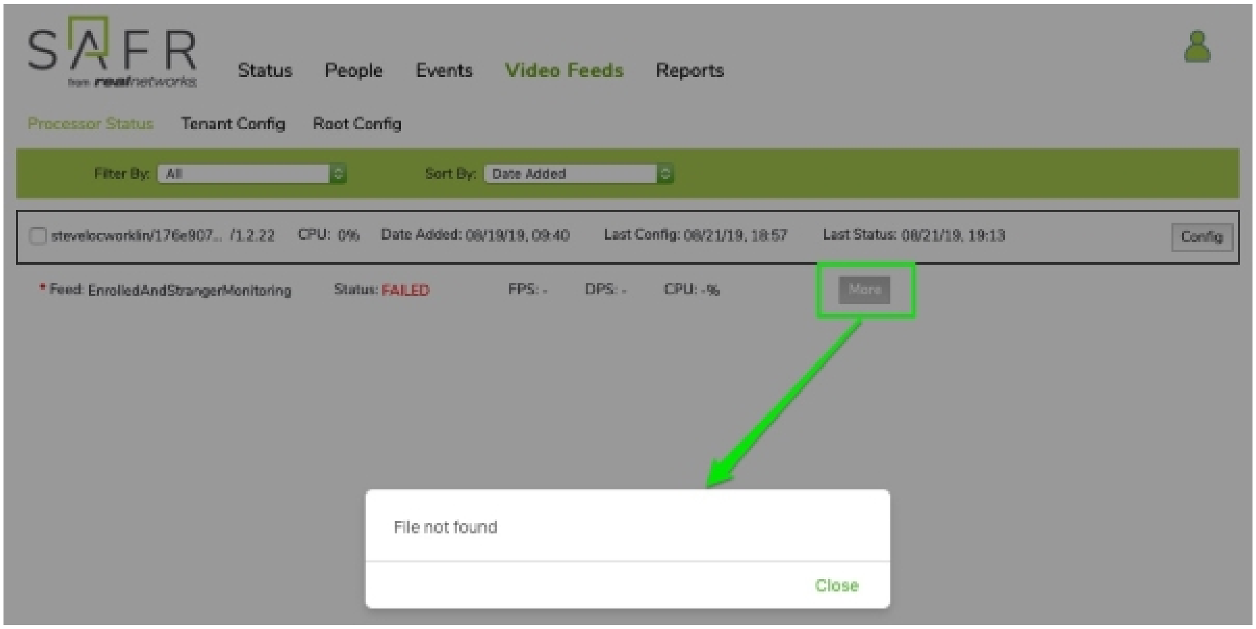
stevelocworklin/176e907... /1.2.22   CPU: 40%   Date Added: 08/19/19, 09:40   Last Config: 08/21/19, 16:45   Last Status: 08/21/19, 16:45

* Feed: EnrolledAndStrangerMonitoring   Status: OK   FPS: 20   DPS: 156   CPU: 40%

9. For a few moments you may see the `Last Config` date in red which means that changes have not yet been applied to VIRGO. Once the changes have been applied, VIRGO should start processing and you should see the `Status` reported as “ok”. You can click **View** to see the current frame of video being processed. This will be updated every second or so.
10. See Troubleshooting Feeds below if the video does not start processing successfully.
11. You can move to the **Events** tab to view the events being processed. You may need to change the search criteria to include August 1, 2019 for the events to show up.

41.1.3 Troubleshoot Feeds

If there is a problem, first check the **More** button on the VIRGO feed as shown below:



Filter By: All

Sort By: Date Added

stevelocworklin/176e907... /1.2.22 CPU: 0% Date Added: 08/19/19, 09:40 Last Config: 08/21/19, 18:57 Last Status: 08/21/19, 19:13

* Feed: EnrolledAndStrangerMonitoring Status: FAILED FPS: - DPS: - CPU: -%

File not found

Additional possible troubleshooting steps:

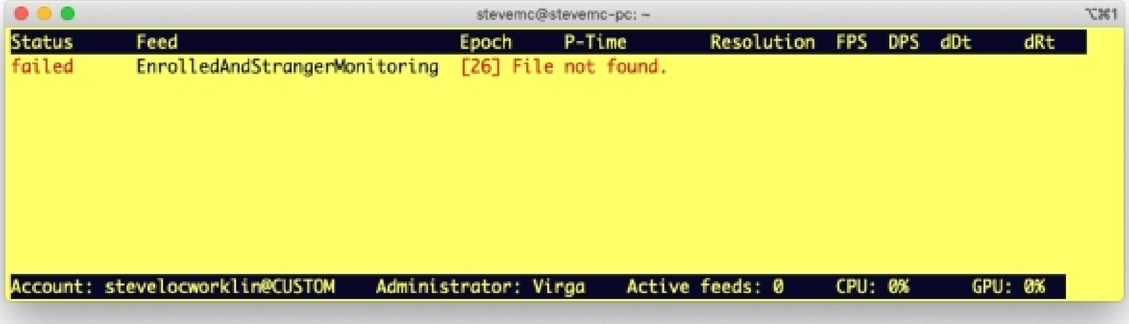
- If you see status as “inactive”, check the enabled option in the queue file. It should be “true”, as in: `"enabled" : true`
- To start the feed, you can run the VIRGO command: `docker exec -it safr_virgo ./virgo feed start queue1`
- To avoid this problem with subsequent files, you may want to edit the feed configuration file to set `enabled` to “true”.

If the steps above do not fix the problem, then it may help to look at the service monitor or service log. Both must be started before running VIRGO in order to capture the error output.

41.1.3.1 View the VIRGO Service Monitor To view the service monitor, log into SAFR Server and run the following command:

```
sudo docker exec -it safr_virgo /opt/RealNetworks/virgo/virgo service
   monitor

or

docker exec -it safr_virgo bash
./virgo service monitor
```

You should see the following:



```
Status      Feed                            Epoch     P-Time        Resolution  FPS  DPS  dDt     dRt
failed      EnrolledAndStrangerMonitoring   [26] File not found.
Account: stevelocworklin@CUSTOM     Administrator: Virga     Active feeds: 0      CPU: 0%      GPU: 0%
```

41.1.3.2 View the VIRGO Service Log

To view the service log, run the following command:

```
sudo docker exec -it safr_virgo /opt/RealNetworks/virgo/virgo service log
    d/feed d/http-cop

or

docker exec -it safr_virgo bash
virgo service log d/feed d/http-cop
```

With the above running, restart the feed. In the Video Feeds Window you may need to do the following to successfully restart the feed:

1. Set the feed `enabled` flag to "false".
2. Save.
3. Change the `enabled` flag back to "true".

Additional Notes:

- Information will be printed to the screen. Use Unix Redirect (>) or "tee -a" command to write output to file.
- Additional options for VIRGO logging can be found here.
- It is useful to have both the service monitor and service log open in separate windows as you start the video feed.

41.2 Process Files with the Command Line Interface

41.2.1 Create a Feed Template

To run VIRGO, you need a feed configuration file. You can use the configuration file we created above in VIRGA as a starting point. This will include most of the common configuration properties needed.

1. Export a feed configuration file from VIRGO to use as a template.

2. Get a list of VIRGO feeds.

   ```
   > sudo docker exec -it safr_virgo ./virgo feed list

   EnrolledAndStrangerMonitoring
   ```

   - EnrolledAndStrangerMonitoring should be the only feed listed.

3. Save that configuration to a file called template.json.

```
> sudo docker exec -it safr_virgo ./virgo feed get
   EnrolledAndStrangerMonitoring

/files/feeds/template.json
```

- The folder /files/feeds here is relative to the VIRGO Docker container. This path is mounted to the same path (/files/feeds) on the host filesystem.

4. Ensure that the file was written.

```
> cd /files/feeds
> ls

template.json
```

5. Edit the file 'template.json' if desired. Generally it's easier to perform editing in the Video Feeds Window before exporting.

   - The Video Feeds Window unnecessarily adds extra escapes to forward slashes in the JSON. This makes the string much harder to read. When using the Video Feeds Window to generate configuration files, the file path string can be simplified as follows:
     - Change this: `file:\/\/\/opt\/RealNetworks\/virgo\/files\/videos\/cam4Aug1_output000.mp4`
     - To this: `file:///opt/RealNetworks/virgo/files/videos/cam4Aug1_output000.mp4`

6. Create a copy of the template file and name it 'queue1.json'. You're making a copy because you'll maintain one file for each feed.

```
> cd /files/feeds
> cp template.json queue1.json
```

7. Update the feed file by setting values for site, source, start time, and filename. Use the utility script *update_virgo_feed.sh* documented below for this purpose.

```
> update_virgo_feed.sh queue1.json Site01 Camera10 1564682400000
   'file:///files/videos/test/vid00.mp4'
```

   - See the script's documentation below for a description of its command line arguments.
   - The file path assumes you have mounted the VIRGO Docker container to /files in the host drive as explained above.

This will set the values for the respective fields in queue1.json. View the file in an editor to confirm edits were made.

41.2.2 Create Feed

You can assign the feed configuration file created in the previous section to a feed. Before doing that, you need to first switch control from the Video Feeds Window to the Command Line Interface. As explained above, the Video Feeds Window and the Command Line Interface do not share the same configuration files. That is why you first exported a configuration file above from the Video Feeds Window. Now that you have a configuration file, you're ready to switch control to the Command Line Interface. The Command Line Interface will start out with no feeds defined. You will then create feeds from the configuration file saved above.

1. Switch control from the Video Feeds Window to the Command Line Interface

```
> sudo docker exec -it safr_virgo ./virgo administrator set virgo
```

Note: This changes control of VIRGO from the Video Feeds Window to the Command Line Interface. The Video Feeds Window will no longer be able to start or stop VIRGO feeds until control is restored back to it. Note that all feeds added to the Video Feeds Window are not available to the Command Line Interface. You will re-create feeds as described below.

2. Create a new feed ‘queue1’ from the queue1.json feed configuration file created above.

```
> sudo docker exec -it safr_virgo ./virgo feed add queue1
   /files/feeds/queue1.json
```

3. Confirm that the feed was added to VIRGO.

```
> sudo docker exec -it safr_virgo ./virgo feed list

queue1
```

4. VIRGO will attempt to start processing the feed right away. Run VIRGO Service Status or VIRGO Service Monitor/

```
> sudo docker exec -it safr_virgo ./virgo service status

queue1 ok
```

See the Service Monitor section above for information on how to run VIRGO Service Monitor. Initially you may see a status reported as “prerolling” which means the feed is starting up.

After processing is done, the status should change from “ok” to “eos” (meaning End of Stream). If any other status is shown, see Troubleshooting Feeds above.

41.2.3 Edit Feed and Reload

To edit a feed and re-load, do the following.

1. Stop the feed with following command:

```
docker exec -it safr_virgo ./virgo feed stop queue1
```

2. Edit the file and correct problems. Re-load the file from disk using following command:

```
docker exec -it safr_virgo ./virgo feed set queue1
   /files/feeds/queue1.json
```

3. Restart the feed with following command:

```
docker exec -it safr_virgo ./virgo feed start queue1
```

41.3 Batch Processing Files with the Command Line

When batch processing multiple files, it’s most efficient to process two or more files in parallel. The optimal number of parallel processes is a function of processor speed and disk/network speeds and should be determined by experimentation on your hardware. VIRGO will try to process files as fast as possible and leverage multiple CPU cores and GPUs but there is generally a limit to how many CPU cores it uses. Processing more files in parallel will allow you to fully utilize the machine resources.

In general the process is as follows:

- N queues are created. Each queue will be able to process 1 file in parallel.
- 1 file is set to be processed on each queue.
- Each of the queues is started. (As long as enabled=true is set, processing will start as soon as the feed file is assigned to the queue.)
- As each queue is started, its status will be “ok”.
- As each queue completes, the status will change to ‘eos’.

- Any feed with status ‘eos’ is assigned the next file in line to be processed.
- Files continue to be added to queues until all files are processed.

41.3.1 Create Feed Queues

In order to create a feed queue, create an additional feed called `queue2`.

```
> cp template.json queue2.json
> update_virgo_feed.sh queue2.json Site06 Camera04
  'file:///files/videos/test/vid01.mp4'
> sudo docker exec -it safr_virgo ./virgo feed add queue2
  /files/feeds/queue2.json
```

You’ll maintain different queue files (queue1.json and queue2.json) in order to allow queue1 and queue2 to be processed independently. Each queue file serves as a record of what is running on the current queue. If an error occurs on one of the queues, you can use the *get_input_stream.sh* script to identify the file that was not processed for the queue and write it to an error log file.

The 2nd feed should start automatically. If you check the VIRGO Service Monitor, you should see something like the following:

```
Status      Feed           Epoch     P-Time        Resolution  FPS  DPS
   dDt     dRt     #D
eos         queue1         04:45:50  00:00:04.343  1920x1080   -    -    -
         -       816
ok          queue2         06:28:36  00:00:04.413  1920x1080   -    -    -
         -       819
```

41.3.2 Create Feed File List

The feed file list is a CSV file with following columns:

- **Video File Relative Path** - The path of the video file. It is relative to the base path set in the *process_files.sh* script.
- **Site** - Specifies site (building, GPS location, etc) name for the purpose of identifying location.
- **Source** - Specifies the camera name for purposes of identifying location.
- **Date** - Internal use only.
- **EpochDate** - Start time of the video file in Unix Epoch date format. This is used to ensure events are created at the proper point in the timeline.
  - This value must be in Epoch milliseconds.
  - See the Process Video Files Script section below for information on creating Epoch values using the *process_files.sh* script.
  - Epoch millisecond values can be created from human readable dates in Excel using this formula: =(D1-DATE(1970,1,1))*86400*1000
    - Where “D1” is the cell containing the start date of the video.

Below is an example of the contents of the feed file list file:

```
vid001.mp4,Site05,Camera025,8/20/19 7:12 AM,1566285120000
vid002.mp4,Site03,Camera024,8/20/19 7:12 AM,1566285120000
xvid003.mp4,Site01,Camera003,8/20/19 4:48 AM,1566276480000
vid004.mp4,Site01,Camera030,8/20/19 12:00 PM,1566302400000
vid005.mp4,Site04,Camera016,8/20/19 12:00 PM,1566302400000
vid006.mp4,Site04,Camera002,8/20/19 7:12 AM,1566285120000
vid007.mp4,Site03,Camera016,8/20/19 12:00 PM,1566302400000
xvid008.mp4,Site02,Camera020,8/20/19 12:00 PM,1566302400000
vid009.mp4,Site05,Camera011,8/20/19 9:36 AM,1566293760000
```

```
vid010.mp4,Site02,Camera030,8/20/19 7:12 AM,1566285120000
vid011.mp4,Site05,Camera021,8/20/19 9:36 AM,1566293760000
vid012.mp4,Site04,Camera009,8/20/19 9:36 AM,1566293760000
vid013.mp4,Site01,Camera001,8/20/19 12:00 PM,1566302400000
vid014.mp4,Site02,Camera017,8/20/19 9:36 AM,1566293760000
vid015.mp4,Site04,Camera014,8/20/19 12:00 PM,1566302400000
vid016.mp4,Site02,Camera010,8/20/19 7:12 AM,1566285120000
vid017.mp4,Site04,Camera030,8/20/19 7:12 AM,1566285120000
vid018.mp4,Site02,Camera022,8/20/19 9:36 AM,1566293760000
vid019.mp4,Site05,Camera006,8/20/19 4:48 AM,1566276480000
vid020.mp4,Site02,Camera012,8/20/19 9:36 AM,1566293760000
vid021.mp4,Site02,Camera005,8/20/19 7:12 AM,1566285120000
xvid022.mp4,Site01,Camera019,8/20/19 4:48 AM,1566276480000
xvid023.mp4,Site02,Camera016,8/20/19 7:12 AM,1566285120000
vid024.mp4,Site04,Camera027,8/20/19 7:12 AM,1566285120000
```

41.3.3 Edit the Process Files Script

Edit the configurable parameters in the *process_files.sh* script. Below are the default values in the script.

```
## USER CONFIGURABLE PARAMETERS ##
##################################
# User Directory
user_dir=main1
# Set this to the location of the feed configuraiton files.
feeds_dir=/files/feeds
# If different than above, Set to the path to feeds from inside docker
   container
docker_feeds_dir=$feeds_dir
# Set this to location of the video files
video_files_dir=/files/videos/1min_segments
###################################
```

- *user_dir* is the user directory where identities and events are stored. It can be useful to edit this parameter when experimenting with different settings.
- The default values for *feeds_dir* and *docker_feeds_dir* are the same because of the most common ways that host file systems are mounted in the docker container. This may not necessarily be the case so the two attribute are separated in the script
- The script assumes video_files path is identical in the host as well as the docker container. Script would need to be modified if this was not the case (one part of the script checks the existence of the file from the host OS while in another place, the path to the video file is passed to VIRGO running in the docker container via the feed configuration file

41.4 Reference

This section includes reference files and example scripts useful in processing video files with the Command Line Interface.

41.4.1 VIRGO Command Line Help

```
Command line interface to the virgo daemon
Syntax:

'virgo' followed by one of the following:

```

```
administrator set <administrator>         sets the administrator. Either
   'virgo' or 'virga'
administrator get                         shows the current administrator

environment get                           shows the current environment
environment set <environment>             sets the current environment
environment list                          shows all supported environments

feed list                                 shows all known feeds
feed get <feed> [path]                    shows the feed configuration
   and optionally saves it as a feed configuration file
feed status <feed>                        shows the current feed status
feed start <feed>                         marks the feed as enabled and
   starts it running
feed stop <feed>                          stops the feed and marks it as
   disabled
feed remove <feed>                        stops the feed and removes it
feed add <feed> <path to config>          reads the feed configuration
   file and adds the new feed to the known feeds
feed set <feed> <path to config>          reads the feed configuration
   file and updates the feed with the new configuration
feed capture-image <feed> <url or path>   captures one or more images
   from the feed and stores them in the directory <url>

service info                              shows the service information
service status                            shows the current status of all
   feeds
service reset                             resets the persistent virgo
   state back to the factory defaults
service log                               continuously shows the current
   service log information
service monitor                           continuously shows the current
   service status and statistical information
service update <version> [<url or path>]  upgrades or downgrades virgo to
   the specified version
service versions                          shows all installed versions

user get                                  shows the current user cloud
   identity
user set [user id]                        sets the current user cloud
   identity

<environment> is an environment name.
<feed> is a feed name.
<version> is a semantic version number (e.g. 1.0.3).

--help | -h     shows this help message.
--verbose | -v  enables the display of more detailed information.

--max-frames    the maximum number of frames to capture from a feed
--frame-delay   the delay between capturing frames. This is in
   milliseconds
--size          the size to which a captured image should be scaled
```

41.4.2 Process Video Files Script

process_files.sh

This is the primary commands that loops through all files and adds them to VIRGO queues for processing.

```
#!/bin/bash
script_dir=$(dirname "$0")

## USER CONFIGURABLE PARAMETERS ##
##################################
# Set this to the location of the feed configuraiton files.
feeds_dir=/files/feeds
# If different than above, Set to the path to feeds from inside docker
   container
docker_feeds_dir=$feeds_dir
# Set this to location of the video files
video_files_dir=/files/videos/1min_segments
# User Directory
user_dir=main1
###################################

# Make sure we got correct number of args.
if [ "$#" -lt 1 ]; then
    echo "Usage: $0 <job list file>
    Where <job list file> csv file with video_file,site,source";exit;
fi
# Check if video file list exists
[ ! -f $1 ] && { echo "$1 video list file not found"; exit 99; };

### Loop thru each line in input file
#########################################
OLDIFS=$IFS; IFS=,
cat $1 | tr -d '\r' | while read fname site source date epoch; do
  echo "Placing file $fname for site: $site, source: $source and time:
     $epoch"
  [ ! -f "$video_files_dir/$fname" ] && {
     echo "$video_files_dir/$fname not found. Skipping" | tee -a
        error.log; continue;
  }

  ### Loop Feeds - Outer loop
  ##  Get list of queues and pass this into inner loop
  ### Break once inner loop exits with successful assignment
  ############################################################################
  assigned=false; echo "processing file $fname"
  while true; do

    ### Loop Feeds - Inner loop
    ### Loop thru each queue - check to see if any are ready for next job
       ###
    ############################################################################
    while read feed_line; do
    TEMP=$IFS; IFS=': ' read queue_id queue_status <<< $feed_line;
       IFS=$TEMP
```

```
      if [ "$queue_status" == "failed" ]; then
        # Something went wrong.  Log file being processed by this queue
           to look at later
        echo $($script_dir/get_input_stream.sh $queue_id.json) >>
           error.log
      fi
      if [ "$queue_status" == "ok" ] || [ "$queue_status" == "prerolling"
         ]; then
        # Skip and go to next feed
        echo "Skipping queue $queue_id with status $queue_status"
      elif [ "$queue_status" == "eos" ] || [ "$queue_status" ==
         "inactive" ]; then
        $script_dir/update_virgo_feed.sh $feeds_dir/$queue_id.json $site
           $source $epoch "file://$video_files_dir/$fname"
        $script_dir/set_virgo_feed_attr.sh $feeds_dir/$queue_id.json
           directory  $user_dir
        echo "Set file $fname on queue $queue_id" | tee -a processed.log;
        docker exec safr_virgo ./virgo feed set $queue_id
           $feeds_dir/$queue_id.json
        if [ "$queue_status" == "inactive" ]; then
          docker exec safr_virgo ./virgo feed start $queue_id
        fi
        # Skip all other feeds and go to next file
        assigned=true; break 1;
      else
        #  Unexpected processing status. Report in error log
        echo "Unexpected virgo feed status $queue_status"  >> error.log
      fi
    sleep 1
    done <<< "$(docker exec safr_virgo ./virgo service status)"
   if $assigned; then echo "Assigned $fname to $queue_id"; break 1; fi
  done
done
IFS=$OLDIFS
```

41.4.3 Utility Scripts

The following scripts are used by the `processing_files.sh` script.

update_virgo_feed.sh

```
#!/bin/bash
if [ "$#" -lt 4 ]; then
    echo "Usage: $0 <feed filename> <sitename> <sourcename> <starttime>
       '<file_path>'";exit;
fi
sed -i -e "s|\"site\" *: *\"[^\"]*\"|\"site\" : \"$2\"|g" $1
sed -i -e "s|\"source\" *: *\"[^\"]*\"|\"source\" : \"$3\"|g" $1
sed -i -e "s|\"reporter.events-initial-date-offset\" *:
   *\"[^\"]*\"|\"reporter.events-initial-date-offset\" : \"$4\"|g" $1
sed -i -e "s|\"input.stream.url\" *: *\"[^\"]*\"|\"input.stream.url\" :
   \"$5\"|g" $1
```

Where:

- **<feed filename>** - Name of the file that contains feed settings.
- **<sitename>** - Specifies the site (building, GPS location, etc) name for the purpose of identifying the location.
- **<sourcename>** - Specifies the camera name for the purpose of identifying the location.
- **<starttime>** - Start time/date of the video. The value given as Epoch in milliseconds. For example, for Aug 1, 2019 at 11 AM, the epoch value is 1564682400000.
  - Try the Chrome extension ‘utime’ to create epoch times. This extension allows you to type any date/time and get epoch values as well as natural language strings such as ‘1 hour ago’. Be sure to set the time format to “milliseconds” or else the time will be off by many years.
- **<file path>** - The fully qualified path to the file relative to the Docker container.

set_virgo_feed_attr.sh

```
#!/bin/bash
if [ "$#" -lt 3 ]; then
    echo "Usage: $0 <feed filename> <attr name> <attr value>";exit;
fi
sed -i -e "s|\"$2\" *: *\"[^\"]*\"|\"$2\" : \"$3\"|g" $1
```

Where:

- **<feed filename>** - Name of the file that contains feed settings.
- **<attr name>** - The JSON attribute name from the feed configuration file.
- **<attr value>** - The value to be assigned to the JSON attribute from the feed configuration file.

get_input_stream.sh

```
#!/bin/bash
if [ "$#" -lt 1 ]; then
    echo "Usage: $0 <feed filename>"
fi
grep input.stream.url $1 | sed -e 's:"input.stream.url" *\:
   *"\([^"]*\)",*:\1:g'
```

Where:

- **<feed filename>** - Name of the file that contains feed settings.

42 VIRGO Tools

The Video Recognition Gateway (VIRGO) installation on Linux includes a couple scripts to manage VIRGO. They are located at`/opt/RealNetworks/SAFR/virgo/app`.

42.1 Back Up virgo-factory.conf

Before you use VIRGO tools you should back up your **virgo-factory.conf**. This configuration file can be found at `/opt/RealNetworks/SAFR/virgo/config`.

42.2 virgo_updateip.sh

This script uses the current local IP to update the VIRGO configuration file (**virgo-factory.conf**). To run it, do the following:

1. Go to `/opt/RealNetworks/SAFR/virgo/app`.
2. Run *virgo_updateip.sh*. The syntax to run the script is `./virgo_updateip.sh`. The following output will be shown.

```
11:22:38 CST - Collect local ip

11:22:38 CST - Local IP 10.10.51.189

11:22:38 CST - Updating configuration files

11:22:38 CST - Applying new virgo configuration
```

The IP has now been changed in **virgo-factory.conf**.

42.3 virgo_configure.sh

This script will reset the VIRGO service. To run it, do the following:

1. Go to `/opt/RealNetworks/SAFR/virgo/app`.
2. Run *virgo_configure.sh*. The syntax to run the script is `./virgo_configure.sh <Username> <Password>`.
   - For *<Username>*, use the *user-id* value found in the **virgo-factory.conf** file.
   - For *<Password>*, use the *user-encrypted-password* value found in the **virgo-factory.conf** file. The following output will be shown.

```
11:54:21 CST - Updating virgo factory configuration
11:54:21 CST - Username realnetworksbei13
11:54:21 CST - Password
   %pHUQfSS4mk7UIrhs5au0aZ\+qshbJPGIx4rEw\/RpCgGpUrYCiWrzc2uh8g9HsxJHf
11:54:21 CST - Cleaning out old virgo configuration
11:54:21 CST - Cloning template file
11:54:21 CST - Collect local ip
11:54:21 CST - Local IP 10-10-51-189
11:54:21 CST - Copy config to /etc/ folder
11:54:21 CST - Reset Virgo to reload configuration
11:54:21 CST - Finished
```

43 Factory Configuration File

Every Video Recognition Gateway (VIRGO) daemon on Linux ships with factory settings which define the default configuration that the daemon should use the first time it starts up. *Virgod* also reverts the current configuration back to the factory settings if it is unable to load the current configuration because of a version mismatch and it is unable to automatically convert the old configuration to the new format.

The factory settings are stored in a JSON file with the name `virgo-factory.conf`. *Virgod* looks in the following locations to find a factory configuration file:

- The home directory of the user who started *virgod*.
- The `/etc` directory.
- The VIRGO bundle directory.

Virgod loads the first factory configuration file that it finds. If it can't find any factory configuration file, it falls back to hardcoded defaults.

43.1 Factory Configuration File Format

The factory configuration file is a JSON file which is organized into (optional) sections:

```
{
   "global": {             // [optional]
      // global state
   },
   "environments": {       // [optional]
      "Foo": {
         // environment-specific URLs
      }
   },
   "feeds": {              // [optional]
      "camera_1": {
         // feed state
      }
   }
}
```

Note: Nearly all keys in a factory configuration file are optional. Only those keys that you explicitly want to override with a custom value need to be specified. *Virgod* uses hardcoded default values for keys that are missing from a factory configuration file.

43.1.1 The Global Section

The following properties are supported in the global section:

Property	Type	Default	Description
status-interval	Int?	5000	Status reporting time interval in ms.
environment	String?	PROD	The name of the environment which should be used by *virgod*. See the “Environments Section” below for a list of pre-defined environment names.

Property	Type	Default	Description
machine-id-prefix	String?	empty string	The machine ID prefix. The default machine ID prefix is the empty string.
machine-id	String?	OS defined machine ID	The machine ID. The default machine ID is derived from the OS provided machine ID. The concatenation of the machine-id-prefix and the machine-id is sent to the cloud in the X-CLIENT-ID header.
client-type	String?	OS defined client type	The client type. This value is sent to the cloud in the X-CLIENT-TYPE header.
user-id	String		The user ID for the cloud account.
user-password	String		The password for the cloud account. Note that the password is stored in clear text. Use *user-encrypted-password* whenever possible instead.
user-encrypted-password	String		The encrypted password for the cloud account.
administrator	String?	cloud	Specifies whether VIRGO should be administrated by VIRGA or whether it should be self-administrated. A self-administrated VIRGO allows you to manage feeds via the VIRGO command line tool.

Property	Type	Default	Description
visible-accelerator-ids	[Int]?		Allows you to specify which GPUs/accelerators VIRGO is allowed to use for video decoding and detection tasks. Only the accelerators listed in this array will be used by VIRGO; all others will be ignored. The value is an array of accelerator IDs. VIRGO will use all available accelerators if this property is not set.

Note that feeds which are assigned to a specific accelerator ID will fail with an error at startup if that accelerator is not in the set of visible accelerator IDs.

43.1.2 The Environments Section

The environments section defines the available cloud environments. Each environment has a name and a set of URLs that point to the hosts in the cloud that provide the required services. An environment may override one of the pre-defined environments. The environment name is used to identify the environment and to switch among environments with the `virgo environment set` command.

The following properties are supported in the environments section:

Property	Type	Default	Description
covi-server-url	URL	none	The face recognition service.
rncv-server-url	URL?	none	The face detection service.
event-server-url	URL	none	The detection and recognition event recording service.
object-server-url	URL	none	The service which stores objects such as images and logs.
admin-server-url	URL	none	The VIRGO administration service.

The following table lists the pre-defined environments:

Name	Alternative name
SAFR Local	LOCAL
SAFR Developer Cloud	DEV
SAFR Partner Cloud	INT2
SAFR Cloud	PROD

You can use the alternative environment name in place of the full environment name.

43.1.3 The Client ID

VIRGO computes the client ID by concatenating the machine-id-prefix and the machine-id properties.

43.2 Example Configuration Files

The following subsections show some typical factory configuration files.

43.2.1 Using VIRGO with a VIRGA Server

This is an example of a configuration file which configures VIRGO to run as a slave to a Video Recognition Gateway Admin (VIRGA) server. VIRGO will continuously report its status to the VIRGA server and the VIRGA server is responsible for pushing state changes to VIRGO.

```
{
   "global" : {
      "environment": "PROD",
      "machine-id-prefix": "foo",
      "user-id": <user ID>,
      "user-password": <password>
   }
}
```

43.2.2 Using VIRGO Standalone

This is an example of a configuration file which configures VIRGO to run as a standalone daemon which does not connect to a VIRGA server. VIRGO starts processing the declared feeds as soon as it starts up. Note that you still have to provide a user ID and a password to allow VIRGO to use the (cloud-based) face recognition and event recording service.

```
{
   "global":{
      "environment": "PROD",
      "machine-id": "argusrn",
      "user-id": <user ID>,
      "user-password": <password>,
      "administrator":"self"
   },
   "feeds":{
      "camera_1":{
         "directory":"test",
         "input.type":"stream",
         "input.stream.url":"file://<absolute path to a movie file>",
         "recognizer.learning-enabled":true,
         "enabled":true
      }
   }
}
```

43.2.3 Defining Custom Environments

This is an example of a configuration file which defines two custom cloud environments. Note that the first custom environment has a new unique name that is separate from any of the pre-defined environments.

The second custom environment, on the other hand, overrides the pre-defined environment name `PROD`. Consequently VIRGO will use the URLs of the custom environment if the `PROD` environment is selected. This allows you to replace the built-in definition of the pre-defined environment.

```
{
   "global": {
      "environment": "Test"
   },
   "environments": {
      "Test": {
         "covi-server-url": "https://covi.test.real.com",
         "event-server-url": "https://event.test.real.com",
         "object-server-url": "https://object.test.real.com",
         "admin-server-url": "https://admin.test.real.com"
      },
      "PROD": {
         "covi-server-url": "https://covi.sim.real.com",
         "event-server-url": "https://event.sim.real.com",
         "object-server-url": "https://object.sim.real.com",
         "admin-server-url": "https://admin.sim.real.com"
      }
   }
}
```

44 Docker

The Video Recognition Gateway (VIRGO) application runs as a Docker Container alongside all the other native services as part of the SAFR Linux Platform.

44.1 Initial Configuration

The VIRGO container starts for the first time with no factory configuration file. It will remain in this state until a new configuration has been generated and activated.

44.2 Configuration

The factory configuration file is generated when the following configuration script is called by CoVi during licensing (kickoff):

```
/opt/RealNetworks/SAFR/virgo/app/virgo/app/virgo_configure.sh
```

The script requires both a *username* and a hashed *password* to be passed in.

NOTE: If either of these are missing the script will not generate the configuration.

The script requires a template file `/opt/RealNetworks/SAFR/virgo/app/virgo/config/virgo-factory.template` in order to generate a new configuration.

Once executed the script will generate a working configuration and will store it in the following file.

NOTE: The existing configuration will be overwritten!

```
/opt/RealNetworks/SAFR/virgo/app/virgo/config/virgo-factory.conf
```

After the configuration is generated the VIRGO container will be restarted to activate the newly generated configuration.

44.3 Service Status

There are two ways to confirm that VIRGO is running and to confirm how long it has been operational.

1. Use the **check** utility located in `/opt/RealNetworks/SAFR/bin`
2. Use the Docker command to show active running containers:
   - **# sudo docker ps**

```
CONTAINER ID  IMAGE                COMMAND                   CREATED
   STATUS        PORTS   NAMES
cf2a2dd33875  safr_virgo:1.1.38   "/bin/sh -c $VIRGO_AâĂę"   18 hours ago
     Up 18 hours              safr_virgo
```

If there is no output check the following:

- Run the same command again with the -a switch to determine if the container is stopped or restarting (i.e. failing).
- Verify there is a valid virgo-factory.conf file located in `/opt/RealNetworks/SAFR/virgo/configs/`
  - Correct user name and IP address. (Same as host IP.)
  - Password is not readable so hard to validate

44.4 Execution

VIRGO container will remain operational both after a failure has occurred as well as if the OS is restarted.

The container is automatically started by the SAFR Platform Installer and stopped by the SAFR Platform Uninstaller.

44.5 Logging

Execute the following command to provide logging output.

```
sudo docker exec -it safr_virgo /opt/RealNetworks/virgo/virgo service log
   <log options>
```

NOTE: Refer to VIRGO Logging for more information on logging options.

44.6 Service Monitor

Live view

```
sudo docker exec -it safr_virgo /opt/RealNetworks/virgo/virgo service
   monitor
```

Active Feeds to CSV

```
sudo docker exec -i safr_virgo /opt/RealNetworks/virgo/virgo service
   monitor > {CSV File} --active-only
```

NOTE: The stats are added to the CSV file every second so the usable data can be large depending on the number of active feeds.

44.7 Upgrade

To upgrade VIRGO you need to perform the following steps depending the platform architecture.

44.7.1 Standalone Container

- Upload new VIRGO Docker Image to the deployment server (location does not matter).
- Load image into local registry.

  ```
  docker load < {image_file}
  ```

- Update `/opt/RealNetworks/SAFR/virgo/app/docker-compose.yml`.
- Restart VIRGO container.

  ```
  docker restart safr_virgo
  ```

44.8 Add Volume Mount to Existing Container

1. Update the compose file to add the additional volume instructions.
   - The format is <local folder>:<docker folder>
   - The <docker folder> will be created if not already existing.

     Example: (Your folder names might be different.)

     ```
     version: "3.6"
     services:
     virgo:
     image: safr_virgo:1.2.12
     container_name: safr_virgo
     restart: on-failure
     pid: "host"
     volumes:
     - /opt/RealNetworks/SAFR/virgo/config/:/etc/virgo
     - /opt/RealNetworks/SAFR/virgo/files:/opt/RealNetworks/virgo/files
     ```

2. Create the local folder to mount into the container.

   ```
   # mkdir -p /opt/RealNetworks/SAFR/virgo/files
   ```

3. Stop and delete the container.

   ```
   # docker-compose -f /opt/RealNetworks/SAFR/virgo/app/docker-compose.yml
      down
   ```

4. Create a container instance with new volume mount.

   ```
   # docker-compose -f /opt/RealNetworks/SAFR/virgo/app/docker-compose.yml
      up -d
   ```

5. Createa test file in local mount point.

   ```
   # touch /opt/RealNetworks/SAFR/virgo/files/testfile
   ```

6. Check that the test file exists inside the container's mount location.

   ```
   # docker exec -it safr_virgo ls -l /opt/RealNetworks/virgo/files
   ```

45 GPU Support

Starting with version 1.1.16, Video Recognition Gateway (VIRGO) supports acceleration of video decoding, graphics processing, and face detection functions via one or more GPUs. VIRGO automatically detects the presence of a compatible graphics card and will use it. On systems without a GPU, VIRGO falls back to doing everything on the CPU.

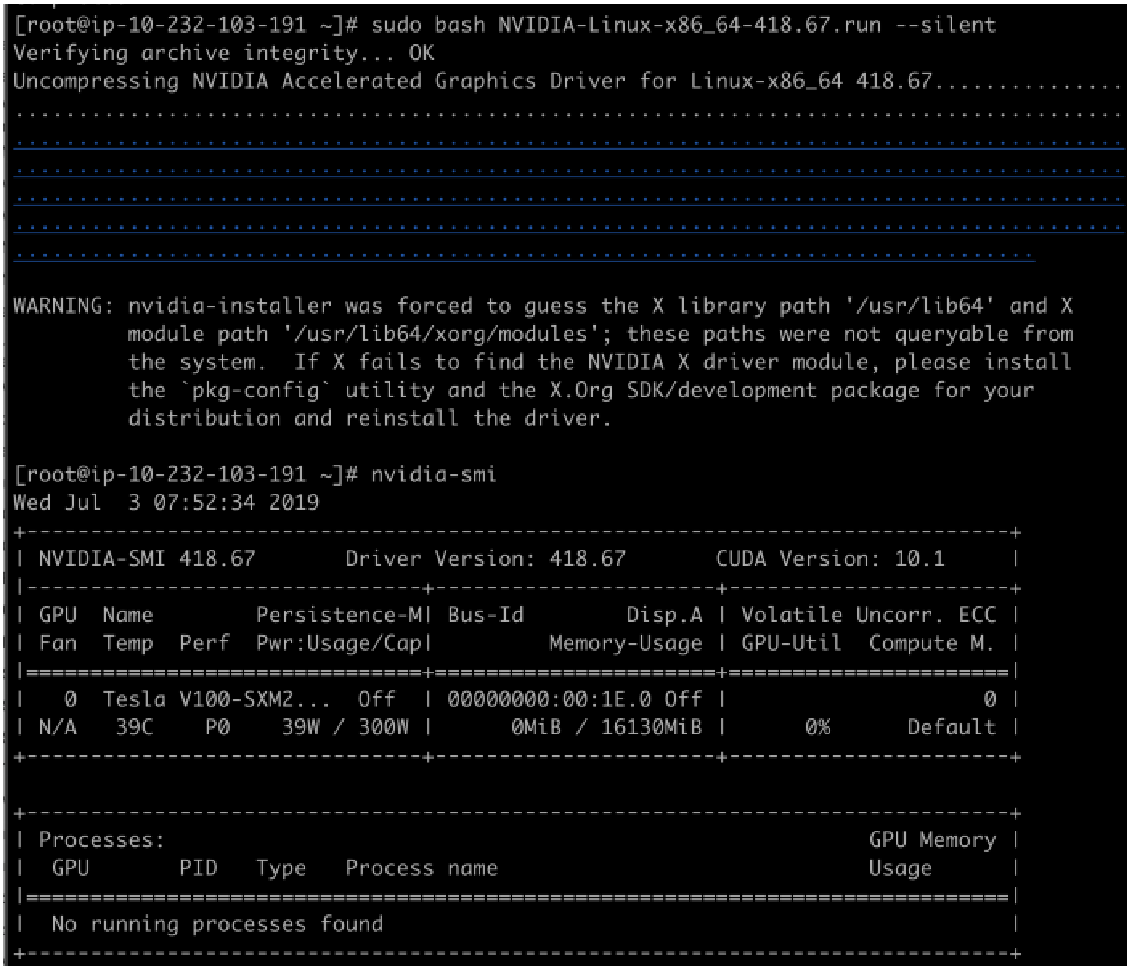
Only Nvidia Compute Unified Device Architecture (CUDA) GPUs are currently supported.

45.1 Linux GPU Requirements and Prerequisites

NVIDIA drivers version 418.67 or greater are required. The CUDA toolkit is not required. In addition, you need to install some prerequisites as described below.

45.1.1 Install Prerequisities

1. Install dependencies.
   - For Ubuntu: Run `DEBIAN_FRONTEND=noninteractive apt-get update -y && apt-get install -y gcc make`
   - For Centos: Run `yum install -y gcc make kernel-devel`
   - For Amazon: Run `yum install -y gcc make "kernel-devel-uname-r == $(uname -r)"`
2. Download the most recent NVIDIA Linux drivers from https://www.nvidia.com/object/unix.html.
   - Example: `curl -LO http://us.download.nvidia.com/tesla/418.67/NVIDIA-Linux-x86_64-418.67.run`
3. Stop x-windows, if running:
   - For Ubuntu: Run `service lightdm stop`
4. Run driver installer:
   - Run `sudo bash NVIDIA-Linux-x86_64-418.67.run --silent`
5. Verify that your installation was successful:
   - Run `nvidia-smi`



```
[root@ip-10-232-103-191 ~]# sudo bash NVIDIA-Linux-x86_64-418.67.run --silent
Verifying archive integrity... OK
Uncompressing NVIDIA Accelerated Graphics Driver for Linux-x86_64 418.67..............
..................................................................................
..................................................................................
..................................................................................
..................................................................................
..................................................................................
...........................................................................

WARNING: nvidia-installer was forced to guess the X library path '/usr/lib64' and X
         module path '/usr/lib64/xorg/modules'; these paths were not queryable from
         the system.  If X fails to find the NVIDIA X driver module, please install
         the `pkg-config` utility and the X.Org SDK/development package for your
         distribution and reinstall the driver.

[root@ip-10-232-103-191 ~]# nvidia-smi
Wed Jul  3 07:52:34 2019
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 418.67       Driver Version: 418.67       CUDA Version: 10.1     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|===============================+======================+======================|
|   0  Tesla V100-SXM2...  Off  | 00000000:00:1E.0 Off |                    0 |
| N/A   39C    P0    39W / 300W |      0MiB / 16130MiB |      0%      Default |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                       GPU Memory |
|  GPU       PID   Type   Process name                             Usage      |
|=============================================================================|
|  No running processes found                                                 |
+-----------------------------------------------------------------------------+
```

6. If your installation was unsuccessful, view the log:
    - Run `less /var/log/nvidia-installer.log`

45.2 Windows GPU Requirements

NVIDIA drivers version 418.67 or greater are required. The CUDA toolkit is not required.

45.3 Enable a Feed to Run on a GPU

There's nothing you need to do to make this happen; VIRGO automatically detects the presence of a suitable GPU and assigns a feed to it. A feed will automatically fall back to the CPU if there's a problem with the GPU or all GPU resources have been exhausted.

VIRGO also takes advantage of multiple GPUs installed in the system. It automatically distributes feeds across all available GPUs. This enables you to easily scale up a system which allows you to run more feeds on a single VIRGO host.

VIRGO returns comprehensive statistical information about a feed. This includes information about which GPU a feed is running on as well as how much of its processing power it's using per second.

45.3.1 Manual Feed Assignment

Sometimes more control over which feed is assigned to the CPU vs a GPU is desired. VIRGO allows you to individually specify for each feed whether it should exclusively run on a GPU or the CPU. This allows you to maximize the use of all available GPUs and the CPU by assigning some feeds exclusively to the GPU and some exclusively to the CPU. The following table shows the available feed accelerator configurations:

VIRGO Feed Property	Property Value	Description
accelerator	auto	VIRGO will automatically pick the best available acceleration type. For example, VIRGO will assign the feed to one of the available GPUs if there is still processing capacity available. Otherwise VIRGO will assign the feed to the CPU.
	cpu	The feed will exclusively run on the CPU and not use any GPU even if a GPU is available.
	gpu	The feed will exclusively run on a GPU and not use the CPU for video decoding, graphics processing, or detection. The feed will fail if no GPU is available.

46 Service Logging

The Video Recognition Gateway (VIRGO) command line tool has a simple logger built in. You enable logging by executing the following command in a shell:

```
> virgo service log <log specification>
```

where the log specification is a space-separated list of log predicates. A log predicate looks like this:

```
level/tag
level/tag[feedName]
```

The first variant sets the log level for the package *tag* to *level* on a global basis. Consequently this log predicate applies to the VIRGO daemon and all feeds it spawns. The second variant allows you to apply the log predicate to a single feed with the name *feedName*. If you specify both a global- and a feed-specific log level for a tag then the level with higher priority is applied.

Note: The VIRGO daemon does not keep a log history. Log information is only generated and retained while you are actively running a `virgo service log` command.

Examples:

Enable DEBUG-level logging for the 'tracking' package in all feeds:

```
> virgo service log D/tracking
```

Enable DEBUG-level logging for the 'capture' and the 'cop-http' packages in all feeds:

```
> virgo service log D/capture D/cop-http
```

Enable DEBUG-level logging for the 'tracking' package in the feed 'foo': (This does not change the current log configuration for any other feed.)

```
> virgo service log D/tracking[foo]
```

The following log levels are supported:

Level	Description
V	Verbose
D	Debug
I	Info
W	Warn
E	Error
O	Off

The order in terms of verbosity, from least to most verbose is OFF, ERROR, WARN, INFO, DEBUG, and VERBOSE.

The following log packages are supported:

Package	Supports feed name?	Description
detection	yes	Object detector related messages
recognition	yes	Face recognizer related messages
tracking	yes	Object tracker messages
capture	yes	Image capture related messages
events	yes	Event reporting related messages
pose-liveness	yes	Pose Liveness Action Recognizer related messages

Package	Supports feed name?	Description
feed	yes	Feed life cycle related messages
cop-http	no	COP over HTTP related messages
config	no	*Virgod* configuration management related messages
updates	no	*Virgod* update initiation mechanism related messages

47 Service Monitoring

The Video Recognition Gateway (VIRGO) command line tool has a service monitoring user interface built in. Execute the following command in a shell window to activate continuous monitoring:

```
> virgo service monitor
```

After executing this command, VIRGO clears the terminal window and presents the following live screen:

```
Status    Feed      PID    Epoch     P-Time        Resolution FPS    DPS
     dDt    dRt    #D     #D-Badge  #D-Face  #D-Skip  #R    #R-Face
   #R-Err  #R-Skip  #Evt   %CPU  GPU#  GPU   GPU-Name
ok        camera_1  14536  12/06/17  00:24:13.450  1280x720   120    8ms
     250ms  120    18     10        0        0        8     0        0
         0        1240   1%    0     VF    GTX 1060
ok        camera_2  67289  13:07:12  80:10:00.000  1920x1080  29.97  8ms
     250ms  1920   1400   0         0        0        1000  50       1
         0        10     4%    1     VF    GTX 1050
inactive  camera_3
```

Note that the screen is live, which means that VIRGO continuously updates it every second. You can quit monitoring by pressing the 'q' key or by pressing Ctrl-C. Also please keep in mind that VIRGO only shows as many columns as fit on the screen. If you do not see all columns then this means that your terminal window is not wide enough. Make the window wider to see all of the columns.

The service monitor UI allows you to scroll up and down when there are more feeds than fit vertically in the terminal window. Use the cursor up key to scroll up and the cursor down key to scroll down.

The following table explains what the various columns in the monitoring output mean:

Column Name	Description
Status	The feed status. This is one of ok, inactive, eos, error, or failure.
Feed	The feed name.
PID	The PID of the feed daemon if the daemon is running
Epoch	The time when the feed processed the first frame in the video stream.
P-Time	The amount of time that the feed has spent on processing the video stream. This is in terms of milliseconds.
Resolution	The width and height of a video frame in pixels
FPS	The frames per second of the input video.
DPS	The number of detections per second.
dDt	The latency of a single detection operation in milliseconds.
dRt	The latency of a single recognition operation in milliseconds.
#D	The number of detection operations that have been triggered.
#D-Badge	The number of badges that have been detected.
#D-Face	The number of faces that have been detected.
#D-Skip	The number of detection operations that have been skipped due to detector overcommitment. This means that no detector was available for a video frame because all detectors were busy at that time.

Column Name	Description
#R	The number of face recognition or reconfirmation operations that have been triggered.
#R-Face	The number of successful face recognition or reconfirmation operations that have been run.
#R-Err	The number of face recognition or reconfirmation operations that have failed for some reason.
#R-Skip	The number of recognition operations that have been skipped due to recognizer overcommitment. This means that no recognizer was available for a face image because all recognizers were busy at that time.
#Evt	The number of events that have been reported.
%CPU	How CPU is used by the feed. Note that this number is in the range 0% to CPU_COUNT * 100%.
GPU#	The GPU ID. Every GPU in the system is assigned a unique ID. This entry is blank if the feed does not use a GPU.
GPU	A string which indicates which modules in the feed are using the GPU: V -> video decoder F -> face detector B -> badge detector O -> object detector An empty/non-existing string indicates that the feed is not using the GPU at all.
GPU-Name	The name of the GPU. Note that the name is not unique because a system may be equipped with more than one GPU of the same model and make. This entry is blank if the feed does not use a GPU.

47.1 Creating CSV Files

You can create a CSV file with all the information from the live service monitor screen by invoking the service monitor like this:

```
> virgo service monitor > my.csv
```

This command tells VIRGO that it should write the service monitor information into a CSV file instead of showing it on the screen. VIRGO will continue to write feed statistics once per second to the CSV file until you stop it by pressing Control-C in your terminal window.

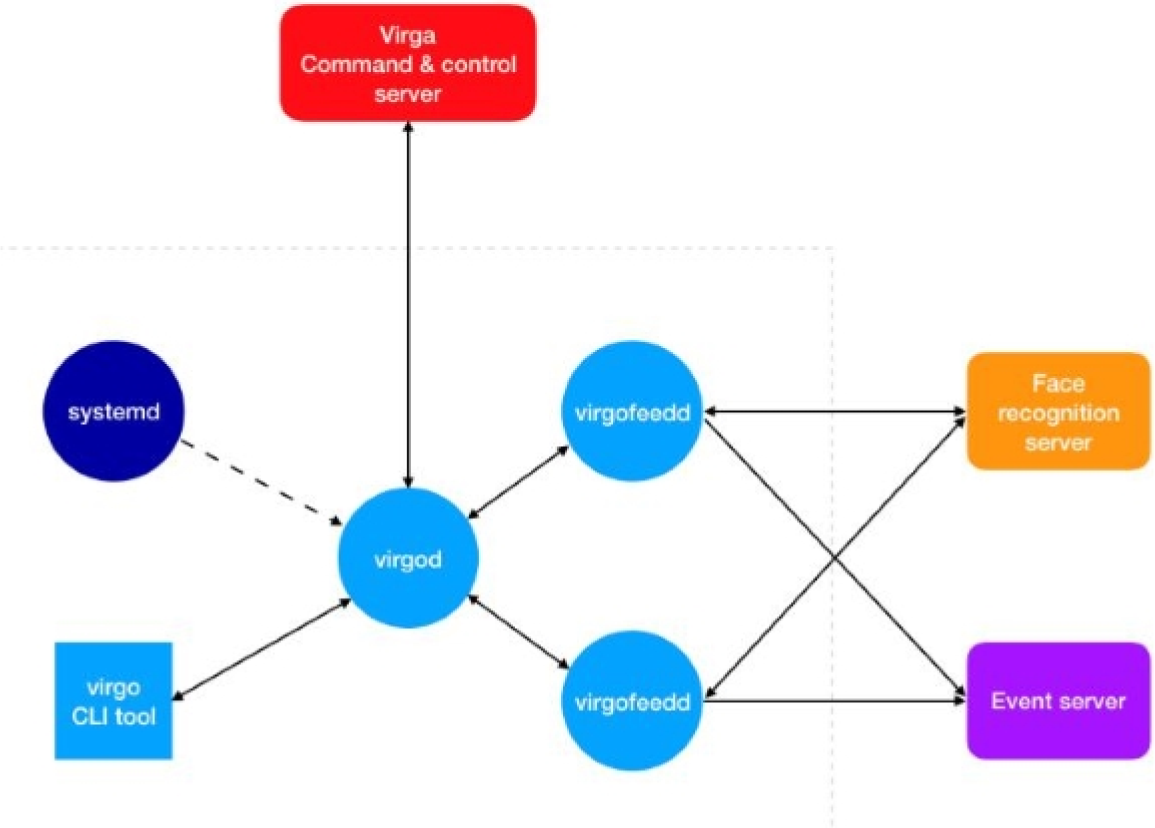
VIRGO writes one line per feed to the CSV file and it repeats this process every second. It even includes inactive feeds by default. If you only want to include active feeds in the CSV file then pass the "–active-only" command line switch to VIRGO.

48 VIRGO Architecture

A single Video Recognition Gateway (VIRGO) installation consists of the following components:

- **virgod**: The VIRGO control daemon. One such daemon is spawned and maintained per VIRGO hardware.
- **virgofeedd**: A *virgod* child process which handles a single video feed.
- **virgo**: The locally available VIRGO command line tool which acts as a Command Line Interface (CLI)-based user interface to the VIRGO daemon.

This diagram shows how those components fit together:



virgod:

- Spawned by the operating system systemd/launchd service. The daemon is automatically restarted by the OS if the hardware power cycles or virgod terminates for some unexpected reason.
- Runs as its own user. The VIRGO user is limited to read/write access to the "virgo" home directory.
- The VIRGO user home directory contains just the ~/Library directory which is the place where libFoundation (used in the implementation of VIRGO) stores the daemon settings.
- Is responsible for spawning the per-video-feed child processes: virgofeedd.
- *virgod* monitors each *virgofeedd* child process that it has spawned and it automatically restarts a *virgofeedd* if it unexpectedly terminates for some reason. (e.g. it ran out of memory)
- Is responsible for caring out all the necessary steps for an update to the VIRGO daemon system.
- Is the only process on the machine which talks to the VIRGA command & control server.
- Carries out any command sent by VIRGO to *virgod*.
- Regularly informs the Video Recognition Gateway Administrator (VIRGA) command and control server about the current status of *virgod*.

virgofeedd:

- Spawned by *virgod*.
- Runs as the same user as *virgod*.
- Receives a video stream. Detects and recognizes faces in that video stream, generates events, and reports them to the event server.
- Receives commands from *virgod*.

virgoupdaterd:

- Spawned by *virgod* after it has received an update request.
- Runs as the same user as *virgod*.
- Downloads the update archive, extracts it, installs the update bundle, and saves the current persistent *virgod* state.
- Restarts *virgod*. (*virgod* takes care of data migration.)
- Monitors *virgod* after restart and rolls back to the previous *virgod* version if the new *virgod* fails to startup or fails to check back in with a commit message in less than a couple seconds.
- After the update has finished, the updater exits.

virgo:

- Implements the local (CLI-based) user interface to *virgod*.
- Offers commands to show the current status, selects the cloud environment, gets a screen capture from a feed, etc.

48.1 VIRGO Bundle (File System Layout)

VIRGO ships as a bundle which supports multiple versions of the VIRGO daemon. The VIRGO bundle directory contains a "versions" directory which in turn contains one sub-directory per installed VIRGO version. The name of a version sub-directory is the semantic version number of the VIRGO installation. The "versions" directory also contains a symlink named "current". This symlink points to the version sub-directory which is currently active.

The version sub-directory stores all necessary executable, library, and data files for the VIRGO version.

VIRGO bundle layout:

```
virgo/
   versions/
      1.0.0/
         virgo
         virgod
         virgofeedd
         virgoupdaterd
         lib/
            <shared libraries>
         model/
            <tensor flow model files>
         virgo-factory.config
      current -> ./1.0.0
   virgo -> ./versions/current/virgo
```

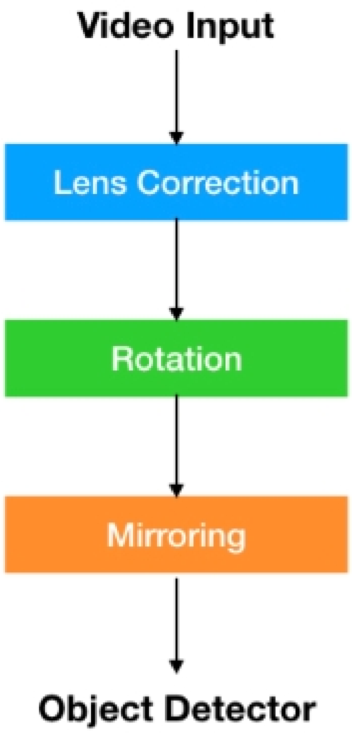
48.2 VIRGO Feeds

A single *virgod* instance manages a set of feeds. Each feed represents a video stream from a camera, a file, or some other video source. Each feed is associated with a set of configuration information which is stored persistently by VIRGO. The configuration information for the feeds can be managed through the VIRGO command line tool or through the video feeds window of the Desktop Client or the Web Console.

Each feed has a name which is unique among the set of feeds of a single *virgod* instance. These names are used as a simple and convenient way to refer to a feed and its configuration. Each feed is managed by a separate *virgofeedd* instance which is started and monitored by *virgod*. *Virgod* will automatically restart a *virgofeedd* instance if it dies for some unexpected reason.

A feed may be enabled or disabled. Only enabled feeds are associated with a *virgofeedd* instance. The enabled state of a feed may be changed through the VIRGO command line tool by issuing a **feed start** or a **feed stop** command. A feed may also be enabled or disabled in the video feeds windows by changing the **enabled** setting. This allows the system to reclaim resources like memory and network bandwidth if a feed is temporarily not needed. Feeds which are no longer needed at all should be removed altogether.

A feed has an input which connects the feed to a video stream. The two types of input currently supported are "stream" and "file". A stream input is specified by a URL which may point to a publicly accessible RTSP or HTTP video stream. Each video frame from the input is first sent through a video post-processing pipeline before it is fed into the object detector and recognizer sub-systems:



First a lens correction algorithm is applied to an incoming video frame. This step removes distortions that may be introduced by the optical system of a camera. After that the image will be rotated to compensate for any undesired rotation that may have been introduced by the physical orientation of the camera. Finally the image may be mirrored to ensure that a camera that is facing a user will produce an image that aligns with what a user expects to see.

49 Troubleshooting

49.1 Linux

49.1.1 Which Linux distributions are supported?

- Ubuntu 16.04(.5+) is known to work and has seen extensive testing.
- Ubuntu 18.04(.2+) appears to work but has not seen extensive testing.
- All other Linux distribution may or may not work; they have not seen any testing.

49.1.2 I just want to do a quick experiment with Video Recognition Gateway (VIRGO). Do I really have to do a full installation?

Actually no. If you just want to run VIRGO temporarily (e.g. to do testing) then there is no need to do a full installation. Do this instead:

1. Create a **virgo-factory.conf** file in your home directory which contains the necessary account, environment, and feed information.
2. Open a shell window and run `virgo/versions/current/virgod -l` in it.
3. Open a second shell window and use it to control VIRGO from there. For example, type `virgo/virgo service monitor` to see the current status of VIRGO.

Once you're done with your work you should terminate VIRGO by typing Control-C in the shell window in which you started *virgod*.

Here is a small example virgo-factory.conf file:

```
{
    "global":{
        "environment": "PROD",
        "machine-id-prefix": "vRGo-Rea18L-X-",
        "user-id": "<Your SAFR cloud account ID here>",
        "user-password": "<Your SAFR cloud account password here>",
        "remote-control-enabled":false
    },


    "feeds":{
        "Axis Q6128-E": {
            "directory":"testy",
            "input.type": "stream",
            "input.stream.url":"rtsp://user:password@101.102.103.104/axis-media/media.
            "enabled":true
        },
    }
}
```

Note that this quick & dirty way of running VIRGO is not suitable for a production system.

For example, VIRGO will stop running as soon as you log out of the system and the VIRGO factory configuration file is not secured which means that passwords (SAFR cloud account, camera IP passwords, etc) may be exposed to 3rd parties.

49.1.3 I've installed VIRGO but all my feeds die with an "Unexpected termination" error. What is wrong?

Your Linux installation is most likely missing a required APT package/library. Please make sure that you follow the installation instructions for Linux precisely. See this page for the list of required APT packages.

To find out which library is exactly missing, invoke the VIRGO feed daemon directly like this:

```
> virgo/versions/current/virgofeedd
```

This will cause the operating system to print the name of the missing library (.so file). Note that this command will print an error message about a missing/broken pipe if no library is missing. This later error is expected but any complaint about a missing dependency/library is not expected and points to a problem you need to fix.

If you see the following, it means that all dependencies are satisfied:

```
> virgo/versions/current/virgofeedd

Fatal error: 'try!' expression unexpectedly raised an error:
   virgofeedd.DTPError.io(message: "Bad file descriptor (9)"): file
   /var/lib/jenkins/workspace/ubuntu_16_04_virgo_trunk_daily/build/virgo-build-x86_64-
   line 31
```

If, on the other hand, you see the following, it means that a library is missing:

```
> virgo/versions/current/virgofeedd

virgo_installer/virgo/versions/current/virgofeedd: error while loading
   shared libraries: libcuda.so.1: cannot open shared object file: No
   such file or directory
```

49.1.4 I've connected a camera to VIRGO and it is perpetually stuck in prerolling mode with the error `Codec parameters not found`. What's going on?

Some cameras have buggy firmware which fail to generate a correct H264 PPS packet if the RTSP transport protocol is set to UDP. Note that VIRGO connects to RTSP cameras via UDP by default because UDP requires less networking resources and has lower latency compared to TCP.

However in this case and to fix this problem you need to tell VIRGO to connect to the camera using TCP instead. Do this by adding the following property to the feed dictionary for the camera:

```
"input.stream.rtsp.transport":"tcp"
```

49.1.5 I've just installed VIRGO, changed some things in the virgo-factory.conf file, and now *virgod* seems to crash all the time?!

Most likely there's a syntax error in the *virgo-factory.conf* file now. For example, you may have forgotten to add a comma at the end of a property. You can run *virgod* like this to see the actual error message:

```
> virgo/versions/current/virgod -l

Factory config error:
   dataCorrupted(Swift.DecodingError.Context(codingPath: [],
   debugDescription: "The given data was not valid JSON.",
   underlyingError: Optional(Error Domain=NSCocoaErrorDomain Code=3840
   "Badly formed object around character 54."
   UserInfo={NSDebugDescription=Badly formed object around character
   54.})))
```

You can also check the *virgod* exit code. It will be 78 (POSIX EX_CONFIG) if there is a syntax error in the factory configuration file.

Note that this kind of error can not be captured by the VIRGO logging system because it happens at the very startup of *virgod* and before the logging system has been initialized.

49.2 macOS

49.2.1 VIRGO crashes when I try to use it

You are most likely trying to run VIRGO on a system which does not have the Swift 5 runtime libraries installed. VIRGO depends on those libraries and Apple started shipping them with macOS beginning with version 14.4.4. If you are running an older OS and are not able to upgrade to a recent version of macOS then you should download the Swift 5 runtime libraries from Apple. See this support article for instructions on how to do this.

49.3 Docker

49.3.1 Feed reports "No Recogniser Available" after feed is added.

This type of error is normally produced when the Face Service is too busy to accept additional requests for recognition.

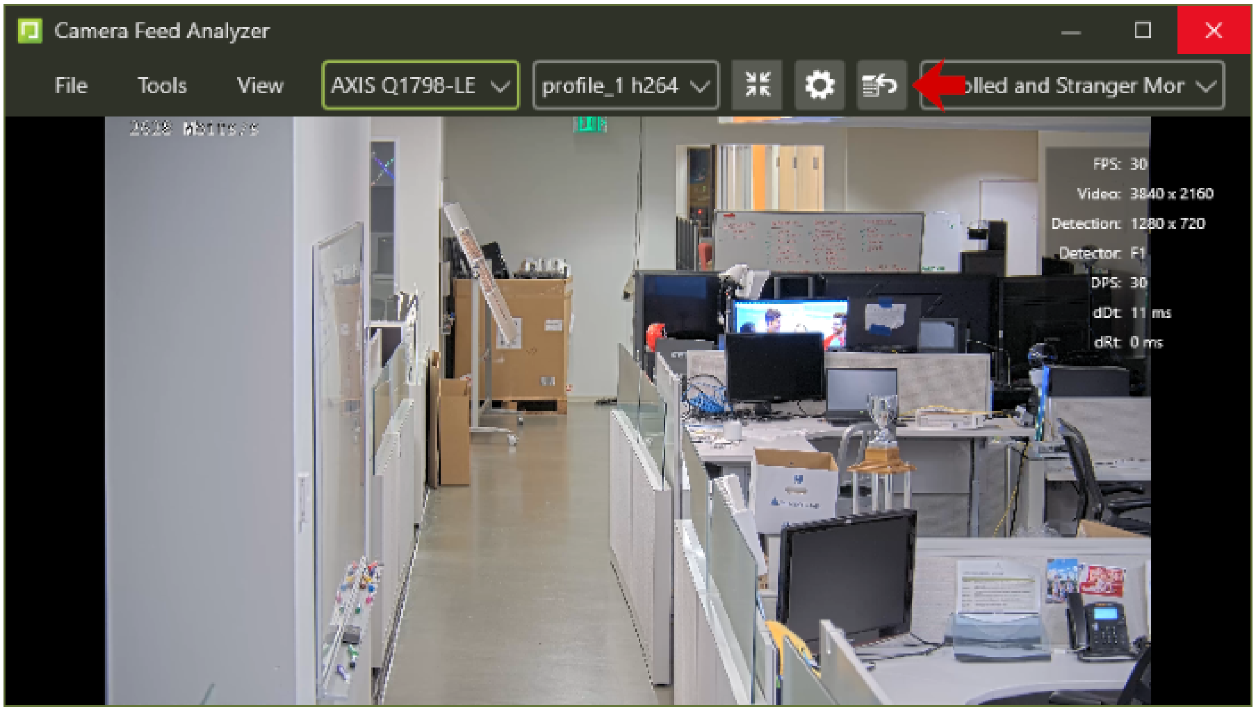
It can also be generated when the VIRGO configuration is incorrect and as such the requests are not getting sent to CoVi and thus time out.

50 Desktop Client

The Desktop Client is used to add and configure cameras, monitor feeds, get alerts, and view activity. It is also used to update and manage the Identity Database. The Desktop Client can be installed on additional laptops or desktops to allow administration and monitoring.

51 Camera Feed Analyzer

This window enables you to easily view and configure the live feeds from the cameras connected to this client. You can select any of the cameras connected to this Desktop Client from the drop-down menu at the top of the window.



For information about the options available from the **View** menu, see View Menu Options.

For information about the configuration options available from the **Camera Settings** button, (i.e. the gear icon) see the Camera Preferences tab of the Preferences Window.

The drop-down menu that's in the top right corner of the window allows you to select the Video Processing Mode to use for the camera video feed being shown in the center of the **Camera Feed Analyzer**. For information about the available Video Processing Modes, see Connect to a Video Feed.

The **Add to Video Feeds for continuous processing in the background** button (the button pointed to by the red arrow) allows you to transfer the camera video feed being shown in the **Camera Feed Analyzer** to a Video Recognition Gateway (VIRGO) video feed, where it will continue to be processed as a background process. The VIRGO feed will be created with the same Video Processing Mode that the original camera feed had, as well as the same Preference settings. A Video Feeds Window will automatically open, which allows you to manage all the currently existing VIRGO feeds.
Note: If you transfer a camera feed to VIRGO which already has a VIRGO feed processing it, then the already-existing VIRGO feed has its Video Processing Mode and settings updated to the transferred camera feed. (i.e. SAFR does not create a second, duplicate VIRGO feed associated with the same camera feed)

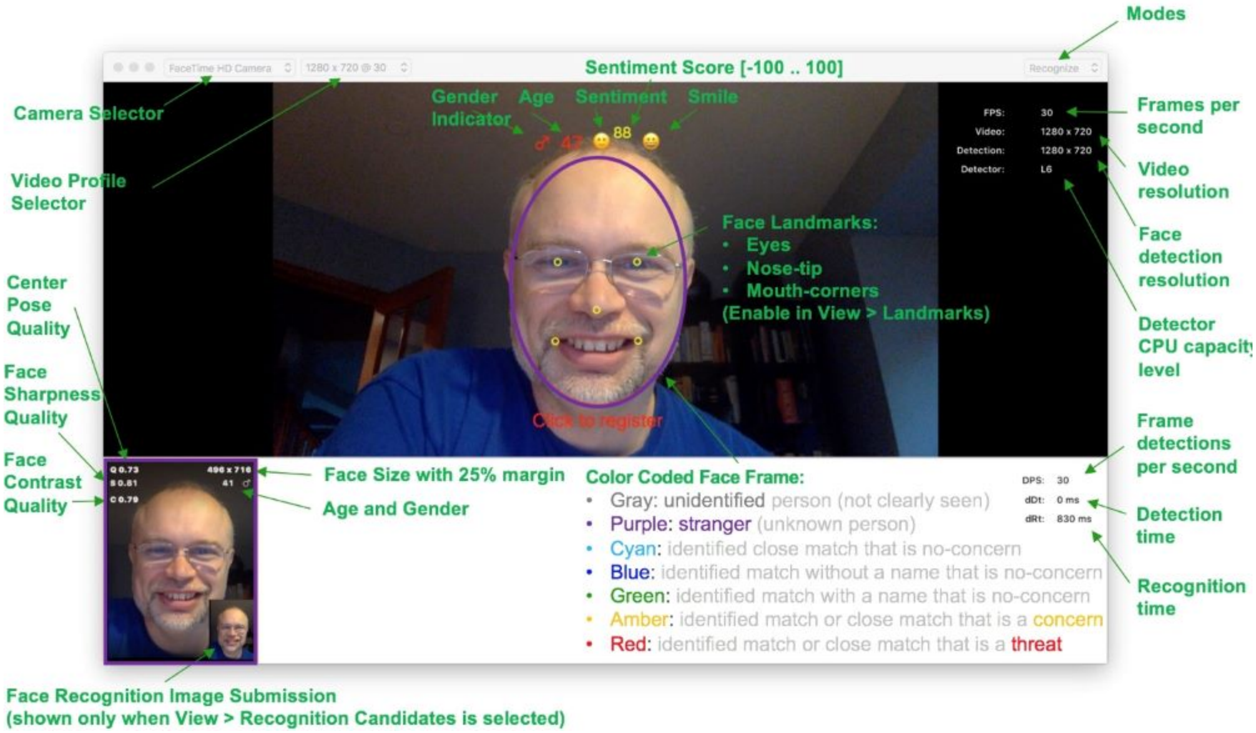
52 View Menu Options

You can customize the information displayed in the Desktop Client's Camera Feed Analyzer by clicking on the **View** menu option when the client has the *Camera Feed Analyzer* open. The following options are available:

- **Tracking Frames:** Available in both macOS and Windows. Enables colored indicator frames overlaid around detected faces and objects. See the color codes section here for a description of what each color indicates.
- **Attributes**: Available in both macOS and Windows. Enables the the selected attributes located in the *Detect* section of the Recognition preferences tab to be displayed above the faces seen in the *Camera* window's video feed.
- **Names**: Available in both macOS and Windows. Displays the name (if known) of recognized people below their faces.
- **Full-Screen Names**: Available in both macOS and Windows. Flashes recognized people's names or badge IDs (if known) over the entire video.
- **Face Landmarks**: Available in both macOS and Windows. Displays the five face landmarks (eyes, nose tip, and the corners of the mouth) on faces viewed in the *Camera Feed Analyzer*. See Interpret Video Feed Overlays for more information about face landmarks.
- **Detection List**: Only available in Windows. Displays a row at the bottom of the screen showing detected faces. Recognition details for each face also appear.
- **Motion Vectors**: Only available in macOS. Shows a motion vector for each tracked face.
- **Recognition Details**: Available in both macOS and Windows. Displays image quality metric values on the facial image(s) along the bottom of the *Camera* window. See Image Quality Metrics Guidance for more information about image quality metrics.

  On Windows selecting this option will automatically select the *Detection List* option (described above) as well.
- **Flash on Key Frames**: Only available in macOS. Causes a white backdrop to flash when a key frame is encountered while processing video files. A key frame is a location on a timeline that marks the beginning or end of a transition.
- **Video**:Available in both macOS and Windows. Disables the video, causing only the overlay elements to show.
- **Performance Metrics**: Available in both macOS and Windows. Displays feed window metrics, including frames per second, video and face detection resolution, and CPU capacity level.
- **Pose Liveness State Data**: Available in both macOS and Windows. Displays pose liveness data. For more information, see Pose Liveness Detection
- **Enter full-screen**: Available in Windows. Enables full screen mode. Ctrl-F exits full screen mode.
- **Enter lock-screen**: Available in Windows. Enables locked mode. While in locked mode all interactive controls are disabled, except those aplicable for the current video processing mode. A lock icon is added when in full screen which can be rapidly tapped 3 times to exit locked mode. Ctrl-L exits lock screen mode.

The following depicts various information that is displayed in the feed view, whether it is live from the camera or recorded video:



FPS: 30
Video: 1280 x 720
Detection: 1280 x 720
Detector: L6

Q 0.73
S 0.81
C 0.79
496 x 716
41 ♂

DPS: 30
dDt: 0 ms
dRt: 830 ms

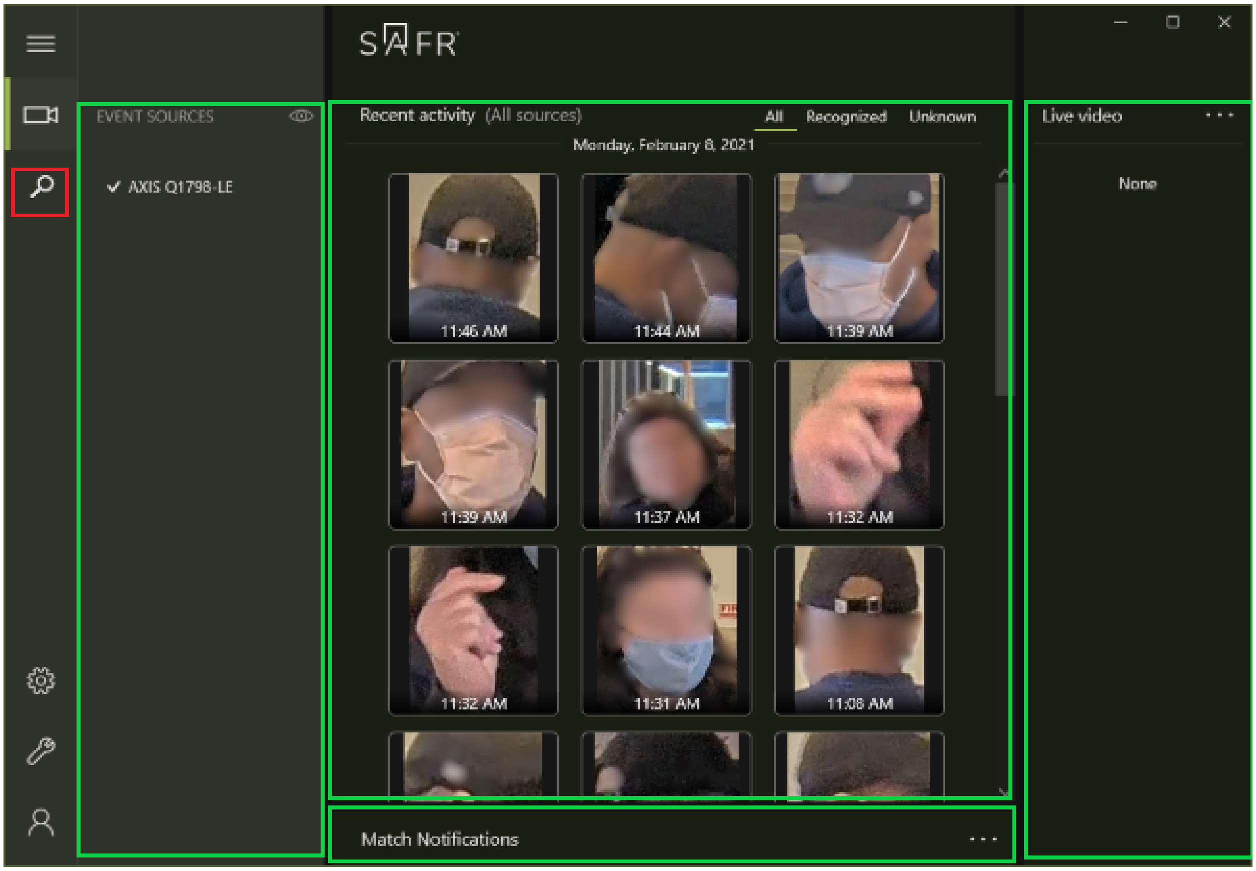
Color Coded Face Frame:

- Gray: unidentified person (not clearly seen)
- Purple: stranger (unknown person)
- Cyan: identified close match that is no-concern
- Blue: identified match without a name that is no-concern
- Green: identified match with a name that is no-concern
- Amber: identified match or close match that is a concern
- Red: identified match or close match that is a threat

53 Operator Console

The Operator Console window gives SAFR operators a unified UI with which to manage cameras, monitor events and activity, and view identity information. The Operator Console is only available on Windows.

In the screenshot below, the major panels of the console are outlined in green. The **Search by Image** button is outlined in red; clicking on this button opens the Search by Image Window.



53.1 Recent Activity Panel (Center Panel)

This panel, located in the center of the Operator Console window, displays thumbnail images of the events that your SAFR system has recently recorded. You can select which events will be displayed by selecting one of the 3 values at the upper right corner of the panel:

- **All**: All events will be displayed.
- **Recognized**: Only events triggered by people who are registered in the Person Directory will be displayed.
- **Unknown**: Only events triggered by people who aren't registered in the Person Directory will be displayed.

Hovering over an event thumbnail causes the thumbnail to be outlined. See Interpret Video Feed Overlays for information about what the different colors mean. **Note**: *Threats* and *Concerns* are always outlined in red and yellow, respectively.

Hovering over an event thumbnail usually also causes a button to appear. Clicking on the button causes the following menu items to appear:

- **View scene**: Brings up a dialogue with a still image of the event. This item is only offered if saving scene images has been enabled for the *Person Type* that triggered the event.

- **Add face**: Enroll the face in your Identity Database.
- **Find similar faces**: Switches to the Search by Image Window to search for similar faces that are already enrolled in your Identity Database.

53.2 Video Sources Panel (Left Panel)

This panel, located along the left side of the Operator Console window, displays all the video sources that are defined for your SAFR system. If a video source is associated with a VIRGO daemon, double clicking it will cause its live video feed to play in the Live Video Panel on the right of the Operator Console window.

Inactive sources are greyed out, while all the sources whose events are included in the Recent Activity Panel have a checkmark next to them. Note that you can check and uncheck sources on this window, but you must configure whether they're active or inactive elsewhere.

Sources with a yellow exclamation mark next to them are experiencing errors with at least one of the video feeds associated with it. (Although each source usually has only one video feed associated with it, it's possible to associate multiple feeds with a single source.) To troubleshoot video feed errors, do the following:

1. Hover over the source with an exclamation mark.
2. Note which video feed is experiencing the error.
3. Go to the Video Feeds Status window to get more information.

The Video Sources panel can be collapsed and expanded by clicking on the *Toggle panel* hamburger icon in the upper left corner of the Operator Console. The eye icon in the upper right of the panel expands and closes the live video panel on the far left of the Operator Console.

53.3 Live Video Panel (Right Panel)

This panel, located along the right side of the Operator Console Window, displays live video feeds.

Note: Live video feeds are only available for feeds associated with a VIRGO daemon. You can create VIRGO daemons in the Video Feeds Status Window.

You can right click on a live video to enable or disable tracking frames. Tracking frames are colored indicator frames overlaid around detected faces and objects. See the color codes section here for a description of what each color indicates.

This panel can be collapsed and expanded by clicking on the eye icon in the upper right-hand corner of the Video Sources Panel. You can view the live video feeds in their own dedicated window by clicking on the Context menu in the upper right corner of the panel, and selecting **Open panel in new window**.

53.4 Match Notifications (Bottom Panel)

This panel, located along the bottom of the Operator Console window, displays notifications of all the events that meet the specified levels of threat specified by the panel's Context menu in the upper right corner of the panel. By default, **Show all Threat** and **Show all Concern** are selected, while **Show all No-Concern** isn't.

The events are aggregated based on the person that triggered the event. Whenever a new event is triggered that meets the threat level criteria, that event is added to the Match Notification panel and the panel is automatically expanded. You can manually collapse or expand the panel by clicking on the panel's header bar.

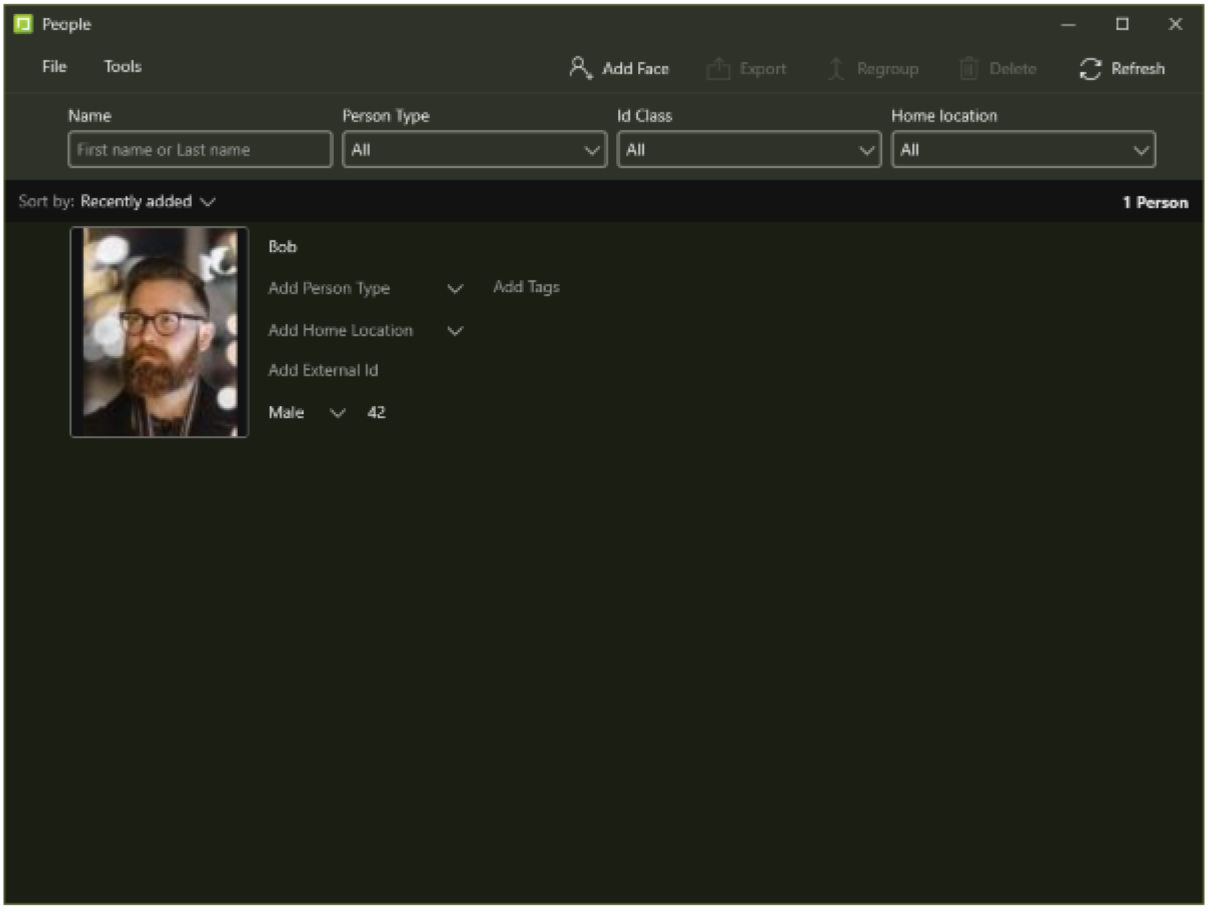
All new events are marked with a red circle in the upper left corner, so you can easily identify newly generated events. To clear the red circles, collapse the Match Notification panel.

54 People Window

This window allows you to view and manage registered people. Clicking on the **Add Face** button at the top of the window allows you to use saved image files (e.g. photos) to register people, while clicking on the **Export** button allows you to export the selected faces to image files.

The **Regroup** button allows you to tell SAFR that two different listings in the Identity Database are actually the same person, and they should be merged together. Merging two separate listings together greatly improve, as that particular person will have two reference images that the SAFR system will be able to use to better identify him or her.

Conversely, the **Regroup** button can be used to separate an Identity Database listing that had been previously erroneously merged from two separate listings.



Name: First name or Last name
Person Type: All
Id Class: All
Home location: All
Sort by: Recently added
1 Person
Bob
Male 42

54.1 People Filters

You can choose which people are listed in the bottom half of the window by specifying any of the following filters:

- **Name**: Filter based on the people's names.
- **Person Type**: Filter based on the people's `Person Types`.
- **Id Class**: Filter based on the people's `Id Classes` (i.e. their threat level).
- **Home location**: Filter basad on the people's `Home locations`.

54.2 Identity Database

All the people in the Identity Database that match your specified filters will be listed in the bottom half of the window. Any people that have an `Id Class` of “Threat” will be outlined in red, while any people with an `Id Class` of “Concern” will be outlined in yellow. You can right click on any of the facial images to reveal the following drop-down menu:

- **Add alternate face**: Allows you to add an alternate facial image for this identity record. You’ll be prompted to select an image file on your local hard drive.
- **Find similar faces**: Switches to the Search by Image Window to search for similar faces that are already enrolled in your Identity Database.

You can sort the people based on the following criteria:

- **Recently added**: Sort the people based on when they were registered to the Identity Database, with the latest registrants at the top.
- **First added**: Sort the people based on when they were registered to the Identity Database, with the earliest registrants at the top.
- **Last name A-Z**: Sort the people alphabetically based on their last names, with last names beginning with “A” at the top.
- **Last name Z-A**: Sort the people alphabetically based on their last names, with last names beginning with “Z” at the top.

54.2.1 Identity Attributes

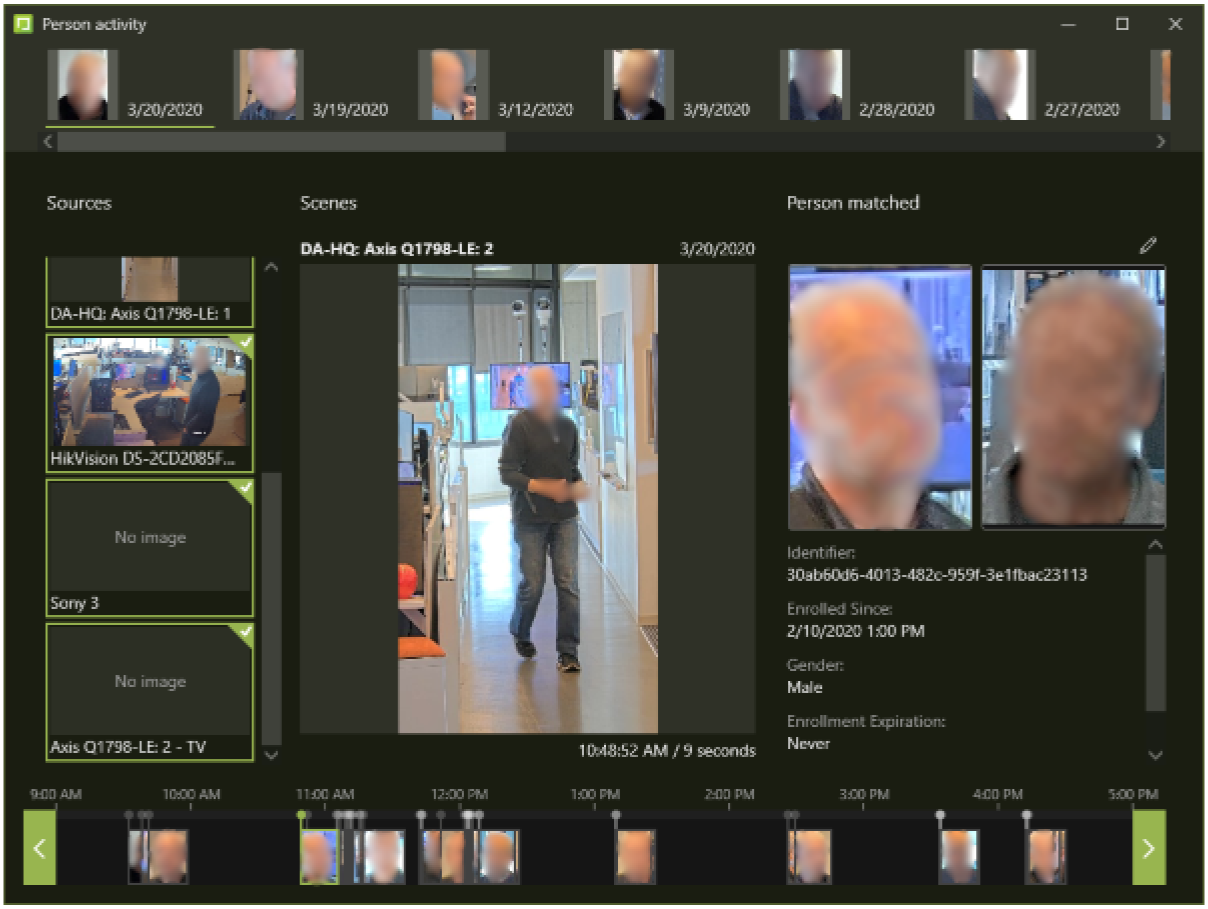
Some of the identity attributes are exposed in the default view of the Person Window, but if you double click on people’s faces, you can view and configure all the identity attributes. Note that most of their identity attributes will be empty until manually enter the information.

- **Identifier**: The person’s unique identifier within SAFR. This value is automatically assigned to them when they’re registered, and cannot be changed.
- **First Name**: The person’s first name.
- **Last Name**: The person’s last name.
- **Id Class**: The person’s threat level. (i.e. Threat, Concern, or No-Concern)
- **Person Type**: The person’s Person Type. Person Types are groupings that you define to differentiate the people registered in your Identity Database. (e.g. “student”, “teacher”, and “staff”)
- **Gender**: The person’s gender.
- **DateofBirth**: The person’s date of birth.
- **Moniker**: Used to realize two-factor authentication with visual badges.
- **External Id**: If the person has been imported from another database, this value can be used to track the identity in both databases.
- **Company**: The company the person works for.
- **Home location**: The person’s Home Location. Home Locations, much like Person Types, are labels that you define to help differentiate the people registered in your Identity Database.
- **Phone**: The person’s phone number.
- **Email**: The person’s email address.
- **Tags**: Any custom tags that you have defined. People can have multiple tags assigned to them.
- **Enrollment date**: The date when the person was registered.
- **Enrollment expiration**: The expiration date of the person’s enrollment.
- **Enrolled site**: The site where the person was enrolled.
- **Enrolled source**: The camera that enrolled the person.
- **Last modified**: The time when this person record was last modified. (e.g. an attribute was updated, a more recent reference image was uploaded, etc.)
- **Modified by**: The user that made the last modification to this person record.
- **Modified site**: The site from which this person record was last modified.

Note: When you double click on a person, there will be a View person activity button just below their face if they have been active recently.

55 Person Activity Window

This window allows you to view the activity over time of a particular person.



Identifier:
30ab60d6-4013-482c-959f-3e1fbac23113

Enrolled Since:
2/10/2020 1:00 PM

Gender:
Male

Enrollment Expiration:
Never

55.1 Person Activity Panel (Top Panel)

The Person Activity Panel shows the dates when the person triggered one or more events. Click on any event to populate the Activity Timeline Panel with all the events of that date.

55.2 Sources Panel (Left Panel)

Shows all the cameras associated with the event(s) that you've selected in the Person Activity Panel and/or the Activity Timeline Panel.

55.3 Scenes Panel (Center Panel)

Shows event scenes from the event you selected in the Activity Timeline Panel if available.

55.4 Person Matched Panel (Right Panel)

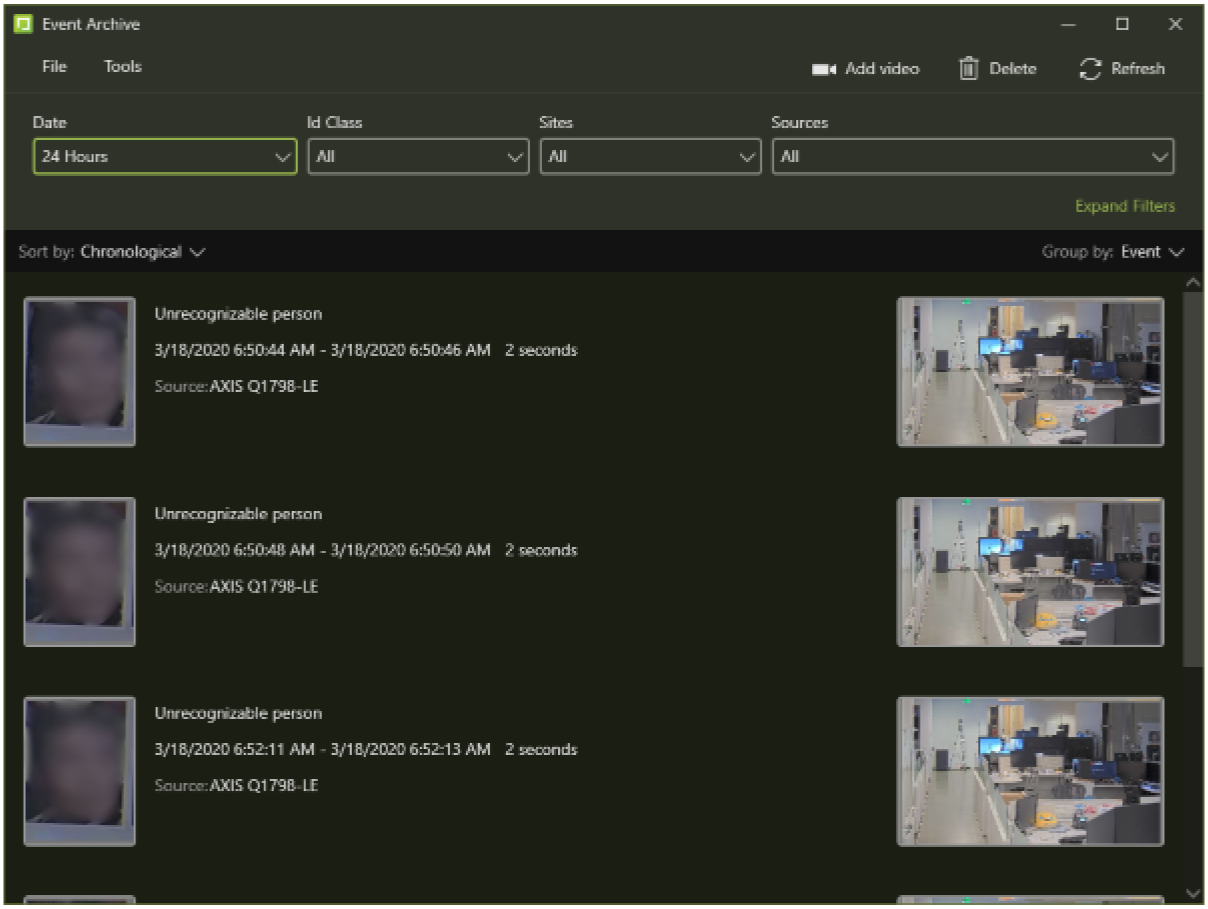
Shows the person whose activity is being described by this window.

55.5 Activity Timeline Panel (Bottom Panel)

Shows the times of day when events were triggered on the date specified by the event you selected in the Person Activity Panel. Selecting the events on this timeline populates the Scenes Panel with event scenes, if available.

56 Events Window

This window allows you to view and manage recorded events. By clicking on the **Add video** button at the top of the window you can also use saved video files to generate events and register people.



Date: 24 Hours

Id Class: All

Sites: All

Sources: All

Sort by: Chronological

Group by: Event

Unrecognizable person
3/18/2020 6:50:44 AM - 3/18/2020 6:50:46 AM 2 seconds
Source: AXIS Q1798-LE

Unrecognizable person
3/18/2020 6:50:48 AM - 3/18/2020 6:50:50 AM 2 seconds
Source: AXIS Q1798-LE

Unrecognizable person
3/18/2020 6:52:11 AM - 3/18/2020 6:52:13 AM 2 seconds
Source: AXIS Q1798-LE

56.1 Event Filters

You can filter the events based on the following criteria. The first 4 filters are always visible, while the others become visible when you click on the **Expand Filters** button.

- **Date**: The date when the event was recorded.
- **Id Class**: The threat level of the person that triggered the event.
- **Sites**: The camera or set of cameras that recorded the event. **Note**: Usually `Sites` are set to multiple cameras.
- **Sources**: The camera or set of cameras that recorded the event. **Note**: Usually `Sources` are set to single cameras.
- **Name**: The name of the person that triggered the event.
- **Person Type**: The `Person Type` of the person that triggered the event.
- **Gender**: The gender of the person that triggered the recording of the event.
- **Tenure**: The date when the identity that triggered the event was registered to the Identity Database.
- **Shortest Gap**: If a person is viewed by one or more cameras multiple times within this time period, all those appearances are considered the same event.
- **Shortest Duration**: The minimum event duration, in milliseconds, to include in your search.
- **Disparate Sources**: If a person is viewed by multiple cameras at the same time, all those appearances are considered the same event when this filter is enabled.

56.2 Event Archive

All the events that match your specified filters will be listed in the bottom half of the window. All listed events will display an image of the person who triggered the event on the far left of the listing, the reference image for that person in the Identity Directory (if available), information about the event, and an image from the event along the right (if available).

You can right click on any of the event face images to reveal the following drop-down menu:

- **Add face**: Enrolls the face in your Identity Database.
- **Find similar faces**: Switches to the Search by Image Window to search for similar faces that are already enrolled in your Identity Database.
- **Delete event**: Deletes the event. Note that deleted events are removed every 5 minutes, so there may be a short delay after you click the **Delete** button before the event actually disappears.

Note: For all events triggered by people registered to your Identity Directory, there will be a `View person activity` button just below the event information that will take you to the Person Activity Window.

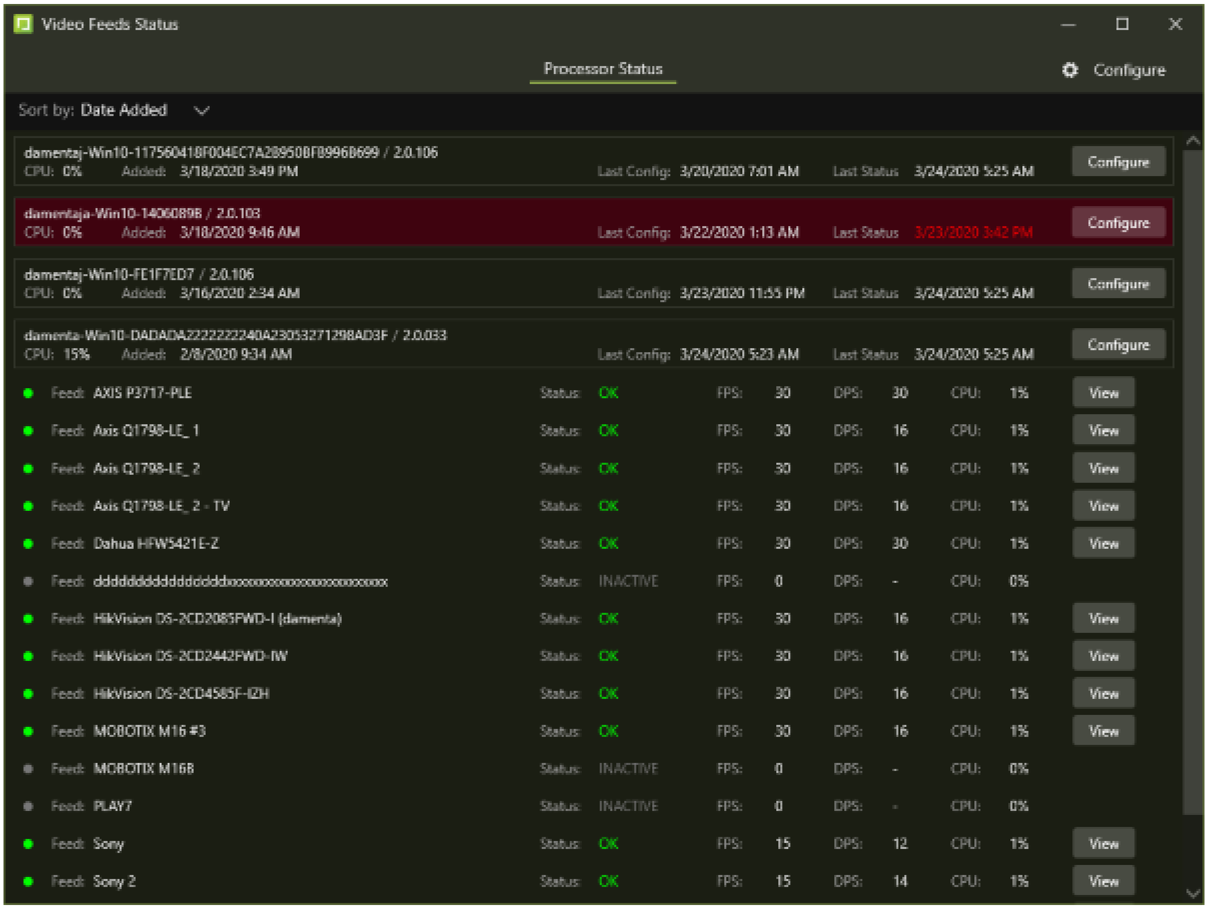
You can sort events by the following criteria:

- **Chronological**: Sort the events based on when they were recorded.
- **Duration**: Sort the events based on how long they last.
- **Name**: Sort the events based on the name of the person that triggered the event, if known.

You also have the option to group events by the person that triggered the event by selecting **Group by: Person** on the right side of the window. If you instead select **Group by: Event**, then the events aren't grouped by person and will instead only be sorted based on the sorting criteria that you have selected.

57 Video Feeds Window

This window allows you to view and configure your Video Recognition Gateway (VIRGO) video feeds. Any feed that's highlighted in red is currently inactive.



Sort by: Date Added

```
damentaj-Win10-117560418F004EC7A289508F8996B699 / 2.0.106
CPU:  0%     Added:  3/18/2020 3:49 PM       Last Config:  3/20/2020 7:01 AM    Last Status  3/24/2020 5:25 AM
damentaja-Win10-1406089B / 2.0.103
CPU:  0%     Added:  3/18/2020 9:46 AM       Last Config:  3/22/2020 1:13 AM    Last Status  3/23/2020 3:42 PM
damentaj-Win10-FE1F7ED7 / 2.0.106
CPU:  0%     Added:  3/16/2020 2:34 AM       Last Config:  3/23/2020 11:55 PM   Last Status  3/24/2020 5:25 AM
damenta-Win10-DADADA2222222240A23053271298AD3F / 2.0.033
CPU:  15%    Added:  2/8/2020 9:34 AM        Last Config:  3/24/2020 5:23 AM    Last Status  3/24/2020 5:25 AM
● Feed:  AXIS P3717-PLE                         Status:  OK         FPS:  30   DPS:  30   CPU:  1%
● Feed:  Axis Q1798-LE_ 1                       Status:  OK         FPS:  30   DPS:  16   CPU:  1%
● Feed:  Axis Q1798-LE_ 2                       Status:  OK         FPS:  30   DPS:  16   CPU:  1%
● Feed:  Axis Q1798-LE_ 2 - TV                  Status:  OK         FPS:  30   DPS:  16   CPU:  1%
● Feed:  Dahua HFW5421E-Z                       Status:  OK         FPS:  30   DPS:  30   CPU:  1%
● Feed:  ddddddddddddddddxxxxxxxxxxxxxxxxxxxxxx Status:  INACTIVE   FPS:  0    DPS:  -    CPU:  0%
● Feed:  HikVision DS-2CD2085FWD-I (damenta)    Status:  OK         FPS:  30   DPS:  16   CPU:  1%
● Feed:  HikVision DS-2CD2442FWD-IW             Status:  OK         FPS:  30   DPS:  16   CPU:  1%
● Feed:  HikVision DS-2CD4585F-IZH              Status:  OK         FPS:  30   DPS:  16   CPU:  1%
● Feed:  MOBOTIX M16 #3                         Status:  OK         FPS:  30   DPS:  16   CPU:  1%
● Feed:  MOBOTIX M16B                           Status:  INACTIVE   FPS:  0    DPS:  -    CPU:  0%
● Feed:  PLAY7                                  Status:  INACTIVE   FPS:  0    DPS:  -    CPU:  0%
● Feed:  Sony                                   Status:  OK         FPS:  15   DPS:  12   CPU:  1%
● Feed:  Sony 2                                 Status:  OK         FPS:  15   DPS:  14   CPU:  1%
```

The feeds can be sorted by one of the following criteria:

- **Date Added**: Sort the feeds based on the date when they were added.
- **Status Date**: Sort the feeds based on the date when the feeds' status last changed.
- **Client Id**: Sort the feeds based on the Id's of the clients to which the feeds are connected.
- **Tenant**: Sort the feeds based on the feeds' tenants.
- **Version**: Sort the feeds based on the feeds' version.

You can click on the **Configure** button to the right of a feed to configure it. Similarly, you can click on the **View** button the right of a camera to see a live video view of the camera.

57.1 Add a VIRGO Video Feed

To add a VIRGO video feed, there must be at least one active video feed connected to this Desktop Client. See Connect Cameras to SAFR for information about how to connect cameras to the client.

Note: You can't add a VIRGO daemon for a video feed that is currently being displayed by the Camera Feed Analyzer.

To add a VIRGO video feed, do the following:

1. Click the **Configure** button on the active video feed that you want to associate the VIRGO daemon with.

2. Hover your mouse over the **feeds** entry. You'll see a + button and a **-** button. Click the + button. You'll be prompted for the following information:
   - **Feed Name**: Enter any name for the video feed you wish.
   - **Mode**: Select the video processing mode that you want the VIRGO daemon to operate in. For a description of the video processing modes, see here.
   - **Processor**: Select the machine that will continuously process the video feed.
   - **Apply Mode Customizations from Preferences**: Enable if you want the Desktop Client preferences applied to the new VIRGO daemon.
3. Press the **Add** button. The VIRGO video feed has been created.

58 Search by Image Window

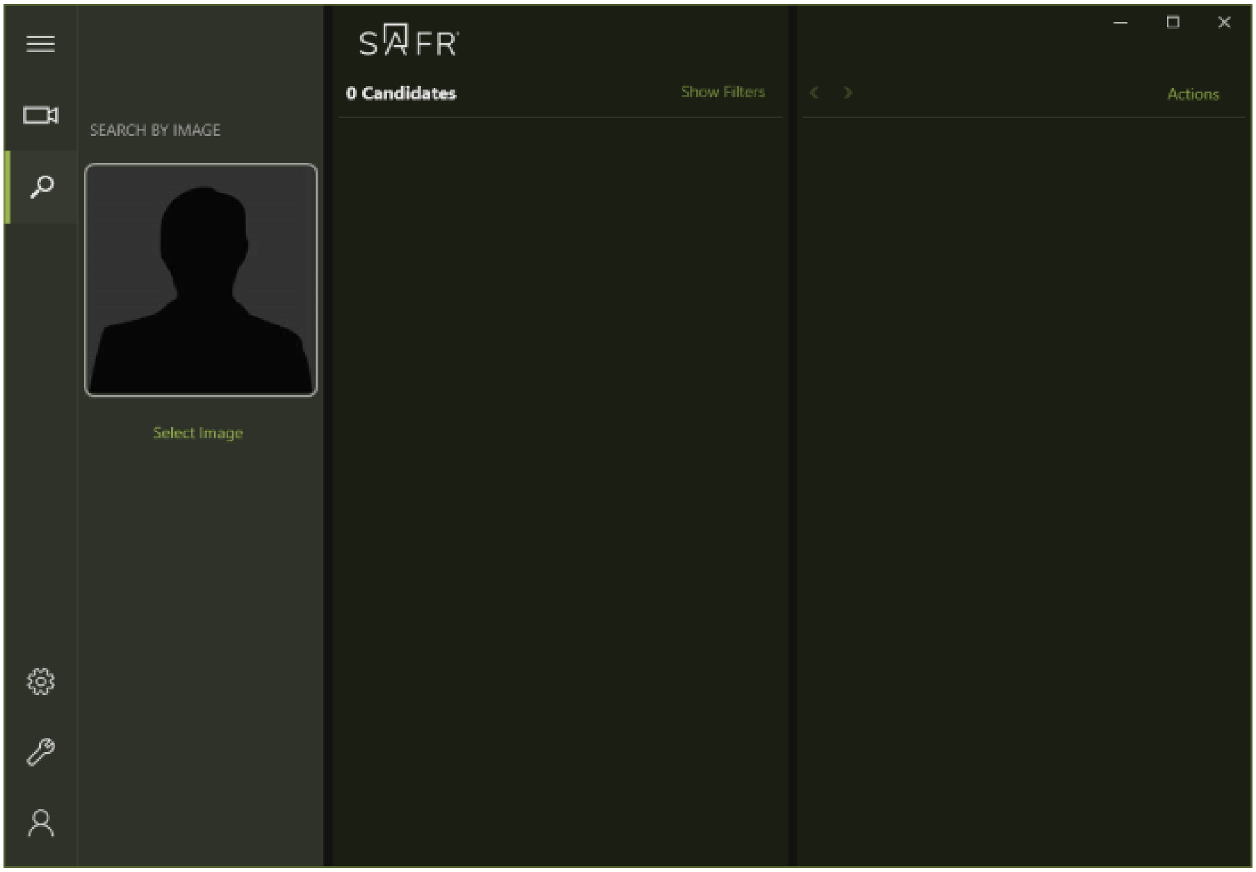
This window allows you to use facial images to search either the Identity Database or the Event Archive. The Search by Image Window is only available on Windows.

58.1 Accessing the Search by Image Window

There are 5 ways to access the Search by Image Window.

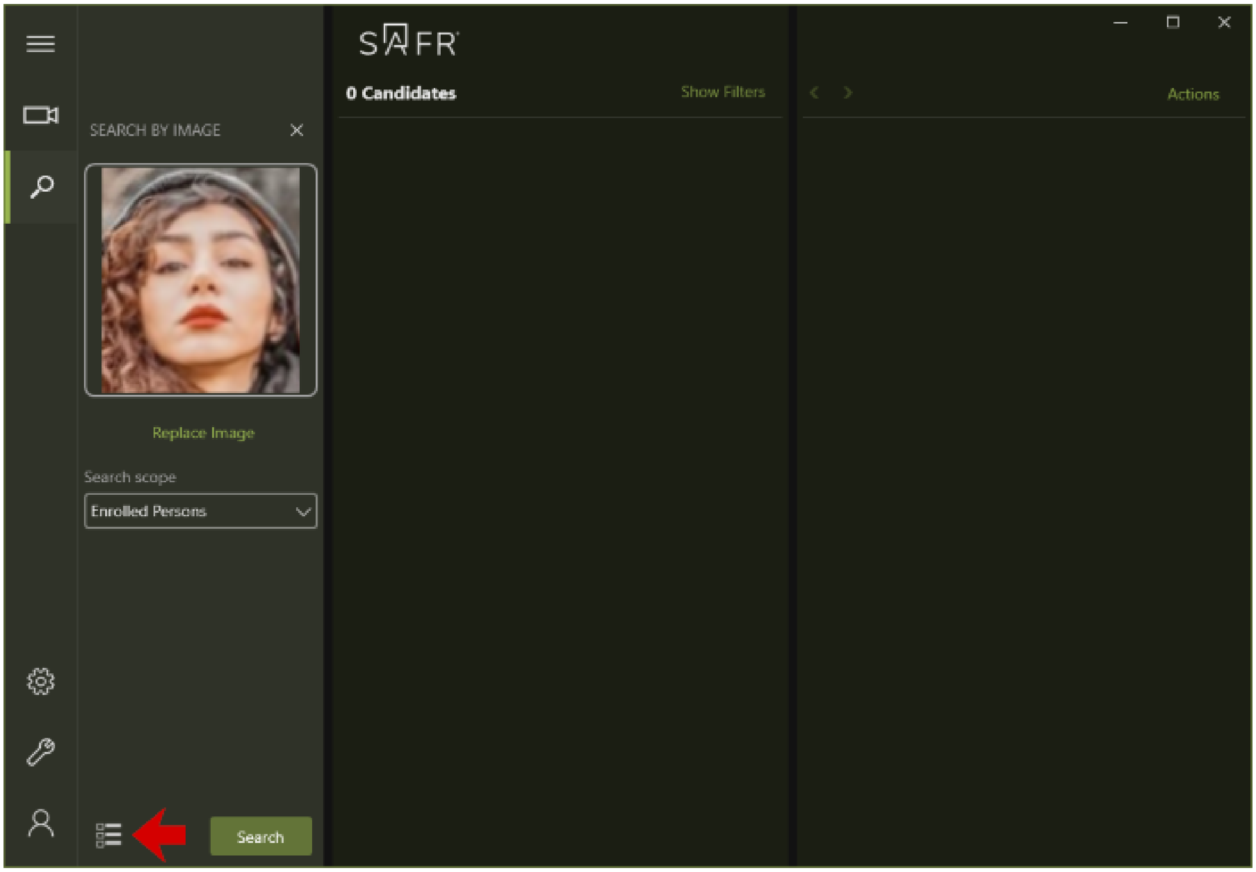
- Select **Search by image. . .** from the **Tools** drop down menu.
- Click on the magnifying glass button on the far left of the Operator Console.
- Right click on a face in the *Recent Activity Panel* (the center panel) of the Operator Console, and select **Find similar faces** from the drop down menu.
- Right click on a face in the People Window, and select **Find similar faces** from the drop down menu.
- Right click on a face in the Events Window, and select **Find similar faces** from the drop down menu.

58.2 Functionality



58.2.1 Search Image Panel

You can click on **Select Image** to select an image file saved on your hard drive. After selecting the file, you'll be prompted to choose which face within the image to use for the search.



Search scope: Enrolled Persons

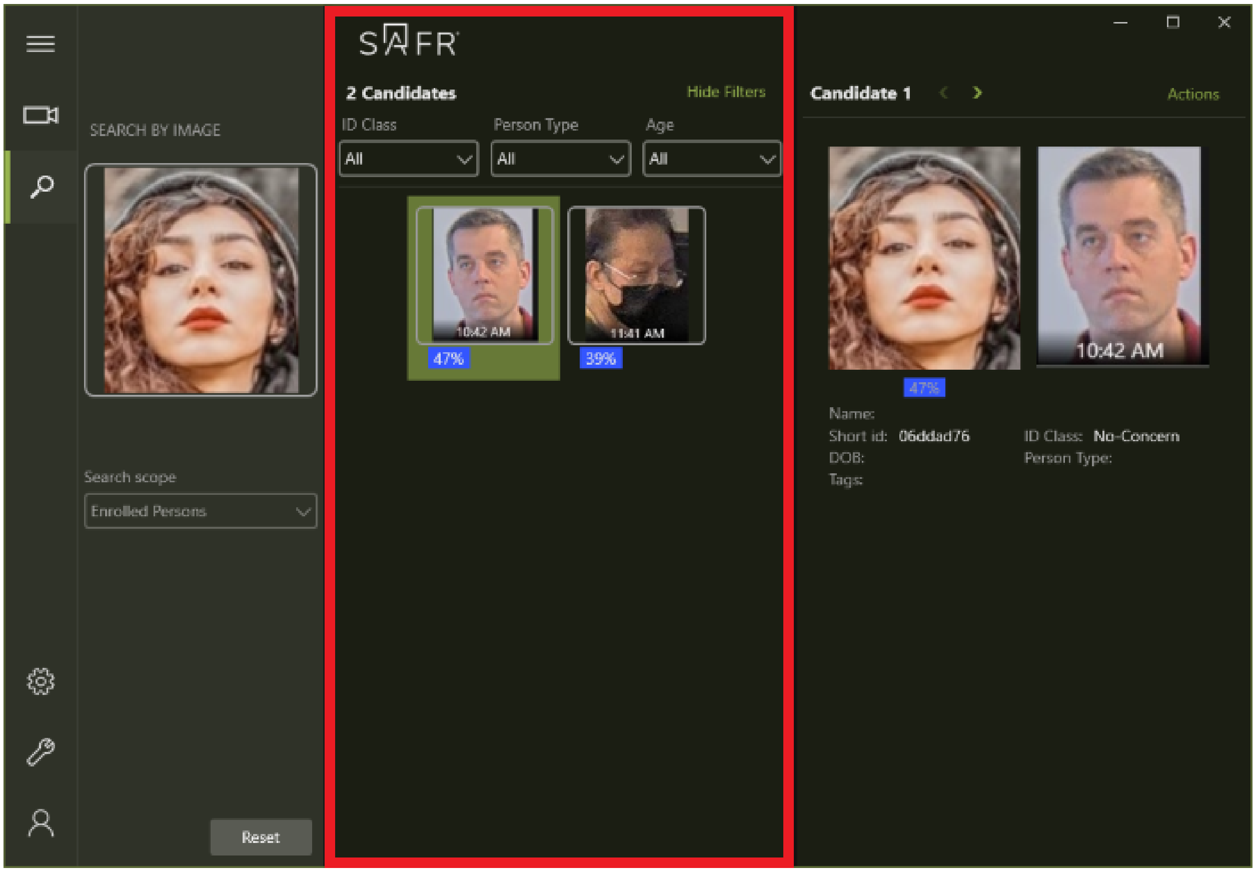
You can click on the **Set candidate list size** button (indicated with the red arrow) to adjust how many potential matches will be displayed.

You'll now select what type of search you want to execute by choosing one of the **Search scope** options and then clicking the **Search** button.

- **Enrolled Persons**: Searches the Identity Database for registered faces that are similar to the search image.
- **Events**: Searches the Event Archive for events containing a face that is similar to the search image.

58.2.2 Candidates Panel

The middle panel shows the possible matches for your search query.



2 Candidates

ID Class: All
Person Type: All
Age: All

Search scope: Enrolled Persons

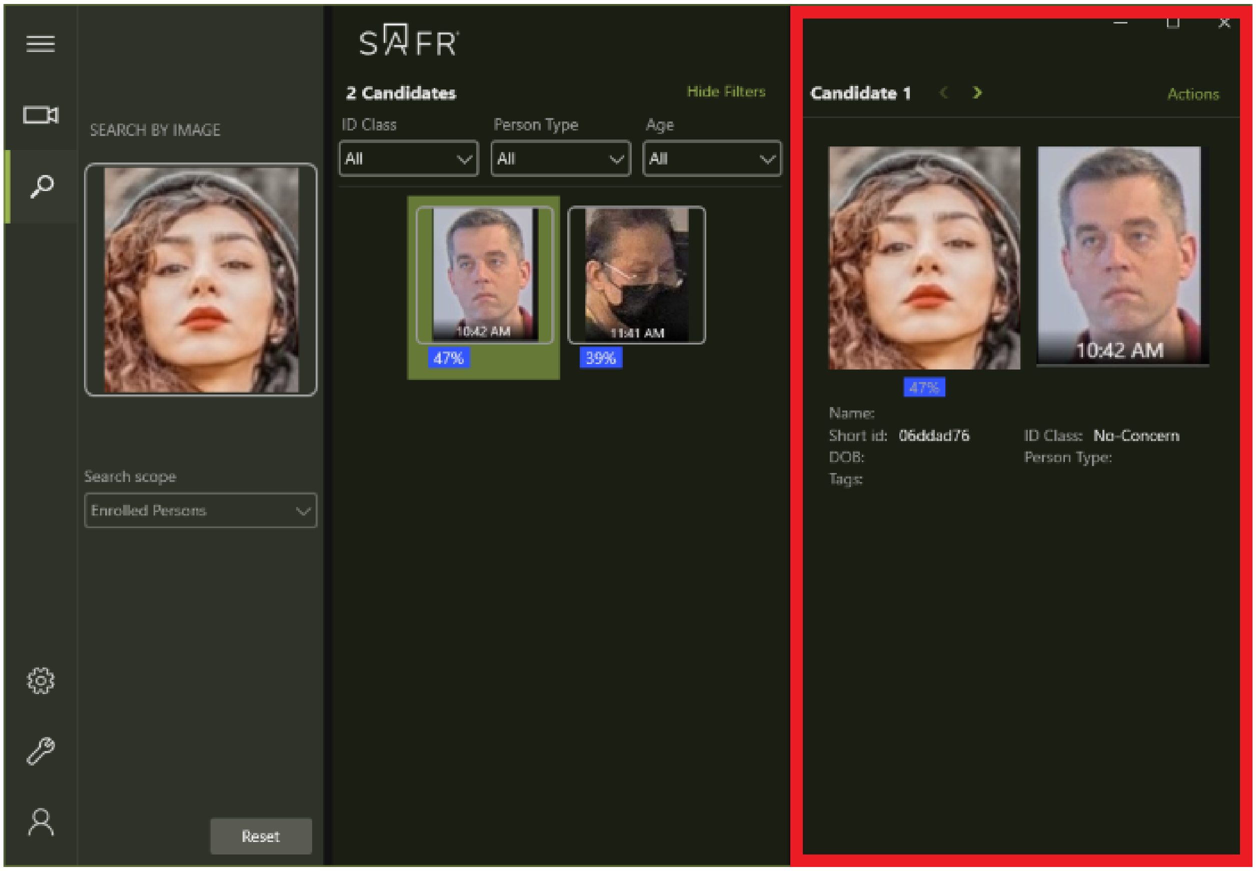
Name:
Short id: 06ddad76
ID Class: No-Concern
DOB:
Person Type:
Tags:

The candidates can be filtered based on *ID Class*, *Person Type*, and/or *Age* by clicking on **Show Filters**. The blue numbers below the candidates show the percent match. You can interpret the percent match values as shown below.

- 100% = Certain match.
- 93%-99% = Close match; medium to high confidence that it's the same person.
- 86%-92% = Possible match; low confidence that it's the same person.
- 82%-85% = Similar face; no confidence that it's the same person.
- 0-82% = Different face.

You can click on any candidate to get more detailed information on that candidate in the comparison panel.

58.2.3 Comparison Panel



2 Candidates

ID Class: All

Person Type: All

Age: All

Search scope: Enrolled Persons

Name:

Short id: 06ddad76

DOB:

Tags:

ID Class: No-Concern

Person Type:

In addition to seeing the selected candidate's attributes shown above, you can perform additional tasks by clicking on **Actions**.

- **View person activity**: Opens the Person Activity Window for the candidate, which shows all the events featuring the candidate.
- **Show quality metrics**: Shows the following image quality metrics for both the search image and candidate. See the Image Quality Metrics Guidance documentation for information about these metrics.
  - **Center pose**
  - **Sharpness**
  - **Contrast**
  - **Face size**
  - **Occlusion**
- **Edit person profile**: Edit any of the candidate's identity attributes. See the People Window documentation for information about the attributes.

59 Account Preferences

The Account Preferences tab allows you to configure the user account currently logged into the Desktop Client. You can change which user is logged in entirely by clicking on the `Logout` button next to the *User Identifier*.

- **User Identifier**: The username of the logged-in user.
- **User Password**: macOS only. The password for the logged-in user.
- **User Directory**: The user directory to use. (The default *user directory* is `main`.)
- **User Site**: macOS only. The default *Site* label to use for events generated by cameras connected to this Desktop Client. This field is optional.
- **Default Site**: Windows only. The default *Site* label to use for events generated by cameras connected to this Desktop Client. This field is optional.
- **User Source**: macOS only. The default *Source* label to use for events generated by cameras connected to Desktop Client. This field is optional.
- **Similar Directory**: macOS only. The directory to use when using the *Similar* video processing mode. If this setting is left blank, then the *User Directory* is used when using *Similar* mode. See Connect to a Video Feed for more information on video processing modes.
- **Environment**: macOS only. Determines which environment your client contacts. The possible values for this field are as follows:
  - *SAFR Developer Cloud*: Internal use only.
  - *SAFR Partner Cloud*: Internal use only.
  - *SAFR Cloud*: Used for cloud deployments.
  - *SAFR Custom*: Used for local deployments. If you select *SAFR Custom*, you will be asked to provide the URLs for the primary SAFR Server services.
- **Server Location**: Windows only. The server associated with the user's account.
- **Report Status**: macOS only. When enabled, video feeds will report their status to the SAFR Server even when a VIRGO daemon isn't associated with it.
  - **Allow Remote Viewing**: macOS only. Enables video feeds to be viewed by any SAFR client, even when a VIRGO daemon isn't associated with them.

60 Camera Preferences

The Camera Preferences tab allows you to add, remove, or configure cameras connected to this Desktop Client. Cameras that have already been automatically discovered are displayed here.

SAFR normally auto-detects all integrated (built-in) cameras, USB-connected cameras, and IP (internet protocol) cameras that support the ONVIF (Open Network Video Interface Forum) protocol as long as they are present on the same network to which the Desktop Client is connected. For more about ONVIF and the ONVIF specification, see the ONVIF home page.

60.1 ONVIF Cameras

- Each camera discovered via ONVIF must have a username and password.
- Be sure that the camera has at least one ONVIF user with administrative privileges, or ONVIF authentication will not work.
- ONVIF video profiles configured on the camera will be available to select in the live video recognition view.
- The date and time configured on the camera must be within five seconds of the system time SAFR is running on.

60.2 Camera Preferences

To configure a camera, select any of the connected cameras in the left panel. If you need to add a camera, see Connect Cameras to SAFR. When a camera is selected, a set of preferences will appear on the right of the window. Which preferences are exposed depends on what kind of camera is selected.

60.2.1 USB Camera Preferences

USB cameras have the following preference settings.

- **Source**: Use this field to override the default source label for the camera. For example, you might want to change a default source label to West Hall or a more meaningful label for your specific environment. Multiple cameras can have the same source label. For example, you could have several cameras labeled as South Exit, or you could label each camera differently ("South Exit 1", "South Exit 2", or other labels that make sense for you and your SAFR environment).
- **Address**: Address of the camera.
- **User**: User name used to log into the camera.
- **Password**: Password used to log into the camera.
- **Contrast Enhancement**: When selected, enhances low-light images and videos by enhancing the contrast.
    - **Low Light Threshold**: Determines the level of light when the exposure boost will be applied. If an image or video has more light than the threshold allows for, then the exposure boost will not be applied.
    - **Exposure Boost**: Determines how much to boost the contrast.
- **Front facing**: Indicates if the camera is front-facing.
- **Rotate Image**: Use the dropdown menu to set a value that rotates to correct for camera orientation. Typically, this correction can be done directly by the camera, but if not, you can adjust it here. Recognition does not work properly unless the camera's image is properly oriented.
- **Enforce Low Latency**: This option is only available for integrated and USB cameras. *Enforce Low Latency* optimizes display and processing by allowing the video frame rate to drop if CPU resources are low.

  This may be needed for some 4K webcam models or any cameras that support particularly high frame rates.
- **Direction of Travel Recognition**: When any of the 4 travel distance settings below are enabled, the Desktop Client will generate a direction of travel event when a person's face travels further than the specified percentage within the camera view field. For example, if `Left travel distance` were set to

50, then a direction of travel event would be generated if somebody's face entered the camera view field from the right edge of the frame, and then moved to the left more than 50% of the camera field view. Note that you can configure multiple travel distance settings at the same time, and a direction of travel event will be triggered if the condition specified by any of the configured travel distance settings were met.

The 4 boundary settings allow you to shrink the size of the camera view for the purpose of generating direction of travel events. If you set one or more of the boundary settings, SAFR will not begin reporting or calculating direction of travel events until the object is within the specified boundaries. The boundary settings can be useful when people are hovering right at the edge of the camera's field of view (e.g. a doorman at the entrance to a hotel).

- **Left travel distance**: Generates a direction of travel event when a face enters from the right edge of the camera view field and travels more than the specified percentage across the field towards the left edge.
- **Right travel distance**: Generates a direction of travel event when a face enters from the left edge of the camera view field and travels more than the specified percentage across the field towards the right edge.
- **Up/Away travel distance**: Generates a direction of travel event when a face enters from the bottom edge of the camera view field and travels more than the specified percentage across the field towards the top edge.
- **Down/Towards travel distance**: Generates a direction of travel event when a face enters from the top edge of the camera view field and travels more than the specified percentage across the field towards the bottom edge.
- **Left boundary**: Specifies the percentage of the left side of the camera view field to exclude from direction of travel event reporting.
- **Right boundary**: Specifies the percentage of the right side of the camera view field to exclude from direction of travel event reporting.
- **Top boundary**: Specifies the percentage of the top side of the camera view field to exclude from direction of travel event reporting.
- **Bottom boundary**: Specifies the percentage of the bottom side of the camera view field to exclude from direction of travel event reporting.

60.2.2 Xenia Camera Preferences

Only available in Windows.

Note: Xenia cameras can connect to SAFR only if you selected the Ximea camera extension when you installed SAFR.

Xenia cameras have the following preference settings.

- **Source**: Use this field to override the default source label for the camera. For example, you might want to change a default source label to West Hall or a more meaningful label for your specific environment. Multiple cameras can have the same source label. For example, you could have several cameras labeled as South Exit, or you could label each camera differently ("South Exit 1", "South Exit 2", or other labels that make sense for you and your SAFR environment).
- **Address**: Address of the camera.
- **User**: User name used to log into the camera.
- **Password**: Password used to log into the camera.
- **Front facing**: Indicates if the camera is front-facing.
- **Rotate image**: Use the dropdown menu to set a value that rotates to correct for camera orientation. Typically, this correction can be done directly by the camera, but if not, you can adjust it here. Recognition does not work properly unless the camera's image is properly oriented.
- **Max frame rate**: Select a value from the dropdown menu to set the max frame rate for the Xenia camera.
- **Contrast Enhancement**: When selected, enhances low-light images and videos by enhancing the contrast.

  - **Low Light Threshold**: Determines the level of light when the exposure boost will be applied. If an image or video has more light than the threshold allows for, then the exposure boost will not be applied.
  - **Exposure Boost**: Determines how much to boost the contrast.
- **Direction of Travel Recognition**: When any of the 4 travel distance settings below are enabled, the Desktop Client will generate a direction of travel event when a person's face travels further than the specified percentage within the camera view field. For example, if `Left travel distance` were set to 50, then a direction of travel event would be generated if somebody's face entered the camera view field from the right edge of the frame, and then moved to the left more than 50% of the camera field view. Note that you can configure multiple travel distance settings at the same time, and a direction of travel event will be triggered if the condition specified by any of the configured travel distance settings were met.
  The 4 boundary settings allow you to shrink the size of the camera view for the purpose of generating direction of travel events. If you set one or more of the boundary settings, SAFR will not begin reporting or calculating direction of travel events until the object is within the specified boundaries. The boundary settings can be useful when people are hovering right at the edge of the camera's field of view (e.g. a doorman at the entrance to a hotel).
  - **Left travel distance**: Generates a direction of travel event when a face enters from the right edge of the camera view field and travels more than the specified percentage across the field towards the left edge.
  - **Right travel distance**: Generates a direction of travel event when a face enters from the left edge of the camera view field and travels more than the specified percentage across the field towards the right edge.
  - **Up/Away travel distance**: Generates a direction of travel event when a face enters from the bottom edge of the camera view field and travels more than the specified percentage across the field towards the top edge.
  - **Down/Towards travel distance**: Generates a direction of travel event when a face enters from the top edge of the camera view field and travels more than the specified percentage across the field towards the bottom edge.
  - **Left boundary**: Specifies the percentage of the left side of the camera view field to exclude from direction of travel event reporting.
  - **Right boundary**: Specifies the percentage of the right side of the camera view field to exclude from direction of travel event reporting.
  - **Top boundary**: Specifies the percentage of the top side of the camera view field to exclude from direction of travel event reporting.
  - **Bottom boundary**: Specifies the percentage of the bottom side of the camera view field to exclude from direction of travel event reporting.

60.2.3 IP Camera Preferences

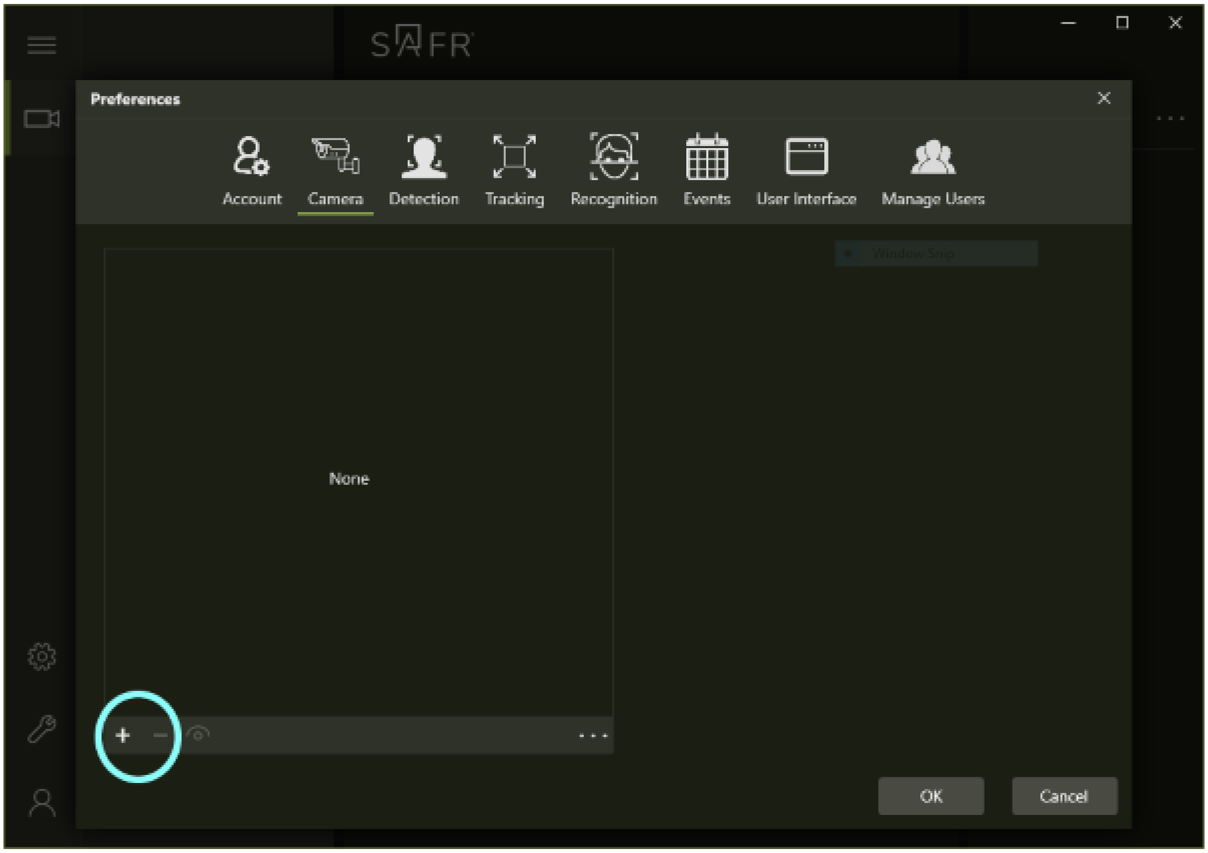
IP cameras have the following preference settings.

- **Source**: Use this field to override the default source label for the camera. For example, you might want to change a default source label to West Hall or a more meaningful label for your specific environment. Multiple cameras can have the same source label. For example, you could have several cameras labeled as South Exit, or you could label each camera differently ("South Exit 1", "South Exit 2", or other labels that make sense for you and your SAFR environment).
- **Address**: Address of the camera.
- **User**: User name used to log into the camera.
- **Password**: Password used to log into the camera.
- **RTSP Transport Protocol:** For the lowest latency, select UDP (User Datagram Protocol). If network packet loss is an issue, select TCP (Transmission Control Protocol).
  Generally, UDP is a faster best effort communication system, whereas TCP is more reliable but slower. For example, if network connectivity is a concern for a particular camera, you might want to change this to TCP. Otherwise, UDP should be adequate in most situations.

- **Contrast Enhancement**: When selected, enhances low-light images and videos by enhancing the contrast.
  - **Low Light Threshold**: Determines the level of light when the exposure boost will be applied. If an image or video has more light than the threshold allows for, then the exposure boost will not be applied.
  - **Exposure Boost**: Determines how much to boost the contrast.
- **Lens Correction**: Enable lens correction to help correct the fisheye effect for very wide-angle lenses. This also improves recognition accuracy although is not needed in most cases.
  - **Coefficient K1**: The "K1" lens correction factor.
  - **Coefficient K2**: The "K2" lens correction factor.
- **Enforce timing for video frames**: When enabled, playback frame syncing to the video clock is enforced.
- **Enable mirroring**: macOS only. When enabled, the camera's video feed is mirrored.
- **Front facing**: Indicates if the camera is front-facing.
- **Rotate Image**: Use the dropdown menu to set a value that rotates to correct for camera orientation. Typically, this correction can be done directly by the camera, but if not, you can adjust it here. Recognition does not work properly unless the camera's image is properly oriented.
- **Back Channel**: When the connected camera is a Mobotix camera, you can set this field to *Mobotix MX* in order to have SAFR report *STRANGER* and *RECOGNIZED* event types to the camera. This feature is necessary if you want to make use of the Mobotix app.
  When the connected camera isn't a Mobotix camera, this setting doesn't have any effect.
  - **Cash Point**: This value must match the configured *cash point* within the Mobotix app. If this *Cash Point* setting doesn't match the cash point within the Mobotix app, the back channel won't work.
- **Frame buffer size**: Sets the size of the camera's frame buffer for buffering network streams.
- **Direction of Travel Recognition**: When any of the 4 travel distance settings below are enabled, the Desktop Client will generate a direction of travel event when a person's face travels further than the specified percentage within the camera view field. For example, if `Left travel distance` were set to 50, then a direction of travel event would be generated if somebody's face entered the camera view field from the right edge of the frame, and then moved to the left more than 50% of the camera field view. Note that you can configure multiple travel distance settings at the same time, and a direction of travel event will be triggered if the condition specified by any of the configured travel distance settings were met.
  The 4 boundary settings allow you to shrink the size of the camera view for the purpose of generating direction of travel events. If you set one or more of the boundary settings, SAFR will not begin reporting or calculating direction of travel events until the object is within the specified boundaries. The boundary settings can be useful when people are hovering right at the edge of the camera's field of view (e.g. a doorman at the entrance to a hotel).
  - **Left travel distance**: Generates a direction of travel event when a face enters from the right edge of the camera view field and travels more than the specified percentage across the field towards the left edge.
  - **Right travel distance**: Generates a direction of travel event when a face enters from the left edge of the camera view field and travels more than the specified percentage across the field towards the right edge.
  - **Up/Away travel distance**: Generates a direction of travel event when a face enters from the bottom edge of the camera view field and travels more than the specified percentage across the field towards the top edge.
  - **Down/Towards travel distance**: Generates a direction of travel event when a face enters from the top edge of the camera view field and travels more than the specified percentage across the field towards the bottom edge.
  - **Left boundary**: Specifies the percentage of the left side of the camera view field to exclude from direction of travel event reporting.
  - **Right boundary**: Specifies the percentage of the right side of the camera view field to exclude from direction of travel event reporting.

- **Top boundary**: Specifies the percentage of the top side of the camera view field to exclude from direction of travel event reporting.
- **Bottom boundary**: Specifies the percentage of the bottom side of the camera view field to exclude from direction of travel event reporting.

60.3 Additional Options



Click + to manually add an IP camera.

Click **-** to delete the configuration for manually added cameras. This option is not available for auto-discovered cameras.

Exporting enables you to create a copy of a camera configuration. Exporting saves a camera configuration to an .acc file in JSON format.

Importing enables you to import a copy of a camera configuration. Use this feature to copy a camera configuration from one SAFR system to another.

61 Detection Preferences

The Detection Preferences tab allows you to configure facial, badge, and person detection characteristics.

Note that if both face detection and person detection are enabled, face objects can be associated with the appropriate person objects, thus enabling SAFR to continue tracking people even when they turn their faces away from the camera. See the Face Detection-Person Detection Tie-In topic for more information about this feature.

- **For Mode**: Specifies which video processing mode is affected by the current settings on this menu. See here for information about the different modes.

61.1 Enable Face Detector

The **Enable face detector** check box must be selected to enable face recognition.

- **Detection service**: Specifies which face detection service will be used.
  - *Standard*: The standard facial detection service that SAFR uses.
  - *High Sensitivity*: A high sensitivity facial detection service which has a lower latency and whose performance doesn't degrade when multiple faces are being analyzed simultaneously. The high sensitivity service consumes many more GPU resourcs than the standard service.
  - *Automatic*: This value will automatically select the high sensitivity service if sufficient GPU resources are available to run it. If there are insufficient GPU resources, then the standard service is used instead.
  - *SAFR*: macOS only. The face detection service that ships with SAFR. We strongly recommend that you use SAFR.
  - *Coreimage*: macOS only. A face detection service native to Apple. Its performance is comparable to SAFR's, but it's slightly less accurate.
  - *Vision*: macOS only. A face detection service native to Apple that's faster but much less accurate than either SAFR or Core Image.
- **Input Size**: This setting is only available if you selected *High Sensitivity* for the **Detection service** setting above. The **Input Size** setting allows you to manage the trade-off between accuracy vs. speed. There are 3 possible values:
  - *Normal*: This is the standard against which the other 2 possible values are measured.
  - *Small*: This value has decreased accuracy but increased speed.
  - *Large*: This value has increased accuracy but decreased speed.
- **Detection Sensitivity Threshold**: The sensitivity threshold when using the *High Sensitivity* facial detection service. The lower this value is, the more lenient the facial detection service will be when attempting to recognize a face, which can result in additional false positives. This setting is only available if you selected *High Sensitivity* for the **Detection service** setting above.
- **Reduce vertical input image size to**: Represents the vertical size in pixels to which the image scanned by the camera is scaled in order to perform face detection. Scaling down reduces CPU usage. 720-pixel resolution is usually sufficient for high-quality face detection without slowing down the CPU. Vertical size can be reduced to 640-pixels and even 480 to greatly reduce CPU usage for face detection, but this reduces the ability to detect and recognize smaller faces.(e.g. faces farther away from the camera)
- **Minimum searched face size**: Defines the minimum face size that can be detected. A searched size of 80, for example, can still manage to detect faces as small as 60x60, but with lower certainty. Lowering this number enables SAFR to detect much smaller faces but also greatly increases CPU usage. This setting is only available if you selected *Standard* for the **Detection service** setting above.
- **Minimum required face size**: Defines the minimum required size for a face to be detected. Any face smaller than the height or width is ignored.
  This is typically set when face detection needs to be limited only to large faces, which indicates a face being closer to the camera. It may not be desirable, for example, to cause a lot of detection events for faces that are too far from a camera to be considered necessary for attention by SAFR.

- **Consecutive confirmations required**: This setting adds consecutive confirmations to the SAFR facial recognition to create more reliable detections. Increase this setting for more reliable but slower facial detection.
- **Generate recognizer hint for detected faces**: Optimizes facial recognition. This setting should usually be enabled, it can be disabled to improve performance if detection is being performed at very low resolutions. Note that if this setting is disabled, recognition accuracy will be reduced.
- **Use custom detection thresholds**: This setting is only available if you selected *Standard* for the **Detection service** setting above. Allows you to customize the detection threshold. When this setting is checked, you can click on the *Configure* button to do the customization.
    - **Initial candidate selection threshold**: Initial face candidate threshold that is used during face detection.
    - **Middle candidate selection threshold**: Middle face candidate threshold that is used during face detection.
    - **Final candidate selection threshold**: Final face candidate threshold that is used during face detection.
- **Frame buffer size**: Sets the size of the frame buffer.
- **Maximum detectors per feed**: Windows only. Specifies the maximum number of detectors that can run concurrently on the same feed.

61.2 Enable RGB Liveness Detector

Enables RGB liveness detection. Only available on Windows machines containing an NVidia card that provides GPU. For a full description of how RGB liveness works, please see the RGB liveness detection topic.

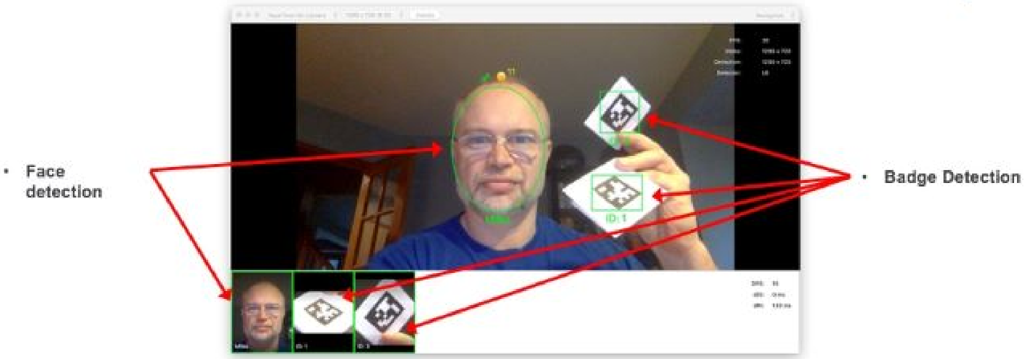
- **Minimum required face size**: The minimum required height and width of a face, in number of pixels, for the *Texture Model* to be used.
- **Minimum required face context size**: The minimum required extra context around faces for the *Context Model* to be used.
- **Minimum required center pose quality**: The minimum face center pose quality for RGB liveness detection to be used.
- **Minimum required face sharpness quality**: The minimum face sharpness quality for RGB liveness detection to be used.
- **Minimum required face contrast quality**: The minimum face contrast quality for RGB liveness detection to be used.
- **Minimum preliminary liveness threshold**: For multimodal *detection schemes*, this is the liveness threshold which the first evaluated model (the *Texture Model*) must exceed before SAFR will bother evaluating the second model. If this threshold is not met, SAFR immediately returns NOTLIVE_CONFIRMED for the subject.
- **Liveness detection threshold**: Specifies how difficult it will be for a subject to be verified as LIVENESS_CONFIRMED.
- **Fake detection threshold**: Specifies how difficult it will be for a subject to be verified as NOTLIVE_CONFIRMED.
- **Detection scheme**: Specifies which RGB liveness model(s) should be used.
    - *Texture Unimodal*: Only the Texture model will be used.
    - *Context Unimodal*: Only the Context model will be used.
    - *Strict Multimodal*: Both the Texture and Context models will be used. For a subject to pass the RGB liveness test, both of the results of the models must meet or exceed the *Liveness detection threshold* value. This is the default option.
    - *Normal Multimodal*: Both the Texture and Context models will be used. For a subject to pass the RGB liveness test, the average of the results of the two models must meet or exceed the *Liveness detection threshold* value.
    - *Tolerant Multimodal*: Both the Texture and Context models will be used. Subjects pass the RGB liveness test when the result of either model meets or exceeds the *Liveness detection threshold* value.
- **Evaluate liveness over N frames**: The number of frames over which liveness should be evaluated.

- **Evaluate fake over N frames**: The number of frames over which fakeness should be evaluated.
- **Minimum confirmations required**: The percentage of frames that must meet the liveness or fake threshold for the subject to be declared either LIVENESS_CONFIRMED or NOTLIVE_CONFIRMED.

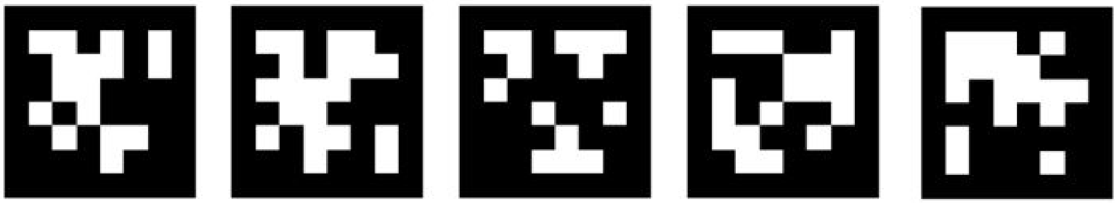
61.3 Enable Badge Detector

The **Enable badge detector** check box must be selected to enable badge detection. Badges are visual representations of users that are quicker and easier to detect than faces. Compared to faces, they are easier to detect and recognize when rotated and when used in low light conditions.

- **Reduce vertical input image size to**: Represents the vertical size a scanned image (badge) is scaled to in order to detect the badge. Scaling down the image reduces CPU usage.
  If your badges are very small, we do not recommend using this setting.
- **Minimum searched badge size**: Defines the minimum badge size that can be detected. Lowering this value enables SAFR to detect very small badges (down to 15x15 pixels) at the cost of increasing the CPU usage. Conversely, increasing this value reduces CPU usage but requires larger badges for successful detection.
- **Minimum required badge size**: Use this setting to require a minimum badge size in pixels. Any badge smaller than this value in either height or width is ignored. This setting is mainly used when badge detection is only expected to occur when the badges are close to the camera. (When cameras are close to the camera, the badge sizes are guaranteed to be larger.) If this value is set too small, SAFR could create many detection events for badges that are far from the camera and therefore not of interest.
- **Consecutive confirmations required**: This setting adds consecutive confirmations to the SAFR detection to create more reliable detections. Increase this setting for more reliable but slower badge detection. Decrease it for faster but slightly less reliable detection.
- **Detection service**: Specifies which badge detection method will be used. Depending on the capabilities of your cameras, lighting conditions, and other variables, certain options may work better with your environment than others.
  You can choose from the following options:
  - **apriltags**: Basic badge detection.
  - **rhinotagsLite**: The fastest badge detector, but it has a lower tolerance for motion blur. It requires cameras with a fast shutter speed.
  - **rhinotagsTeam**: Faster badge detector, but it has little resilience to motion blur.
  - **rhinotagsFlex**: Fast badge detector with moderate resilience to motion blur.
  - **rhinotagsFull**: Badge detector with robust handling under various conditions and a strong resilience to motion blur.
    rhinotagsFull is the recommended option.



• Sample Badges (can be printed for experimentation – 2"x2" intended size):



  A full set of badge images supported by SAFR is available at https://github.com/anqixu/apriltag/tree/master/tag36h11.
  - It is recommended these images be re-sized to at least 2" x 2" size using the nearest neighbor algorithm (to maintain sharp edges) before use with SAFR.
  - Although a single badge of displayed format can express only 587 different IDs, multiple badges can be combined to increase the number of expressible IDs into the billions. For example, using 6 badges providesover 827 billion expressible IDs.
- **Frame buffer size**: Sets the size of the frame buffer.
- **Maximum detectors per feed**: Windows only. Specifies the maximum number of detectors that can run concurrently on the same feed.

61.4 Enable Person Detector

Only available on Windows. The **Enable person detector** check box must be selected to enable person detection.

Note: Person detection is not available on the Lite Desktop Client.

- **Minimum required person to screen proportion**: Specifies the ratio of the person to the screen height. This can be between 0 - 1 and allows for decimal precision. For example, if you don't want the person to show up unless they are greater than 25% of the image height, specify a value of 0.25.
- **Consecutive confirmation required**: Number of consecutive detections that are required before reporting that the person (based on object id) was actually detected. This setting can be used to filter out false positives.
- **Detect persons every**: This can be used to avoid running person detection on every frame. Since person detection requires a lot of GPU processing if the hardware is not powerful enough this value can be changed so that you only attempt to detect people every Nth frame to save processing power to keep up with real-time detection.
- **Person detection threshold**: Detection threshold to use when matching persons. The higher the threshold the more strict the matching will be and the higher the confidence will be that the actual person matches.
- **Person separation threshold**: Threshold controlling the person separation when the persons are overlapping. This determines how much overlap is needed before no longer detecting the object with the weaker footprint.

- **Detection service**: This setting may have one of 3 values:
  - *Maximum accuracy*: Uses a larger model for the best accuracy, but the speed will be the slowest of the 3 options.
  - *Maximum speed*: Uses a smaller model for the fastest speed, but the accuracy will be the lowest of the 3 options.
  - *Balanced*: Uses a medium-sized model to have average precision and speed.
- **Input size**: Sets the person detector input size. This setting allows you to manage the trade-off between accuracy vs. speed. There are 3 possible values:
  - *Normal*: This is the standard against which the other 2 possible values are measured.
  - *Small*: This value has decreased accuracy but increased speed.
  - *Large*: This value has increased accuracy but decreased speed.
- **Frame buffer size**: Sets the size of the frame buffer.
- **Maximum detectors per feed**: Specifies the maximum number of detectors that can run concurrently on the same feed.

61.5 Enable Object Detector

macOS only. Internal use only.

61.6 Workload Limits

- **Max Simultaneous Detections**: macOS only. Specifies the maximum number of detectors across all video feeds that can be used on this Desktop Client. This setting defaults to the number of cores that your machine has. We strongly recommend that you do not change this setting; doing so could greatly decrease your system's performance.

62 Tracking Preferences

The Tracking Preferences tab allows you to configure settings for tracking detected people.

- **For Mode**: Specifies which video processing mode is affected by the current settings on this page. See here for information about the different modes.
- **Maximum change to continue tracking**
  - **Face position**: The maximum face position change to continue tracking, specified in percentage relative to the original face image size.
  - **Person position**: The maximum person position change to continue tracking, specified in percentage relative to the original person image size.
  - **Size**: The maximum size change to continue tracking, specified in percentage relative to the object size.
- **Stop tracking a face after it has lingered for**: Specifies how many additional frames SAFR will continue to keep a tracked face around after SAFR has failed to detect the face in the most recent frame.
- **Minimum recognitions to lock on to identity**: Minimum number of consecutive recognition attempts that must produce the same identity before SAFR locks onto the identity.
- **Minimum recognitions to learn identity**: Minimum number of consecutive face recognitions required to register a face into the Identity Database. This setting also affects the quality of the reference face signature recorded for an identity.

  Increasing this value increases the minimum quality of the face signature stored for a newly registered identity, but it decreases how quickly SAFR registers new identities.
- **Initial recognition attempts**: Number of initial recognition attempts to make on an unrecognized face as quickly as possible.
- **Failed recognition back-off interval**: After making the initial recognition attempts as quickly as possible, back up the amount specified by this setting for each subsequent recognition. This continues until the retry interval is reached.
- **Retry failed recognition after every**: The interval in which to run recognition requests if the face has not been recognized.
- **Reconfirm identity after every**: Specifies how often a face's identity is reconfirmed, in milliseconds. If you set this value to zero, SAFR continues to re-confirm a face's identity at its normal rate. Increasing this value increases the confirmation rate and improves SAFR tracking subjects in crowded settings at the cost of increased CPU and network usage.
- **Update identity every**: Updates the identity when the currently saved identity is older than the updated identity.
- **Minimum failed recognitions to stop tracking identity**: When a face is being tracked recognitions are continually confirming the identity. The identity is also being verified if it is transferred from a person object. In these cases, if the recognition or verification consecutively fails this number of times then the identity will be reset and no longer associated with the face because we are no longer sure it is the same identity.
- **High precision tracking**: Decreases event fragmentation and increases the stickiness of SAFR's tracking algorithm at the cost of computer processing power. This setting should be enabled if you are experiencing duplicate or missing Direction of Travel events. See Camera Preferences for information about the Direction of Travel feature.
- **Update identity with better image**: Updates the identity when the currently saved identity is of lower quality (in all aspects) than the new image.
- **Enable correlation of faces by size**: Enables face correlation of tracked faces, which compares detected faces looking for a change in area.

  In most situations this setting should be enabled, but disabling it may help performance when there is only a single face to track and head movements are very fast.
- **Enable motion prediction**: Enables face motion prediction, which predicts which direction the face is moving in order to maintain tracking.

  In most situations this setting should be enabled, but disabling it may improve performance when tracked faces are moving in highly irregular motion patterns.

- **Stop tracking on failed recognition**: Enabling this option causes identity tracking to stop when SAFR doubts a confirmation of face tracking failures. SAFR is then forced to obtain a new identity lock. Enabling it may produce more discontinuity in recognition events and provide additional protection against mistaken tracking.
  This setting rarely needs to be enabled.
- **Reconfirm identity in video after each Key Frame**: When a key frame is encountered in a video file all the faces that are being tracked are marked as unconfirmed so that their identities are reconfirmed to make sure they are the same person. This setting only applies to video files; it can't be used with live video. If a video file does not represent recorded live video then this can typically be set to true for better tracking during scene changes.

63 Recognition Preferences

The Recognition Preferences menu allows you to configure a variety of settings that affect how SAFR detects, tracks, and recognizes faces and identities.

63.1 Understand Facial Recognition

There are three key elements of SAFR facial recognition, each of which consists of a variety of settings that can be changed to help you fine-tune SAFR's performance:

- Detection: When a face is detected by a SAFR camera, as well as various settings that can affect how a face is recognized; minimum size, and resolution for a face to be detected, and more.
- Recognition: When a face is recognized it is compared against the SAFR database of recognized faces so it can be identified. Over time, SAFR collects and compares more images of an individual's face to help it build a catalog of variations to enable it to better recognize someone's face under different lighting conditions, angles, and with differing characteristics, such as a beard or different colored hair.
- Tracking: When a face is tracked, how long it is tracked, and other characteristics such as how tolerant the tracking is of motion and when to stop tracking.

63.2 Understand Occlusion Detection

SAFR can detect occluded faces. Occlusion constitutes any obstruction of the key facial features by, for example, a scarf, a hand, glasses, a mask, or hair draping over the face. Occlusion detection can be used to:

- Filter out highly occluded faces while learning them in the wild and preventing the storing of ambiguous face references in the Identity Database. For example, occlusion detection might be used when attempting to register players sitting at a table to prevent registering them with an occlusion feature, such as wineglass in front of their faces that may later create recognition inaccuracies.
- Update the occurrence event record with better face images without the occlusion to increase the value of the image stored with the event for presentation and investigation purposes.

63.2.1 Occlusion Detection Related to Events

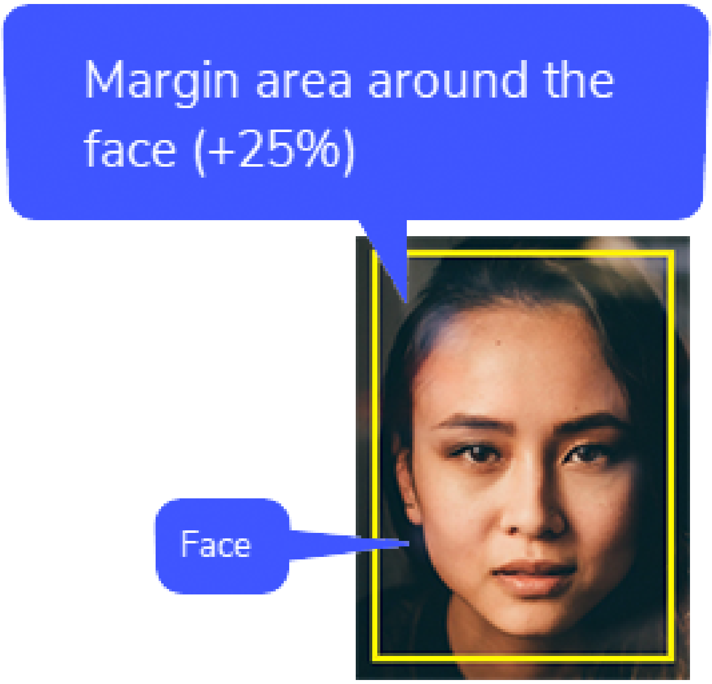
When the server returns an occlusion threshold in a recognition response, the most recent value is passed to the posted event under the following circumstances:

- When the event image is posted or updated.
- When the event `idClass` is set or updated.

The effect is the occlusion value in the event reflects occlusion of either the most recent event image update or an idClass change, whichever occurs last.

63.3 Recognition Preference Settings

- **For Mode**: Specifies which video processing mode is affected by the current settings on this page. See here for information about the different modes.
- **Minimum required face size**



  - **For recognition**: Defines the minimum required face size in pixels to attempt recognition. It includes a 25% margin around the face.
    The minimum face size (with +25% percent margin) for maximum recognition accuracy is 240 pixels. Faces smaller than 240 pixels may have slightly reduced recognition accuracy.
    This setting can be as low as 60 pixels if other image characteristics are favorable.
  - **For merging**: Defines the minimum required face size (+25% margin) to attempt merging a captured face with the existing reference image for an already registered identity.
    When in *Learn and Monitor* video processing mode, SAFR may merge reference images for an identity to improve its understanding of different face characteristics for that identity. Consider this as a catalog of variations for a face. See Select a Video Processing Mode for more information about modes.
  - **For learning/strangers**: Defines the minimum required face size (+25% margin) to enable SAFR to store a reference image for a new identity.
  - **For full accuracy**: Establishes the point below which face size is considered when determining image quality. Once the face size goes above this setting, face size is no longer used to determine the quality of the image. This setting should be changed only in very special circumstances.
- **Minimum required center pose quality**
  - **For recognition**: Defines the minimum required quality for a face posed directly in front of the camera (center posed) to attempt recognition. Center pose quality (CPQ) ranges from one to zero. A score of 1 is given to a face looking straight into the camera. Any deviation from this position diminishes center pose quality. Center pose quality of a face in full profile position is given a score of zero. Recognition from any pose is possible, but accuracy is reduced for faces that are in extreme profile positions.
  - **For merging**: Defines the minimum required face center pose quality to attempt merging with existing reference images for a recognized identity.
  - **For learning/strangers**: Defines the minimum required face center pose quality to enable SAFR to store a reference image for a new identity.
  - **For direct gaze detection**: If a face's center pose quality is above this value, the face is determined to be gazing directly at the camera, but if the center pose quality is below this value, the face is determined to be turned away. The longer the face gazes directly at the camera, the longer events' directGazeDuration is.

- **Use advanced settings for learning/strangers**: Enable if you want to modify the advanced learning setting below.
    - **Max Yaw**: Indicates minimum required yaw value to attempt registering somebody into the Person Directory. Yaw measures how much a face is turned to the left or right; a value of 0 indicates that the face is looking straight ahead.
    - **Max Pitch**: Indicates minimum required pitch value to attempt registering somebody into the Person Directory. Pitch measures how much a face is tilted up or down; a value of 0 indicates that the face is looking straight ahead and isn't tilted at all.
    - **Max Roll**: Indicates minimum required roll value to attempt registering somebody into the Person Directory. Roll measures how much a face is tilted to one side or the other; (i.e. the person's ear is moved closer to their shoulder) a value of 0 indicates that the face isn't tilted to either side.

- **Minimum required face sharpness quality**
  - **For recognition**: Indicates minimum required face sharpness quality to attempt recognition.
  - **For merging**: Indicates minimum required face sharpness quality to attempt recognition.
  - **For learning/strangers**: Indicates minimum required face sharpness quality to store as a reference for a new identity.
- **Minimum required face contrast quality**
  Contrast quality defines the difference between the color of a subject's face and the background.
  - **For recognition**: This setting indicates the minimum amount of contrast quality (lower or higher contrast) for SAFR to attempt a recognition.
  - **For merging**: Defines the minimum required face contrast quality to attempt merging a captured face with its existing references in the SAFR system.

  - **For learning/strangers**: Indicates the minimum required face contrast quality to store as a reference for a new identity.
- **Maximum allowed occlusion**
  - **For learning/strangers**: Indicates the maximum occlusion value allowed for a face to be registered to the Person Directory. When this setting is set to 1, no occlusion filtering is applied, and the default configuration is ignored.
  - **Learn occluded faces**: Select to have the system learn occluded faces. The check box is cleared by default.
- **Clipping tolerances**
  Clipping occurs when a face is only partially captured by a camera.
  - **For recognition**: This value defines the maximum amount of clipping tolerance (as a percentage of width or height) to attempt recognition. Faces not fully in the field of view are not recognized unless within this clipping tolerance threshold.
  - **For learning/strangers**: Indicates maximum allowed face clipping tolerance (as percent of width or percent height) to store as a reference for a new identity.
- **Identity recognition threshold**: Determines the strictness of the face recognition when declaring identity matches between a face and stored identity image. You can independently set the Identity Recognition Thresholds for the following:
  - **Camera**: Sets the threshold for images. (e.g. photos)
  - **Video**: Sets the threshold for video feeds and saved videos.
  - **Similar**: Sets the threshold for *Similar* comparisons when running Similar video processing mode.
  - **Masked face threshold offset**: Sets the threshold when detecting masks.
  - **Proximity threshold allowance**: A boost value that is added to the Identity Recognition Threshold.
    For detailed information about Identity Recognition Threshold and Proximity Threshold Allowance, see Identity Recognition Thresholds.
- **Maximum recognizers per feed**: Windows only. Specifies the maximum number of recognizers that can run concurrently on the same feed.

63.4 Detect

Select the check box to enable the detection of the following characteristics:

- **Identity**: The identity of the user in the SAFR system, such as their name.
- **Occlusion**: Obstructing the full view of the face by using, for example, a mask, glasses, or using a hand to block a part of the face.
  Note that if you disable this setting, then the *Mask* setting below will automatically be deselected as well.
- **Mask**: When enabled, SAFR will evaluate all occluded faces to see if they're covered by a mask. If they are, then SAFR will use the mask enhanced model to attempt to recognize the face behind the mask. If the occluded face isn't covered by a mask, then the normal occluded model will be used instead. Only standard blue or white surgical masks are currently supported; SAFR is unable to use the enhanced mask recognition model with masks of different colors or with masks that have customized patterns Enabling this setting will greatly increase SAFR's ability to recognize faces covered by masks, but it will needlessly slow down the system if there aren't any masks.
  Note that the *Occlusion* setting above will automatically be enabled if this setting is enabled.
  - **Mask Detection Mode**: Specifies the mode to be used for mask detection.
    - *Precise*: This mode produces the least number of false positives (i.e. detecting that a person is wearing a mask but there is no mask), but it suffers from the lowest true positive rate. (i.e. detecting masks that are actually there)
    - *Sensitive*: This mode produces the highest true positive rate, but it suffers from the highest number of false positives.
    - *Normal*: This mode produces a moderate amount of both false positives and true positives.
  - **Mask Detection Threshold**: Specifies the threshold at and above which mask detection will conclude that mask=true.
  - **Consecutive confirmations required**: Adds additional required consecutive confirmations for SAFR to recognize faces covered by masks. Increasing this setting creates more reliable but slower facial detection.
- **Gender**: Enables the detection of gender information.
- **Age**: Enables the detection of age information.
- **Sentiment**: Enables the detection of sentiment information.
- **RGB liveness action**: Only available on Windows machines containing an NVidia card that provides GPU. Enables the RGB liveness recognizer. When enabled, this recognizer creates events based on the RGB liveness feature when cameras view somebody enrolled in your SAFR Identity Database.
  - **Consecutive recognitions for live**: Windows only. Number of consecutive recognition attempts that must be successful for a LIVENESS_CONFIRMED event.
  - **Consecutive recognitions for fake**: Windows only. Number of consecutive recognition attempts that must be successful for a NOTLIVE_CONFIRMED event.
  - **Identity recognition threshold boost**: Windows only. The amount to temporarily boost identity recognition attempts during RGB liveness actions.
- **Smile action**: Enables the smile action recognizer.
  - **Pre-smile delay**: The amount of time, in milliseconds, that there should be no smile.
  - **Smile duration**: The amount of time, in milliseconds, that the smile should last.
  - **Identity recognition threshold boost**: The smile threshold to boost temporarily during the smile action.
  - **Transition thresholds**: Specifies what range of sentiment values are classified as happy, neutral, and unhappy.
  - **RGB liveness validation**: Only available on Windows machines containing an NVidia card that provides GPU. Enables RGB liveness validation during smile actions. Enabling this is required to use the *Secure Access with Smile and RGB Liveness* video processing mode.
    - **Consecutive recognitions for live**: Windows only. Number of consecutive recognition attempts that must be successful for a LIVENESS_CONFIRMED event.
    - **Consecutive recognitions for fake**: Windows only. Number of consecutive recognition attempts that must be successful for a NOTLIVE_CONFIRMED event.

- **Pose liveness action**: Enables the pose liveness action recognizer.
  - **Center pose quality**: Minimum center pose quality to use when detecting the initial center pose.
  - **Profile pose quality**: The maximum center pose quality to use when detecting the final profile pose.
  - **Max profile confidence at start**: Maximum profile pose confidence to allow during the initial center pose detection phase.
  - **Min profile confidence at end**: Minimum profile pose confidence to allow during the final profile pose detection phase.
  - **Min profile pose yaw**: The minimum profile pose yaw value that is required during the final profile pose detection phase.
  - **Center pose consecutive confirmations required**: Number of consecutive center pose confirmations required to enter the initial center pose detection phase
  - **Profile pose consecutive confirmations required**: Number of consecutive profile pose confirmations required to enter the initial center pose detection phase.
  - **Min profile similarity**: Minimum similarity score required when verifying the final profile pose.
  - **Min Detections Per Second**: Minimum number of frames per second required during the process.
  - **Min transition poses**: Minimum number of required center pose samples during the transition from center to profile pose.
  - **Max CPQ jump in continuous tracking**: Maximum change between samples while the pose is changing from center to profile.
  - **Max CPQ jump after tracking loss**: Maximum change between samples while the pose is changing from center to profile if lingering.
  - **Max profile pose roll**: The maximum roll threshold in either direction in which the face can rotate when determining whether the face is in profile pose.
- **3D Liveness**: Windows only. Enables 3D liveness. 3D liveness is a special feature of certain Intel RealSense camera models that allows them to distinguish flat images from 3 dimensional ones, thus allowing SAFR to tell the difference between a real face and a photo. This feature only works with Intel RealSense D415 and D435 cameras; if you don't have any cameras of those types connected to SAFR, then this feature will not work.
  - **Liveness Threshold**: Windows only. Specifies the 3D liveness threshold.

63.5 Workload Limits

- **Max Simultaneous Recognitions**: macOS only. Specifies the maximum number of recognizers across all video feeds that can be used on this Desktop Client. We strongly recommend that you do not change this setting; doing so could greatly decrease your system's performance.

64 Events Preferences

The Events Preferences tab allows you to enable and configure event reporting and event replies. Every reported event contains start and end times as well as information about where it occurred and who or what triggered it. The location of the event is indicated by the site and source labels, which can be defined for each camera. Events are stored in the Event Archive and can be easily retrieved in real time or on demand for search and analytics. Events are reported promptly as they occur and when they end. Events are updated as a better understanding of what they represent becomes available.

- **For Mode**: Specifies which video processing mode is affected by the current settings on this page. See here for information about the different modes.

64.1 Report Events

Enables event reporting. Event reporting enables SAFR to log and track events over time and gain additional insight into your SAFR system and usage patterns.

- **Include Unrecognizable Events from Camera**: Enable this option to report the appearance of unrecognized people captured by camera feeds. Unrecognized people are people that the SAFR system can't see well enough to compare it to its People Directory.
- **Include Unrecognizable Events from Video**: Enable this option to report the appearance of unidentified individuals captured in video files imported into SAFR. Unrecognized people are people that the SAFR system can't see well enough to compare it to its People Directory.
- **Include Stranger Events**: Enable this option to report when cameras see strangers. Strangers are people who the SAFR system can see well enough to compare him/her to individuals stored in its People Directory, and there isn't a match.
  - **Min Age**: The minimum age of strangers that will trigger stranger events. If a stranger younger than the specified minimum age is detected, no stranger event is generated.
  - **Max Age**: The maximum age of strangers that will trigger stranger events. If a stranger older than the specified maximum age is detected, no stranger event is generated.
  - **Only if occluded**: When enabled stranger events are only reported when the stranger is occluded. This can be useful if you want to catch people who are attempting to bypass your security system by intentionally occluding their faces.
- **Include Speculated Identity Events**: Enables reporting events for speculated people. A "Speculated Identity" is a face that isn't a 100% match with a face in the Person Directory, but is close.
- **Include Secondary Events**: When enabled, face events that are tied to a person event will be included. Enabling secondary events often produces undesired "noise", so it is turned off by default. See Face Detection-Person Detection Tie-In for information about secondary events.
- **Preserve Event Face Image**: Select this check box if you want the images that trigger an event to be saved with the event report.
  - **Max Image Size**: The maximum size of the event face images, in pixels.
  - **Image Margin**: Windows only. Specifies how much extra space around the face to include in the event face image.
  - **Image Format**: macOS only. Choose whether you want the event face image saved as a JPG or a PNG.
- **Preserve Event Scene Thumbnail Image**: Select this option if you want a thumbnail of the scene image in which the event occurred to be saved with the event report.
  - **Max Image Size**: The maximum size of the event scene thumbnail images, in pixels.
  - **Image Format**: macOS only. Choose whether you want the event scene thumbnail image saved as a JPG or a PNG.
- **Include Unrecognizable Event Images**: Specifies if event images should be stored for events triggered by unrecognizable people.
- **Include Stranger Event Images**: Specifies if event images should be stored for events triggered by people who aren't registered in your Identity Database.
- **Reporting delay**: The number of seconds an event report is delayed in order to properly assess the

nature of the event. For example, a person who may at first seem unknown may become known after a second observation.

- **Min Identified Event Duration**: The minimum duration required for an event representing a known person to be recorded as an event.
  This setting helps filter out noise or brief appearances that may not be worth reporting as a system event.
  If this setting and *Reporting Delay* have different settings, the greater number is used.
- **Min Unrecognizable Event Duration**: The minimum duration of an event representing an unrecognizeable person to be recorded as an event.
  If this setting and *Reporting Delay* have different settings, the greater number is used.
- **Min Stranger Event Duration**: The minimum duration of an event representing a stranger to be recorded as an event.
  If this setting and *Reporting Delay* have different settings, the greater number is used.
- **Min Masked Event Portion**: Indicates the portion of the qualifying mask detection samples during the event that are needed to indicate the presence of a mask for mask presence to be attributed to an event.
- **Update in-progress event attributes**: If this is enabled then any event properties that change will be updated at the specified interval. Many properties do change periodically, such as images or other averages that are continually computed.
  - **Update interval**: Specifies the interval time in which to update event properties that change.
  - **Include qualified images with updates**: Windows only. When enabled, SAFR will include qualified images with in-progress event attribute updates. These qualified images will be included even if they're lower quality than earlier images.
- **Update with higher quality image**: Update the thumbnail images with higher quality images during the course of the event if possible.

64.2 Listen for Event Replies

- **Display Reply Message**: Enables SAFR to display reply messages on the screen.
- **Display until end of event**: When enabled, the event reply is continuously displayed until the event ends. When this isn't enabled, the event reply is only shown for a couple seconds.
- **Generate auto-replies**: macOS only. Internal use only. Do not enable this setting.
- **Positive Reply**: Allows you to configure positive replies.
  - **Sound**: Specifies which sound, if any, is played when a positive event occurs.
  - **Voice**: Specifies which voice, if any, will be used.
  - **Overlay image**: Windows only. The image to be displayed while a positive reply is being displayed. If no image has been uploaded, then the live view of the video feed will be shown.
  - **Text Color**: Windows only. Specifies the text color of the reply.
  - **Background**: macOS only. The background image to be displayed while a positive reply is being displayed. If no image has been uploaded, then the live view of the video feed is used as the background.
- **Neutral Reply**: Allows you to configure neutral replies.
  - **Sound**: Specifies which sound, if any, is played when a neutral event occurs.
  - **Voice**: Specifies which voice, if any, will be used.
  - **Overlay image**: Windows only. The image to be displayed while a neutral reply is being displayed. If no image has been uploaded, then the live view of the video feed will be shown.
  - **Text Color**: Windows only. Specifies the text color of the reply.
  - **Background**: macOS only. The background image to be displayed while a neutral reply is being displayed. If no image has been uploaded, then the live view of the video feed is used as the background.
- **Negative Reply**: Allows you to configure negative replies.
  - **Sound**: Specifies which sound, if any, is played when a negative event occurs.
  - **Voice**: Specifies which voice, if any, will be used.
  - **Overlay image**: Windows only. The image to be displayed while a negative reply is being

    displayed. If no image has been uploaded, then the live view of the video feed will be shown.
  - **Text Color**: Windows only. Specifies the text color of the reply.
  - **Background**: macOS only. The background image to be displayed while a negative reply is being displayed. If no image has been uploaded, then the live view of the video feed is used as the background.
- **Reaction Delay**: Delays the event reporting to the server by this amount in seconds.

64.3 Remove Events

Enables removing of events according to the preferences selected within this section.

- **Remove events**: It's *Enabled* if either anonymous or non-anonymous events are configured to be removed. Click the **Change...** button to configure the event removal.
  - **Remove Anonymous Events after**: Determines how many days to wait before removing events triggered by people without a *name* attribute in the Person Directory. If this value is set to zero, then anonymous events won't be automatically removed.
  - **Remove Known Identity Events after**: Determines how many days to wait before removing non-anonymous events. If this value is set to zero, then non-anonymous events won't be automatically removed.

64.4 Remove Identities

Enables removing of identities according to the preferences selected within this section.

- **Remove identities**: It's *Enabled* if either anonymous or non-anonymous events are configured to be removed. Click the **Change...** button to configure the event removal.
  - **Target Directory**: Determines the directory whose identities are to be automatically removed.
  - **Remove Anonymous Identities after**: Determines how many days to wait before identities that don't have a *name* attribute. If this value is set to zero, then anonymous identities won't be automatically removed.
  - **Remove Identities of person type**: Select the *Person Type* of the identities you'd like removed. If you don't modify this field, then identities of all *Person Types* will be removed.
  - **after**: Determines how many days to wait before removing identities of the specified *Person Type*. If this value is set to zero, then identities with *Person Types* won't be automatically removed.

64.5 Event Biometric Indexing

Sets up biometric indexing on events. Biometric indexing is required to allow event searching by image on the Web Console's Events Page or the Desktop Client's Search by Image Window.

This setting is only visible to users with CONFIG_PRIVILEGE or SUPER_CONFIG_PRIVILEGE privelege levels. See Manage Users Preferences for information about user privilege levels.

- **Enable biometric event indexing**: Enables event biometric indexing.
  - **User directory name**: The name of the user directory whose event archive you want to biometrically index.
  - **Indexing speed**: The speed at which the event can be located when search by image is executed. Faster indexing speeds can lower system performance.
  - **Immediately index new events**: Specifies if events should be biometrically indexed as soon as they're created. Enabling this option can affect your system performance when events are created.
  - **Only index events occurred after specific date**: When checked, specifies the date after which events should be biometrically indexed. If this checkbox isn't checked, then all the events in the event archive are indexed.

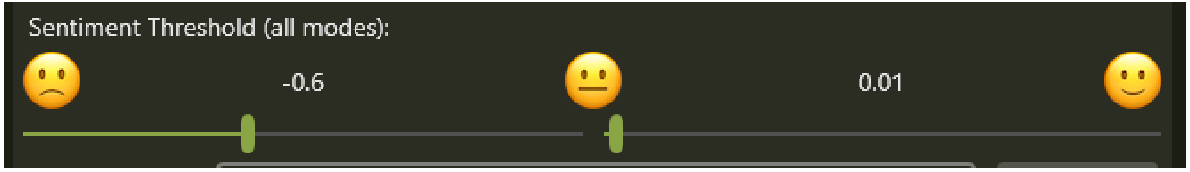
65 User Interface Preferences

The User Interface tab allows you to customize your Desktop Client's user interface.

- **For Mode**: Specifies which video processing mode is affected by the current settings on this page. See here for information about the different modes.
- **Default Window**: Windows only. Chooses the window that the Desktop Client will default to when it starts.
- **Operator Console is the primary application window**: Windows only. Sets the Operator Console as the primary application window. If this setting is not selected, then the Camera Window is the primary application window. Setting a window to be the primary application window means the Desktop Client will shut down when the window is closed.
- **Language**: Windows only. The language the Desktop Client uses.

65.1 Video

- **Accelerated Video Decoding**: Windows only. When enabled, the GPU will be used for video decoding.
- **Accelerated Video Rendering**: Windows only. When enabled, the GPU will be used for video rendering.
- **Loop playback**: Enable this to loop playback of a video file. This is primarily useful for looping video demos.
- **Name display time**: Use the slider to set the number of seconds a newly recognized person's name is flashed on the screen.
- **Minimum name refresh time**: This setting defines how long (in seconds) a recognized person must be out of camera view before their name is flashed again should they reappear in the camera view.
- **Highlight border thickness**: Use the slider to set the thickness (in pixels) of the frame displayed around faces and badges.
- **Custom highlight colors**: Windows only. Allows you to customize the colors for the video feed overlays.
- **Overlay text size**: Specifies the size of the text in the video feed overlay.
- **Name display message (#N = Full name, #F = First name, #U = Last name)**: Use this option to display a custom message to registered and recognized entrants.
  Use #N as a placeholder for the name of any recognized person. For example, Welcome, #N would display "Welcome, <recognized person's name>." The message is only displayed to registered persons.
- **Speak name display message**: When enabled, the *Name display message* will be spoken aloud.
- **Average Age and Gender**: During its normal operation, SAFR estimates the age and gender of people in every frame of a video stream independently. Thus, a person's displayed age can fluctuate by 10 years frame to frame. When this setting is enabled, ages and gender for a detected person are averaged over time, which creates a much smoother and more accurate experience. This setting has no effect on recognized people whose age or gender are specified within the Identity Database; SAFR will always display the stored values.
- **Sentiment Thresholds**: Specifies what range of sentiment values are classified as unhappy, neutral, and happy. The two Sentiment Threshold sliders are arranged as shown below.



Sentiment Threshold (all modes):

- **Background**: macOS only. Default background image that shows all the time. It can change based on events. If this is left blank, the video feed is shown.
- **Overlay image**: Windows only. Default background image that shows all the time. It can change based on events. If this is left blank, the video feed is shown.
- **Enable Registration**: Windows only. Enable this to allow unknown users to register their faces.

- **Start automatically after signing into Windows**: Causes the Desktop Client to automatically start when you sign in to Windows. This setting does nothing when kiosk mode is on.
  - **Configure Windows automatic sign-in**: Enables configuration of the credentials used when the Desktop Client automatically starts. If no credentials are entered, then SAFR won't be able to automatically start.
- **Kiosk mode**: Windows only. Enables kiosk mode for the Desktop Client. In Kiosk mode, the Windows taskbar won't be visible and the Desktop Client will be the only program that runs after signing into the account. Elevation is used to change the setting, and the current user SID is passed through such that a low-privilege kiosk user and a high-privilege admin user account may be configured on the machine. If the Desktop Client crashes for any reason while in kiosk mode, it will immediately and automatically restart.

  When you enable kiosk mode, a dialogue box will pop up with the following 2 settings you can enable:
  - **Also prevent this PC from going to sleep or activating the screen saver**
  - **Also prevent Windows Update automatic updates (manual update check required)**
- **Require sign-in every time SAFR Desktop starts**: Windows only. When enabled, users will be prompted to sign in every time the Desktop Client starts.

65.2 Similar

- **Overlay zoom**: macOS only. Sets the size of the comparison faces.
- **Maximum faces to select**: macOS only. Sets the maximum number of faces to compare to the target face.

65.3 Registration

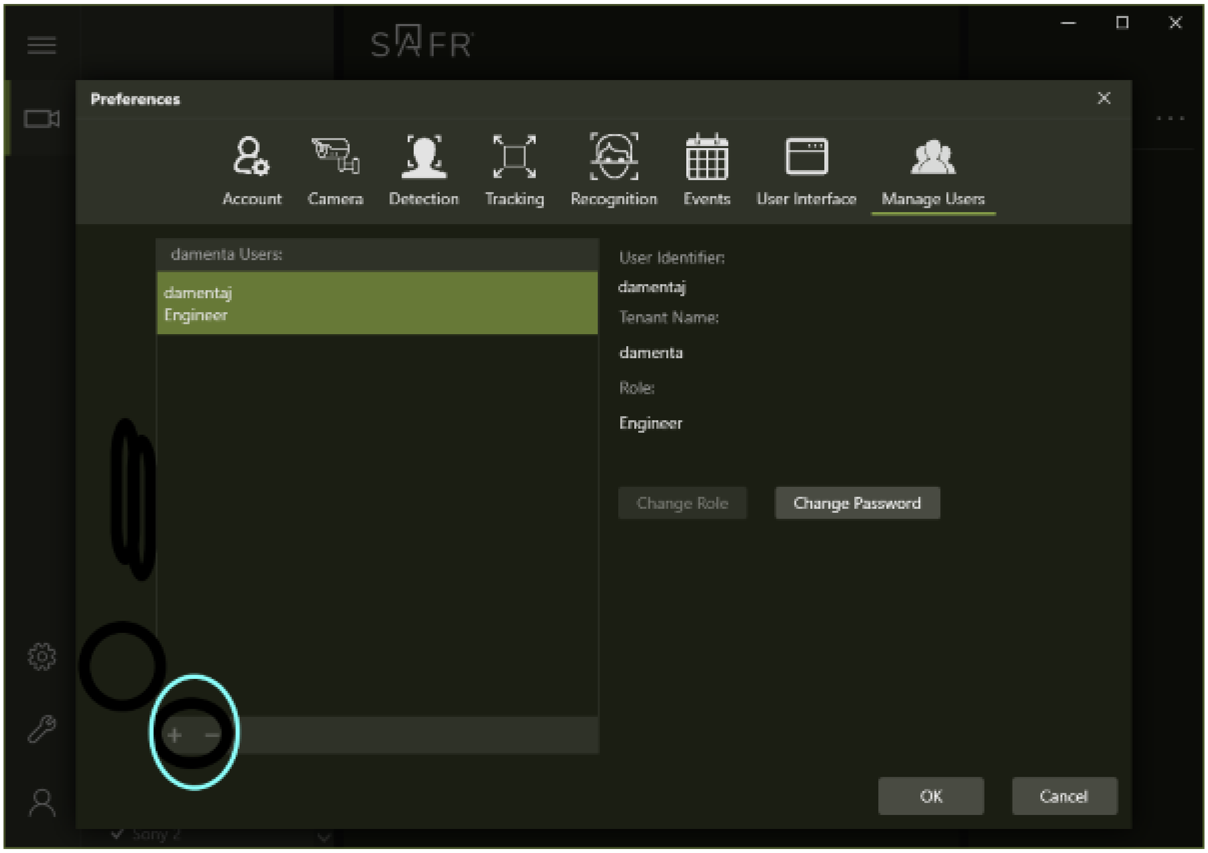
- **Enable Registration**: macOS only. Enable this to allow unknown users to register their faces.
  - **Min Age**: macOS only. Sets the minimum age that a person must be in order to register themselves. This setting is optional.
  - **Person Type**: macOS only. Automatically sets the *Person Type* of all people who register at this Desktop Client to the specified value.
  - **Home Location**: macOS only. Automatically sets the *Home Location* of all people who register at this Desktop Client to the specified value.

66 Manage Users Preferences

The Manage Users Preferences tab allows you to change your password, add or remove users, and edit users' access levels.

Most user roles can change their own password by clicking on the **Change Password** button. Administrators can also use this button to change the passwords of other users in their tenant, while super administrators can change the passwords of any other user. This can be useful when a password reset is needed.

Administrators can change other users' roles within their tenant by clicking on the **Change Role** button, while super administrators can change any user's role.



damenta Users:

damentaj
Engineer

User Identifier:
damentaj

Tenant Name:
damenta

Role:
Engineer

Administrators and super administrators can click the + button to add a new user. Similarly, administrators can click the **-** button to delete a user in their tenant, while super administrators can click **-** to delete any user.

66.1 Privilege Types

The following privilege types determine what access priveleges have been granted to users:

Privilege	Scope	Object	Description
READ_EVENT_PRIVILEGE	Tenant	Events	For monitoring events, allows access to CVEV GET /events and CVOS GET /stream and /object

Privilege	Scope	Object	Description
WRITE_EVENT_PRIVILEGE	Tenant	Events	For posting events and event data , allows CVEV POST /event and CVOS POST /stream and /object
READ_PRIVILEGE	Tenant	People	Allows matching of faces against known people, reading people's stored info, reading user info, etc.
WRITE_PRIVILEGE	Tenant	People	Allows insertion of new faces into an identity database and modification of personal information of recognized people within the user's tenant.
DELETE_PRIVILEGE	Tenant	People	Allows deletion of recognized people and faces within the user's tenant.
CONFIG_PRIVILEGE	Tenant	Config (Video, Settings)	Allows changes to any of the configuration values on the Video Feeds Window within the user's tenant.
ACCOUNT_PRIVILEGE	Tenant	Self	Allows changes to a user's own account properties, such as setting password, but doesn't allow changing other users' account properties.
ACCESS_PRIVILEGE	Tenant	Account	Allows making changes to users within the same tenant, including addition and deletion of users. APIs that require ACCOUNT_PRIVILEGE accept ACCESS_PRIVILEGE as well.
SUPER_READ_PRIVILEGE	Global	Events, People, and Config	Allows viewing recognized people and faces, reading VIRGO configurations, etc. across tenants.

Privilege	Scope	Object	Description
SUPER_WRITE_PRIVILEGE	Global	Events, People, and Config	Allows making changes to recognized people and faces properties, changes to virgo configurations, etc. across tenants.
SUPER_DELETE_PRIVILEGE	Global	People	Allows deletion of recognized people and faces across across tenants.
SUPER_CONFIG_PRIVILEGE	Global	Events and Config	Allows changes to any of the configuration values on the Video Feeds Window across tenants.
SUPER_ACCESS_PRIVILEGE	Global	Accounts	Allows admin of users across tenants.
LICENSE_RETRIEVAL_PRIVILEGE	Global		Allows the user to retrieve and edit SAFR license information. See On-Premise Licensing or Cloud Licensing for information about SAFR licenses.

66.2 User Roles

Every user account has a user role assigned to it, which determines the access priveleges granted to the user. The following user roles are defined:

Role	Cumulative Role	Privileges	Description
Analyst	Analyst	READ_EVENT_PRIVILEGE	Analysts can read event data stored in their tenants.
Monitor	Analyst + WRITE_EVENT_PRIVILEGE, READ_PRIVILEGE	READ_EVENT_PRIVILEGE, WRITE_EVENT_PRIVILEGE, READ_PRIVILEGE	Monitors can read identity and event data stored in their tenants. They can also write event data.
User Proxy	Monitor + WRITE_PRIVILEGE	READ_EVENT_PRIVILEGE, WRITE_EVENT_PRIVILEGE, READ_PRIVILEGE, WRITE_PRIVILEGE	User proxies can read and write all person and event data stored in their tenants.
Editor Proxy	User Proxy + DELETE_PRIVILEGE	READ_EVENT_PRIVILEGE, WRITE_EVENT_PRIVILEGE, READ_PRIVILEGE, WRITE_PRIVILEGE, DELETE_PRIVILEGE	Editor proxies can read and write all identity and event data stored in their tenants. They can also delete person data in their tenants.

Role	Cumulative Role	Privileges	Description
User	User Proxy + ACCOUNT_PRIVILEGE	READ_EVENT_PRIVILEGE, WRITE_EVENT_PRIVILEGE, READ_PRIVILEGE, WRITE_PRIVILEGE, ACCOUNT_PRIVILEGE	Users are identical to user proxies except they can also change their own passwords.
Editor	Editor Proxy + ACCOUNT_PRIVILEGE	READ_EVENT_PRIVILEGE, WRITE_EVENT_PRIVILEGE, READ_PRIVILEGE, WRITE_PRIVILEGE, DELETE_PRIVILEGE, ACCOUNT_PRIVILEGE	Editors are identical to editor proxies except they can also change their own passwords.
Engineer	Editor + CONFIG_PRIVILEGE	READ_EVENT_PRIVILEGE, WRITE_EVENT_PRIVILEGE, READ_PRIVILEGE, WRITE_PRIVILEGE, DELETE_PRIVILEGE, CONFIG_PRIVILEGE, ACCOUNT_PRIVILEGE	Engineers can manage all data stored in their tenants.
Administrator	Engineer + ACCESS_PRIVILEGE	READ_EVENT_PRIVILEGE, WRITE_EVENT_PRIVILEGE, READ_PRIVILEGE, WRITE_PRIVILEGE, DELETE_PRIVILEGE, CONFIG_PRIVILEGE, ACCESS_PRIVILEGE	Administrators can manage all users and data within their tenants.
Super Administrator	Administrator + SUPER_READ_PRIVILEGE, SUPER_WRITE_PRIVILEGE, SUPER_DELETE_PRIVILEGE, SUPER_CONFIG_PRIVILEGE, SUPER_ACCESS_PRIVILEGE, LICENSE_RETRIEVAL_PRIVILEGE	READ_EVENT_PRIVILEGE, WRITE_EVENT_PRIVILEGE, READ_PRIVILEGE, WRITE_PRIVILEGE, DELETE_PRIVILEGE, CONFIG_PRIVILEGE, ACCOUNT_PRIVILEGE, ACCESS_PRIVILEGE, SUPER_READ_PRIVILEGE, SUPER_WRITE_PRIVILEGE, SUPER_DELETE_PRIVILEGE, SUPER_CONFIG_PRIVILEGE, SUPER_ACCESS_PRIVILEGE, LICENSE_RETRIEVAL_PRIVILEGE	Super administrators can manage all users and data across all tenants. This role is only available to local deployments; in cloud deployments SAFR administrators adopt the role of super administrators, by design, since the SAFR engineering team is responsible for managing the SAFR servers for cloud deployments.
Founder	Founder	LICENSE_RETRIEVAL_PRIVILEGE, ACCOUNT_PRIVILEGE	Internal use only.

67 Cloud Licensing

SAFR systems require a license to operate.

SAFR licenses limit usage according to the following metrics:

- **Expiration date**: The date when the SAFR license expires. After this date, SAFR software discontinues operation.
- **Max Feeds per Hour**: Maximum number of video feeds that can be used at one time by the SAFR system. If you attempt to connect more video feeds than your license allows, the excess video feed connection attempts will all fail. Existing video feeds must be disconnected for a period of 1 hour before new video feeds are allowed to re-use the license.
  Note: If a single camera is providing video feeds to 2 different Desktop Client instances, that counts as 2 video feeds for licensing purposes.
- **Max Faces**: Maximum number of people that can be registered with the SAFR system's Person Directory. Attempting to add people above this limit results in an error.

License limit metrics for your SAFR license can be found on the Status page of the Web Console.

68 Windows Desktop Client Logging

There are two different types of logging available for the Windows Desktop Client.

- Trace Logging - Warnings, errors, and general application information.
- Windows Event Log - Crash information.

68.1 Trace Logging

You'll be using the tool **DebugView.exe** to view trace logging information. It can be downloaded here: https://docs.microsoft.com/en-us/sysinternals/downloads/debugview

To log trace information to a file, add the following to *Argus.exe.config*. (Default location: `C:\Program Files\RealNetworks\SAFR\Argus\Argus.exe.config`)

```
<configuration>
    <system.diagnostics>
        <trace autoflush="true" indentsize="4" >
            <listeners>
                <add name="SAFRTraceLogFileListener"
                     type="System.Diagnostics.TextWriterTraceListener"
                     initializeData="C:\Users\user\AppData\Local\Temp\SAFRDesktop.log"
                        >
                     <filter type="System.Diagnostics.EventTypeFilter"
                        initializeData="All" />
                </add>
                <remove name="Default" />
            </listeners>
        </trace>
    </system.diagnostics>
</configuration>
```

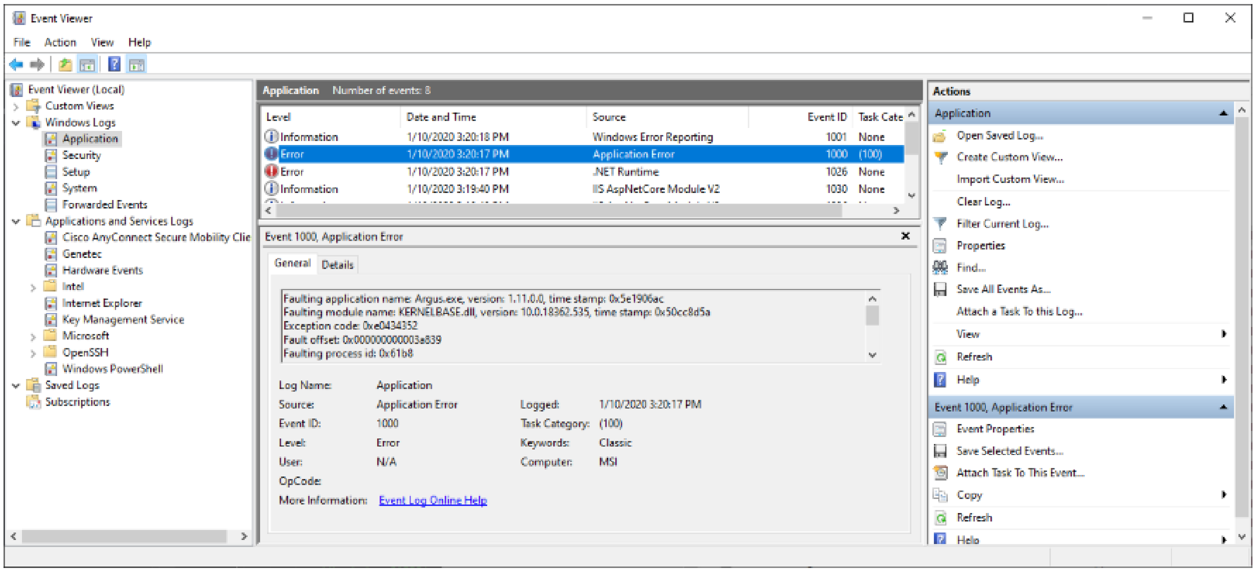
- SAFRDesktop.log will be the trace log. You can specify any location you prefer, so long as you have the necessary permissions to write data to the specified location.
- The filter line is optional. It's for restricting the log data to certain event types.
- To enable both DebugView and file logging, remove the <remove name="Default" /> line.
- You shouldn't leave trace logging on all the time, for the following reasons:
  - The log file will continuously grow until the disk is full.
  - Performance will be slowed when trace logging is turned on.
  - Trace logging is not thread-safe and may cause unexpected problems.

The *app.config* tracing configuration schema is located here: https://docs.microsoft.com/en-us/dotnet/framework/configure-apps/file-schema/trace-debug/

68.2 Windows Event Log

To view Windows Event Log information, do the following:

1. Open the Windows Event Viewer.
   1. Right click on the Windows icon in the bottom left corner of the screen.
   2. Select **Search*.
   3. Search for **Control Panel**.
   4. Click on **Administrative Tools**.
   5. Click on **Event Viewer**.
2. In the left hand pane, select **Windows Logs**->**Application** from the tree.
3. Look for "Application Error" in the **Source** column in the middle pane.



Application Number of events: 8

Level	Date and Time	Source	Event ID	Task Cate
Information	1/10/2020 3:20:18 PM	Windows Error Reporting	1001	None
Error	1/10/2020 3:20:17 PM	Application Error	1000	(100)
Error	1/10/2020 3:20:17 PM	.NET Runtime	1026	None
Information	1/10/2020 3:19:40 PM	IIS AspNetCore Module V2	1030	None

Event 1000, Application Error

```
Faulting application name: Argus.exe, version: 1.11.0.0, time stamp: 0x5e1906ac
Faulting module name: KERNELBASE.dll, version: 10.0.18362.535, time stamp: 0x50cc8d5a
Exception code: 0xe0434352
Fault offset: 0x000000000003a839
Faulting process id: 0x61b8
```

Log Name: Application

Source: Application Error   Logged: 1/10/2020 3:20:17 PM

Event ID: 1000   Task Category: (100)

Level: Error   Keywords: Classic

User: N/A   Computer: MSI

OpCode:

More Information: Event Log Online Help

69 SAFR Desktop Command Line Install Options

Note: These command line install options are only available on Windows.

Silent installation of SAFR Desktop on Windows can be achieved by invoking the SAFR Desktop installer via the command line and using the `/S` switch.

Example:

```
SAFRDesktop_win_1_8_442_10_18_19.exe /S
```

The Windows SAFR Desktop installer provides several options for configuring the component selection during install. Each component can be disabled or enabled by using the following syntax:

- `SAFRDesktop_win_1_8_442_10_18_19.exe /S /COMPONENT=YES`
- `SAFRDesktop_win_1_8_442_10_18_19.exe /S /COMPONENT=NO`

Examples:

```
SAFRDesktop_win_1_8_442_10_18_19.exe /S /VIRGO=YES /Actions=NO

SAFRDesktop_win_1_8_442_10_18_19.exe /Actions=YES /Digifort=YES
```

69.1 Command Line Install Options

Feature Type	Component	Flag	Default	Notes
Silent Install	Silent Install	/S	Disabled	
Component	SAFR Actions and ARES	/Actions	Enabled	
Component	Desktop Client	/Application	Enabled	
Component	VIRGO	/VIRGO	Enabled	
SAFR Client Component	GPU Accelerated Detection	/CUDA	Enabled	OK to install even if NVIDIA drivers aren't installed.
VMS Integration Plugin	Avigilon Plugin	/Avigilon	Disabled	Only one VMS Plugin allowed. The first specified plugin will be used.
VMS Integration Plugin	Digifort Plugin	/Digifort	Disabled	Only one VMS Plugin allowed. The first specified plugin will be used.
VMS Integration Plugin	Genetec Plugin	/Genetec	Disabled	Only one VMS Plugin allowed. The first specified plugin will be used.
VMS Integration Plugin	GenetecFR Plugin	/GenetecFR	Disabled	Only one VMS Plugin allowed. The first specified plugin will be used.

Feature Type	Component	Flag	Default	Notes
VMS Integration Plugin	Geutebrueck Plugin	/Geutebrueck	Disabled	Only one VMS Plugin allowed. The first specified plugin will be used.
VMS Integration Plugin	Milestone Plugin	/Milestone	Disabled	Only one VMS Plugin allowed. The first specified plugin will be used.
VMS Integration Plugin	Video Insight Plugin	/VideoInsight	Disabled	Only one VMS Plugin allowed. The first specified plugin will be used.
Camera Extension	Ximea Camera Extension	/Ximea	Disabled	
Installation Location	Installation Location	/D	`C:\Program Files\RealNetworks\SAFR`	Must be the last command line argument. Do not use quotes.

70 Desktop Client Command Line Interface

The following Desktop Client command line commands can help Windows users to programmatically manage large numbers of Desktop Clients. To run any of these commands, you should first:

1. Right click on the Windows icon in the bottom left corner of the screen, then select either **Command Prompt (Admin)** or **Windows PowerShell (Admin)**.
2. Within the command prompt, navigate to `C:\Program Files\RealNetworks\SAFR\Argus`.
3. Again within the command prompt, run whichever command line command you want, following the syntax rules specified below.

70.1 settings Command

Add or update settings. The Desktop Client will run with no UI, update the settings, and then exit.

Syntax:

```
argus.exe settings <settings JSON>
```

Settings names can be found by looking in the user.config file located at `C:\Users\<Your Username>\AppData\Local\RealNetw`

Example:

```
argus.exe settings "{'Environment': 'CUSTOM', 'CoViServer':
    'http://localhost:18080/covi-ws', 'EventServer':
    'http://localhost:18092/', 'ObjectServer': 'http://localhost:18086',
    'VRGAServer': 'http://localhost:18084'}"
```

70.2 quit Command

Closes all instances of SAFR without saving the current configuration settings.

Syntax:

```
argus.exe quit
```

70.3 quitandsave Command

Saves current configuration settings and closes all instances of SAFR.

Syntax:

```
argus.exe quitandsave
```

70.4 copysettings Command

This command either updates camera settings for a single camera if a path to an .acc file is provided, or it replaces the entire settings file if a path to a .config file is provided.

There is no UI displayed, and the process exits after completion.

Syntax:

```
argus.exe copysettings <importFilePath>
```

where <importFilePath> is either: <filename>.acc or <filename>.config

70.5 SecondaryJoin Command

Runs `python.exe <scriptargs>`. This is used to launch safr-worker.py with elevated priveleges for secondary joins while also keeping the console window hidden.

Syntax:

```
argus.exe SecondaryJoin <scriptargs>
```

70.6 import Command

Imports faces from a single image (JPEG or PNG) into the Identity Database.

The people import UI will pop upso that you can complete the import.

This command is intended to be called exclusively by Windows File Explorer in the SAFRImport verb registration, which enables single-selecting image files and starting import as if the user selected the file from within the Desktop Client.

Syntax:

```
argus.exe import <filePath>
```

71 Connect a Face Recognition Panel

A face recognition panel is a mobile device running the Mobile Client that has been placed in **Secure Access** or **Secure Access With Smile** video processing mode. It's intended to be used at the door (usually placed behind safety glass on the inner side of a door) as part of the SAFR Secure Access setup. A face recognition panel generates events which are sent to the SAFR Server, and are then picked up by ARES, which in turn triggers the door unlock action.

71.1 Download and Install the Mobile Client

To install the Mobile Client, go to the SAFR Download Portal, download the Mobile Client specfic to your mobile device's OS, and then run the installer.

You can also download the client from the *Apple App Store* if you're using an iOS device by doing the following:

- Go to the *Apple App Store* and search for *SAFR Recognition*.
- Using your browser, navigate to *itunes.apple.com/app/id1376830890*.

71.2 Connect the Mobile Client to a SAFR Server

To connect your Mobile Client to a SAFR Server, do the following:

1. Make sure your mobile device is connected to the internet and that it can make a network connection either to the SAFR Cloud (for cloud deployments) or to your SAFR Server (for on-premise deployments).
2. Start the Mobile Client.
3. Sign in using your credentials.
    - Enter your user ID and password in the sign-in dialog that appears on the screen. **Note:** Make sure the front-facing camera of your mobile device has a view of your face when signing in. Your face is not recorded, but it must be detected for sign-in to be offered.
    - Tap the **Sign In** button that appears at the top of the screen.
    - Enter your credentials, select the agreement to terms of service check box, and tap **Sign In**.

If successful, the **Sign In** button disappears and a purple frame is displayed around your face with **Tap to Register** displayed underneath.

Note: For on-premise deployments, iOS devices require that your primary SAFR Server have an SSL certificate. See SSL Certificate Installation for instructions on how to do this.

71.3 Configure the Mobile Client as a Face Recognition Panel

To configure the Mobile Client as a face recognition panel, do the following:

1. Start the Mobile Client.
2. Open the settings menu by tapping the gear icon in bottom left corner of the screen.
3. Tap the video processing mode selector at the top center of the screen, and select either **Secure Access** or **Secure Access With Smile**.
    - **Secure Access** mode generates an event when a person is recognized.
    - **Secure Access with Smile** mode generates an event when a recognized person is observed changing expression from non-smiling to smiling.
      Note: When in **Secure Access** or **Secure Access With Smile** mode, video is turned off by default. If you want to show the video, you can override the default behavior from the settings menu (gear icon) in the **User Interface** tab.
4. Complete the **User Site** and **User Source** fields.
    - The **User Site** labels the site (e.g. My-Office) at which you are deploying SAFR.
    - The **User Source** labels the entrance location (e.g. Front-Door) at which the mobile device is placed.

Note: *Site* and *Source* labels are associated with every registration as well as with every other event and are crucial in making the source of registrations as well as other events traceable.

5. (Optional) Configure the mobile device into Locked Mode to lock in the Mobile Client as the exclusive application for the device.

   Note: Locking your mobile device locks the device to the Mobile Client and prevents any disruption in the face registration panel operation due to operating system updates or unauthorized user interference. It isn't necessary to lock your mobile device if you merely want to try out the Mobile Client as a face registration panel. However, you should lock the device before deploying the registration kiosk in a production environment.

72 Connect a Registration Kiosk

A SAFR registration kiosk is a mobile device running a Mobile Client that has been placed in Registration Kiosk video processing mode. It's intended to be used to take pictures of users and enable them to register their faces and identity information with the SAFR system.

72.1 Download and Install the Mobile Client

To install the Mobile Client, go to the SAFR Download Portal, download the Mobile Client specfic to your mobile device's OS, and then run the installer.

You can also download the client from the *Apple App Store* if you're using an iOS device by doing the following:

- Go to the *Apple App Store* and search for *SAFR Recognition*.
- Using your browser, navigate to *itunes.apple.com/app/id1376830890*.

72.2 Connect the Mobile Client to a SAFR Server

To connect your Mobile Client to a SAFR Server, do the following:

1. Make sure your mobile device is connected to the internet and that it can make a network connection either to the SAFR Cloud (for cloud deployments) or to your SAFR Server (for on-premise deployments).
2. Start the Mobile Client.
3. Sign in using your credentials.
    - Enter your user ID and password in the sign-in dialog that appears on the screen. **Note:** Make sure the front-facing camera of your mobile device has a view of your face when signing in. Your face is not recorded, but it must be detected for sign-in to be offered.
    - Tap the **Sign In** button that appears at the top of the screen.
    - Enter your credentials, select the agreement to terms of service check box, and tap **Sign In**.

If successful, the **Sign In** button disappears and a purple frame is displayed around your face with **Tap to Register** displayed underneath.

Note: For on-premise deployments, iOS devices require that your primary SAFR Server have an SSL certificate. See SSL Certificate Installation for instructions on how to do this.

72.3 Configure the Mobile Client as a Registration Kiosk

To configure the Mobile Client as a registration kiosk, do the following:

1. Start the Mobile Client.
2. Open the settings menu by tapping the gear icon in the bottom left corner of the screen (on iOS devices) or the hamburger icon in the upper left corner of the screen (on Android devices).
3. Tap the mode selector at the top center of the screen, and select **Registration Kiosk**.
4. Complete the **User Site** and **User Source** fields.
    - The **User Site** identifies the site (e.g. My-Office) at which you are deploying the SAFR System.
    - The **User Source** identifies the registration kiosk (e.g. Registration-Kiosk) as the source of registrations.
      Note: *Site* and *Source* labels are associated with every registration as well as with every other event and are crucial in making the source of registrations as well as other events traceable.
5. (Optional) Configure the mobile device into Locked Mode to lock in the Mobile Client as the exclusive application for the device.
   Note: Locking your mobile device locks the device to the Mobile Client and prevents any disruption in the registration kiosk operation due to operating system updates or unauthorized user interference. It isn't necessary to lock your mobile device if you merely want to try out the Mobile Client as a registration kiosk. However, you should lock the device before deploying the registration kiosk in a production environment.

72.4 Register and Organize SAFR Users in your System

Although users can self-register their faces at a registration kiosk, they are not automatically registered and approved in the system or granted access privileges. SAFR administrators can classify and control access to resources by using the Person Directory to assign various categories and tags to registrants. For more information on searching, viewing, and organizing registrants, see Manage People in the Person Directory.

For example, you can require every registrant to be assigned a **Person Type** property and base access to certain resources on that property. Think of **Person Type** as a category for your users, such as Staff, Maintenance, Administrator, or anything else you might like to define. The **Home Location** and **Person Type** properties associated with registrants can be adapted to different needs for different organizational purposes. You can also use the **Home Location** and **Person Type** properties to filter information. For example, in a school setting you might use **Home Location** to denote the grade of a student, and **Person Type** might be defined as *student*.

Click **Add Home Location** or **Add Person Type** to add new options or choose from the existing ones. Existing options appear as options in the menu.

73 Customize a Registration Kiosk

Each registration kiosk can be customized to prompt for additional required or optional information from the registrant. You can also customize:

- The registration prompt.
- Registration completion message.
- The default **Person Type** or **Home Location** for the registrant.
- A minimum age requirement for registrants, estimated based on the registrant's face.

73.1 Customize the Registration Prompt

To customize the registration prompt, do the following:

1. In the Mobile Client, tap the gear icon (settings) > **User Interface**.
2. Enter a new text next to **Prompt**.

73.2 Assign Default Person Type or Home Location Values

It may be desirable to assign a default **Person Type** or **Home Location** value to all registrants who complete registration at a particular registration kiosk. For example, if a registration kiosk is located in the admissions office, anyone registered there could be assigned the **Person Type** of *Student* or perhaps *Employee*. Anyone registered at the registration kiosk placed at a specific location could be given a default **Home location** corresponding to the town in which the registration kiosk is located. This can save administrative time. Both **Person Type** and **Home Location** can be changed by the administrator after the registration when needed.

To configure the default **Person Type** or **Home Location**:

1. In the Mobile Client, tap the gear icon (settings) > **User Interface**.
2. Enter a value for **Person Type** if desired. By default, **Person Type** is not assigned.
3. Enter a different value for **Home Location**. By default, **Home Location** is set the same as the **Site** label specified in the **Account** settings.

The **Home Location** field associated with every person registered can be used for various purposes. For example, in a school settings, it could be used by the administrator to enter the building name in which a student's home classroom is located. **Home Location** and **Person Type** fields offer filtering based on labels used for these fields and can become important organizational tools. They are named generically to allow labels to be created on the fly by simply entering them. You should decide how to use these labels and then use them consistently to get the most value from them.

Note: As a best practice, neither of these fields should have more than two dozen labels for ease of use.

73.3 Restricting Registration to a Minimum Age

It may be desirable to prevent registration of people below a certain age. The Mobile Client can be configured to asses a person's age and not offer registration to people below a specified minimum age.

To configure the minimum registration age:

1. In the Mobile Client, tap the gear icon (settings) > **User Interface**.

2. Enter the desired value for **Min Age**.

3. (Optional) Change **Show Attributes** to *Off*.

   Tip: Switching **Show Attributes** to *Off* prevents displaying the assessed age to the registrant. Because some people may be sensitive to this feedback, it is recommended that age not be shown.

4. On the **Recognition** tab, change **Detect Age** to *On*. With age detection set to *On*, the restriction is now active.

73.4 Customize the Registration Form

To customize the message your kiosk displays to registrants, do the following:

1. In the Mobile Client, tap the gear icon (settings) > **User Interface > Form: Customize**.
2. For any fields you want to add to your form, change the *Hidden* indicator to either *Required* or *Optional*. Any field that is marked as *Required* needs to be filled out by the registrant before registration is allowed to be complete.
    - The *Name*, *Company*, *Mobile*, and *Email* fields have fixed meanings. While you can customize prompt names for these fields, information entered for these fields is registered under the prescribed meaning. If you do not want to have this information gathered during registration, keep these fields hidden. Do not re-label them to a different meaning.
      Note: *Name* cannot be hidden and must be entered by the registrant.
    - If you need to gather information in addition to these prescribed fields, use the generic fields labeled by default as *Field*. These form entries have no prescribed meaning. Any information provided through these fields appears as tags in the registered person's record. If you want to give the entered information a tag name, complete the *Tag* field for each entry. If *Tag* is completed, information the registrant fills out for this field is prefixed with "Tag=" when appearing in person's record (e.g. Car Make=Ford). If the tag is not filled out, the information provided by the registrant appears on its own in the list of tags associated with the registered person.
3. Enter the names for the fields and add any information placeholder text. (e.g."Type Your Name Here")
4. Change the labels for the actions buttons if desired.
5. Enter the completion message displayed once the registration process is successfully completed.

74 Configure a Mobile Device into Locked Mode

Single App Mode for iOS, or Lock Task Kiosk Mode for Android, allows you to lock a mobile device into running only one application even if it is rebooted. This mode allows the device to be fully locked from any unauthorized access, and the device will remain locked until Single App or Lock Task Mode is explicitly disabled.

You can control how users interact with devices using Single App or Lock Task Mode by enabling or disabling any of the following features:

- Screen auto-lock
- Touch input
- Screen rotation
- Volume control
- Sleep/wake button
- Side switch

74.1 Requirements

You'll need a Macintosh computer running 10.14 Mojave or later to set up Single App Mode.

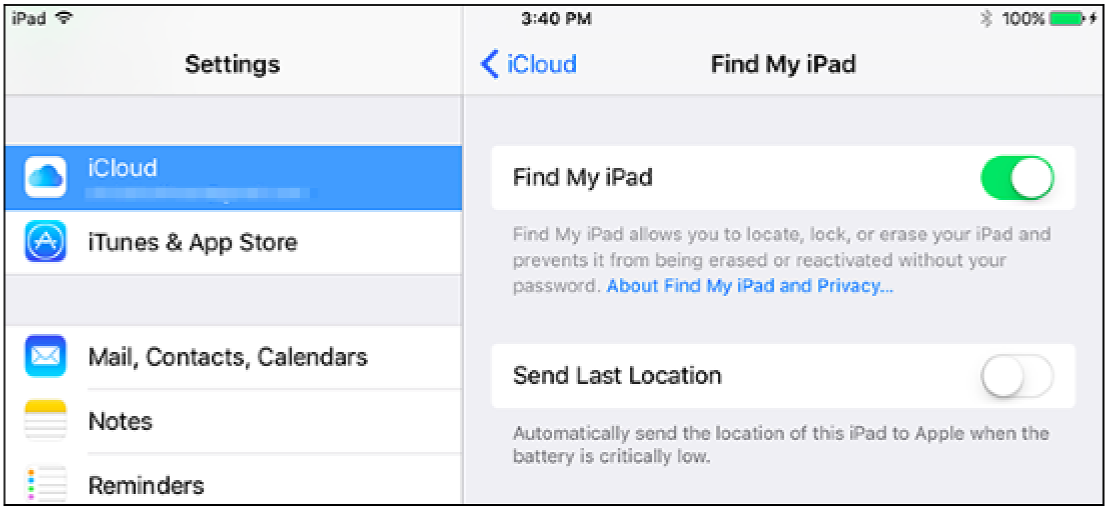
You'll need 2 Android devices to set up the most secure mode for Android devices: Lock Task Mode. If you only have a single Android device, then you can only set up the less secure Screen Pinning Mode.

74.2 Put an iOS Device into Supervised Mode

Warning: Putting an iOS device in Supervised Mode wipes all the information on the device and resets it.

To put an iOS device into Single App Mode, the device must first be put into Supervised Mode. To do this, do the following:

1. Go to **Settings > (User) > iCloud > Find My iPad/iPhone**. Disable the **Find My iPad/iPhone** switch by entering the password.



Find My iPad

Find My iPad allows you to locate, lock, or erase your iPad and prevents it from being erased or reactivated without your password. About Find My iPad and Privacy...

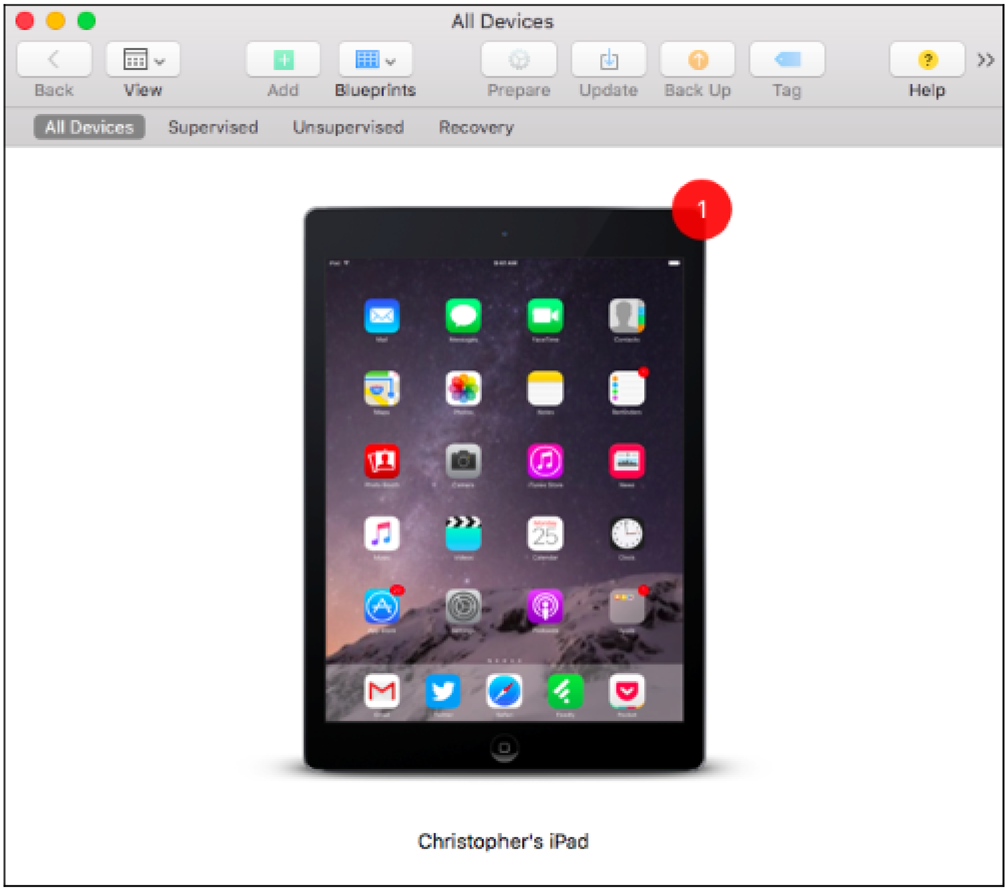
Send Last Location

Automatically send the location of this iPad to Apple when the battery is critically low.

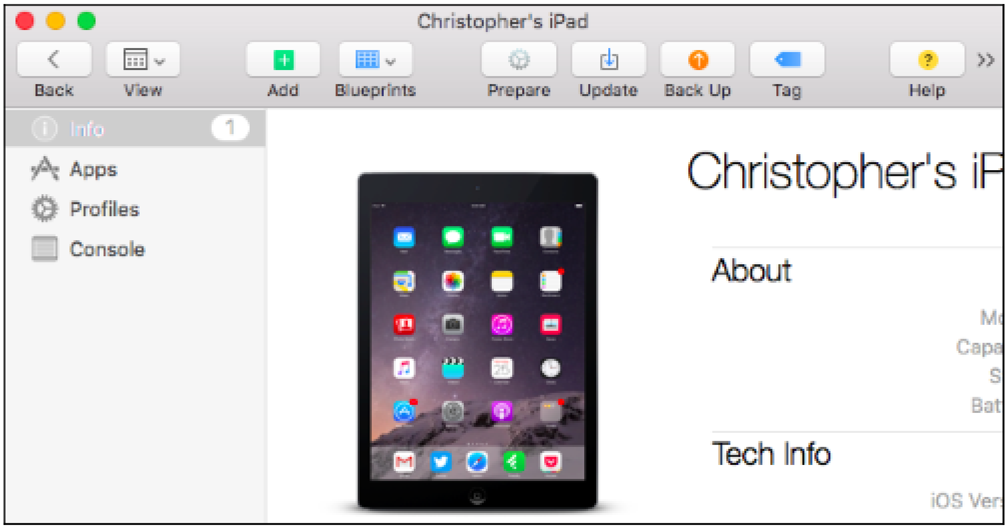
2. On your Mac, launch the **App Store** application and search for **Apple Configurator 2**. Download and install this application on the computer.

3. Plug in your iOS device to your Mac.

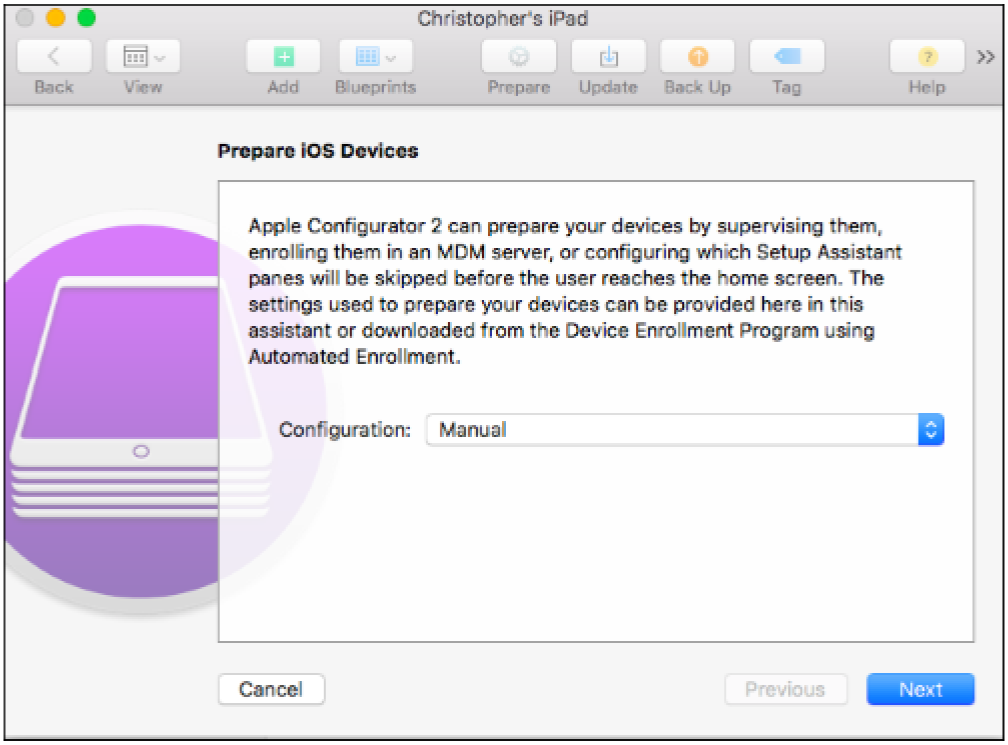
4. Launch **Apple Configurator 2**. You should see something that looks like the image below.



5. Double-click the device.
6. On the **Details** screen about the device, click the **Prepare** button.



7. From the **Configuration** menu, select **Manual**.

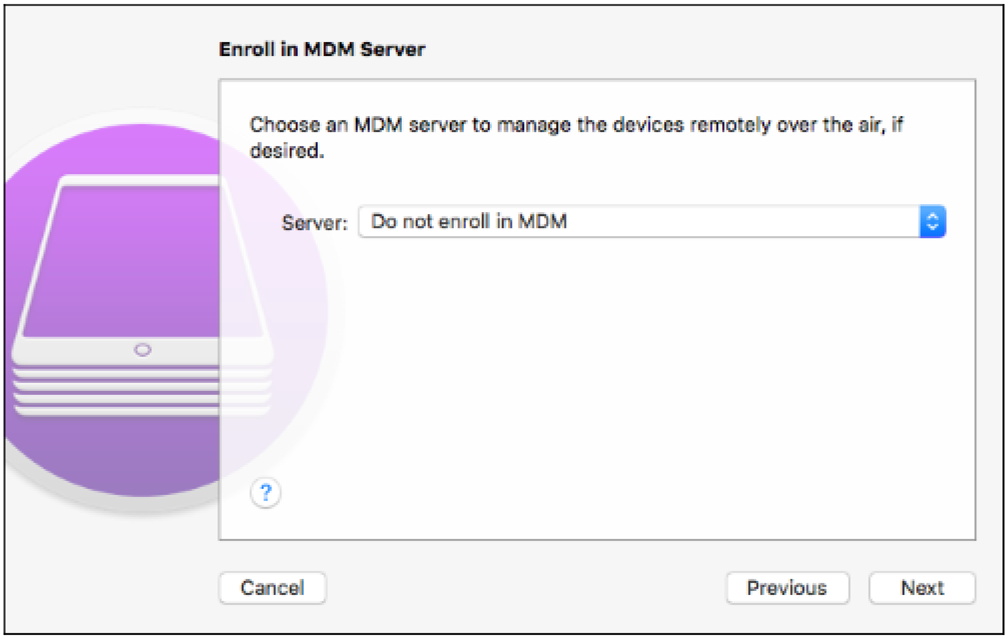


Prepare iOS Devices

Apple Configurator 2 can prepare your devices by supervising them, enrolling them in an MDM server, or configuring which Setup Assistant panes will be skipped before the user reaches the home screen. The settings used to prepare your devices can be provided here in this assistant or downloaded from the Device Enrollment Program using Automated Enrollment.

Configuration: Manual

8. From the **Server** menu, select **Do Not Enroll in MDM** unless you have an MDM server you want to use and enroll your device to.

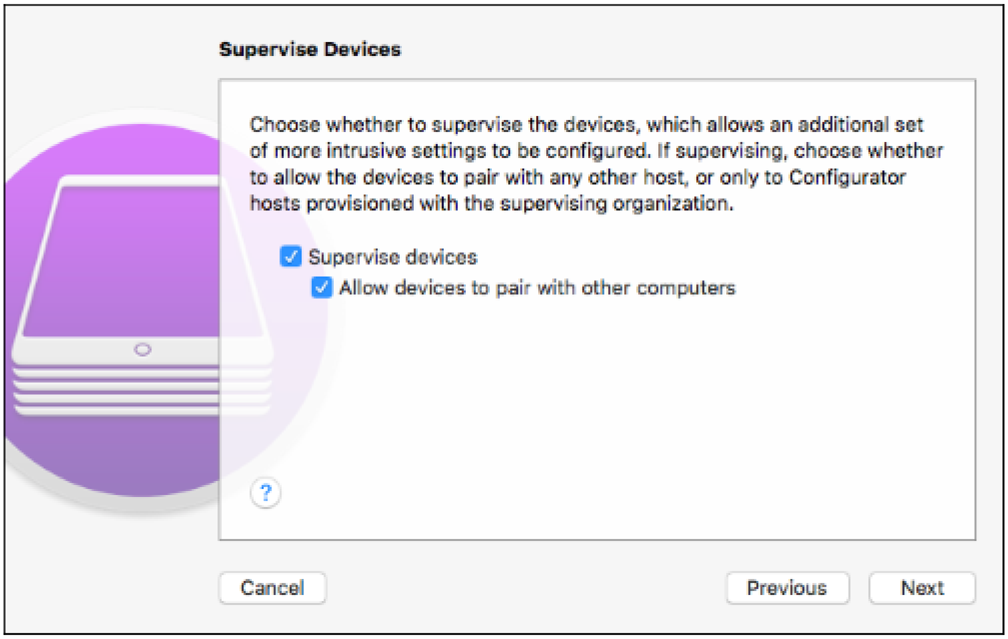


Enroll in MDM Server

Choose an MDM server to manage the devices remotely over the air, if desired.

Server: Do not enroll in MDM

9. If you selected **Do Not Enroll**, you must now plug the mobile device into your Mac to configure it.
10. Click the **Supervise Devices** check box. If you want the device to be configured on multiple computers leave the default **Allow Device to Pair with Other Computers** selected.



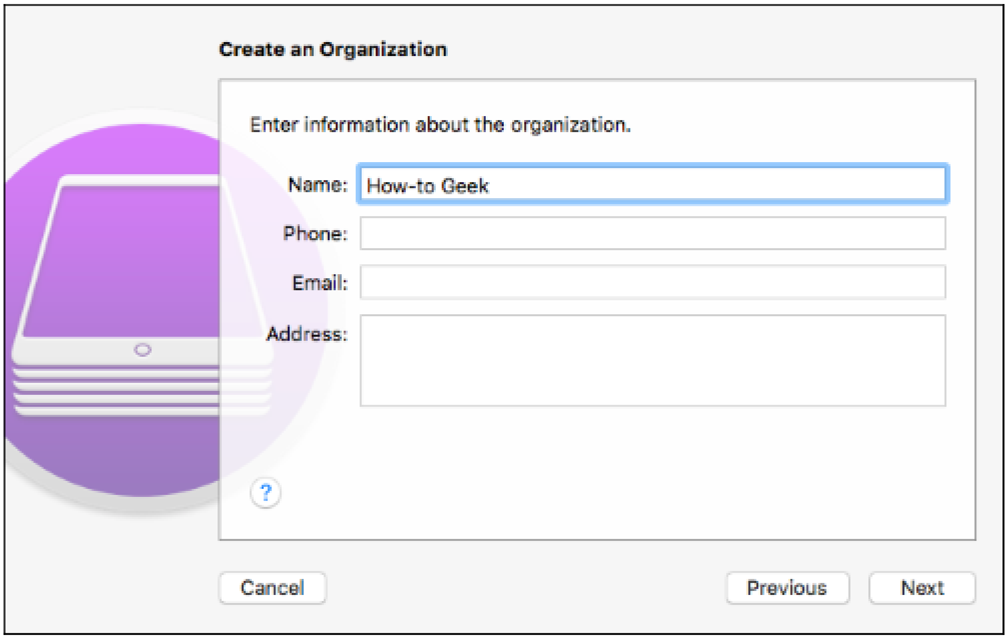
Supervise Devices

Choose whether to supervise the devices, which allows an additional set of more intrusive settings to be configured. If supervising, choose whether to allow the devices to pair with any other host, or only to Configurator hosts provisioned with the supervising organization.

☑ Supervise devices

☑ Allow devices to pair with other computers

11. Enter your organization information.



Create an Organization

Enter information about the organization.

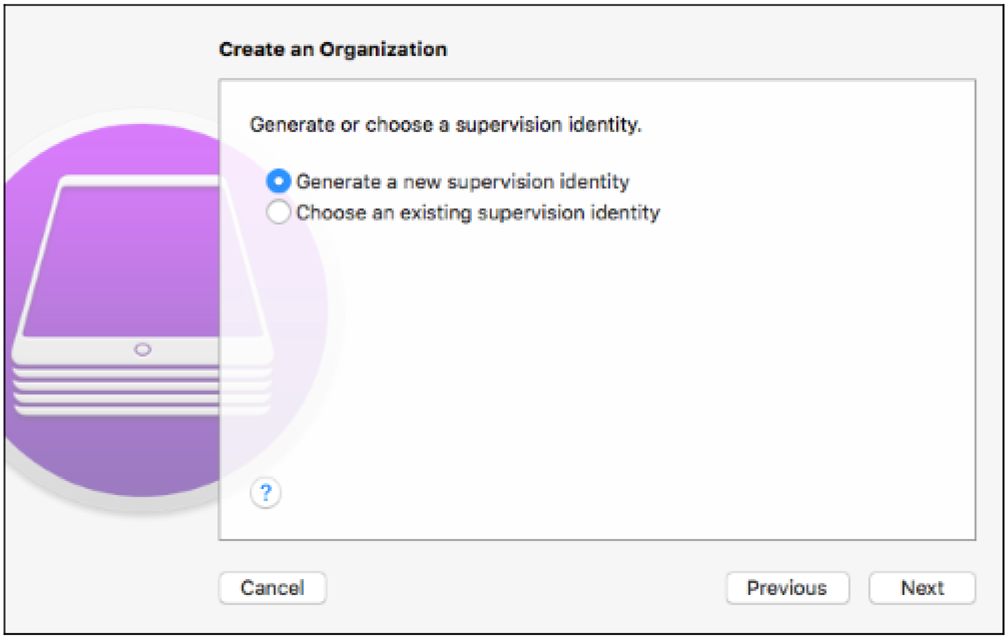
Name: How-to Geek

Phone:

Email:

Address:

12. If you've previously generated a supervision identity at some point, select **Choose an Existing Supervision Identity**. Otherwise, you'll need to generate one by selecting **Generate a New Supervision Identity**.

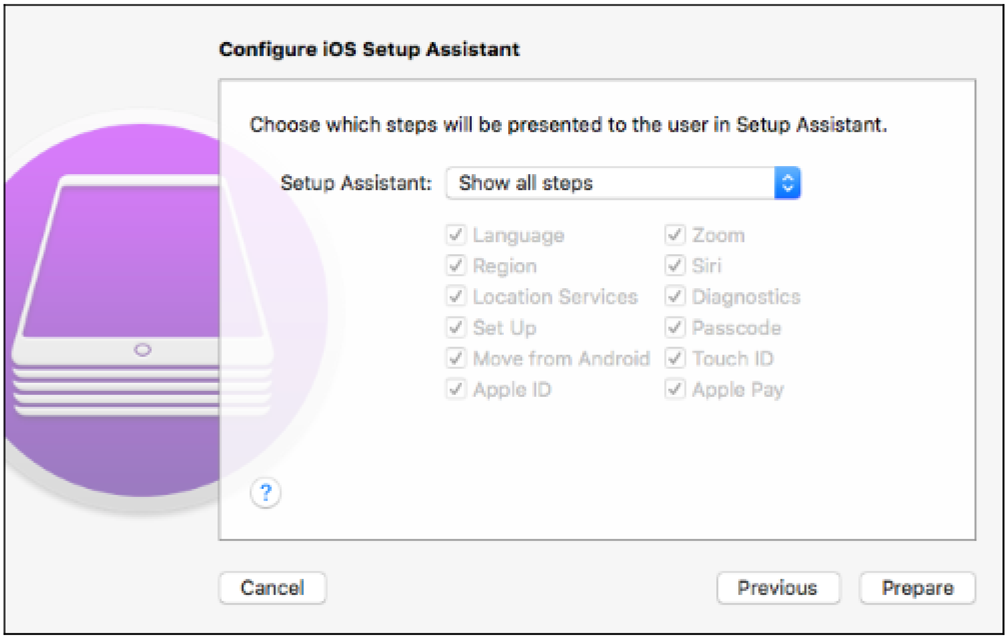


Create an Organization

Generate or choose a supervision identity.

● Generate a new supervision identity
○ Choose an existing supervision identity

13. Select the options you want the device to run after it is reset. The default options are generally sufficient.



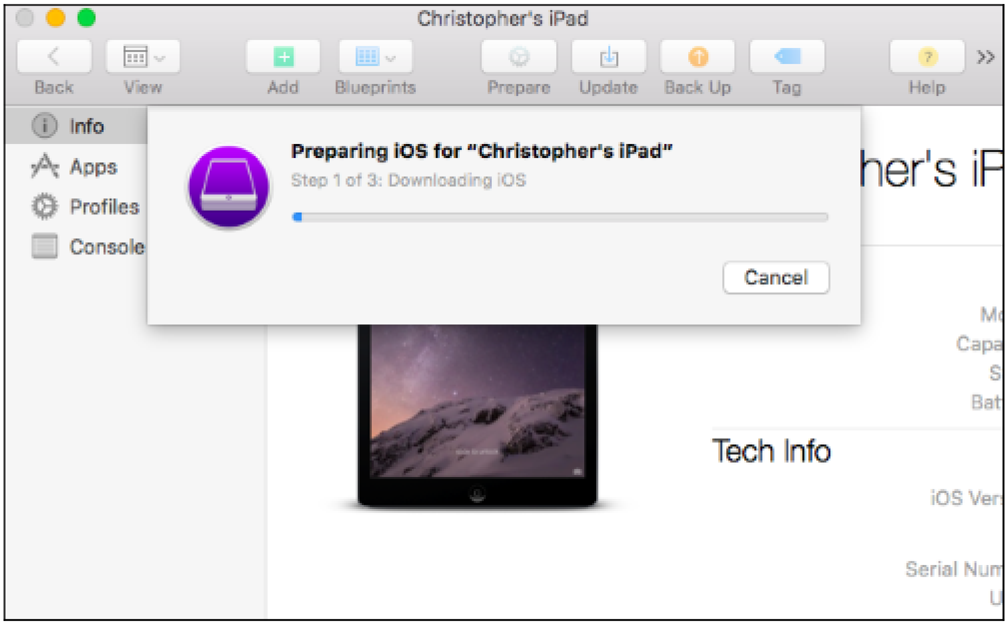
Configure iOS Setup Assistant

Choose which steps will be presented to the user in Setup Assistant.

Setup Assistant: Show all steps

☑ Language
☑ Region
☑ Location Services
☑ Set Up
☑ Move from Android
☑ Apple ID
☑ Zoom
☑ Siri
☑ Diagnostics
☑ Passcode
☑ Touch ID
☑ Apple Pay

14. Click **Prepare**. A status bar will be displayed as ithe iOS device is configured in supervised mode. **WARNING:** Clicking the **Prepare** button wipes all information on the device and resets it.



Preparing iOS for "Christopher's iPad"

Step 1 of 3: Downloading iOS

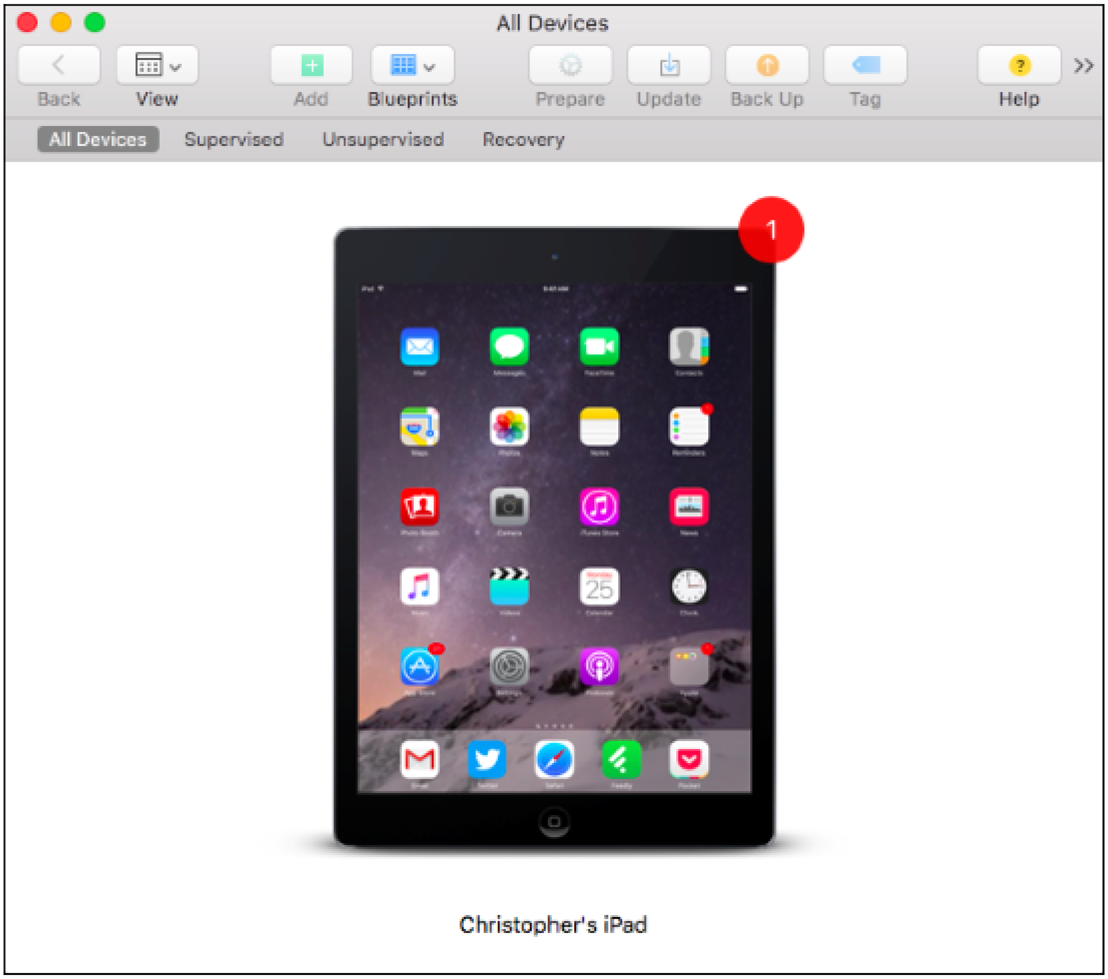
After the device is wiped and rebooted it will be running in supervised mode.

74.2.1 Enable Single App Mode

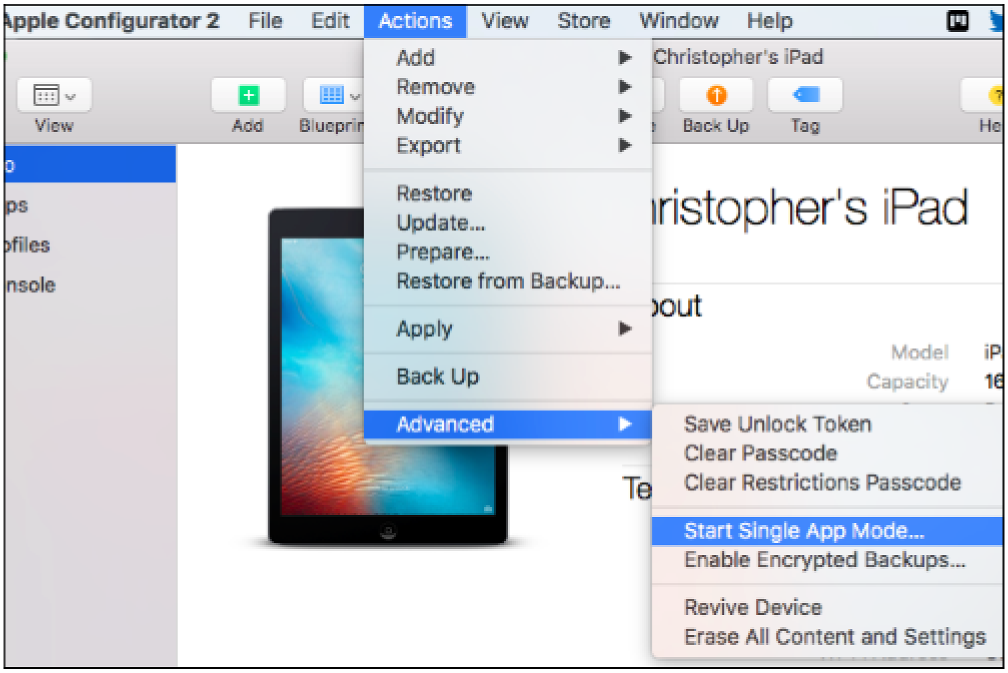
Note: To continue from this point, the iOS device should be in supervised mode. If the iOS device is not in supervised mode, repeat the instructions from the prior section first to put it in supervised mode.

To enable or disable Single App Mode, do the following:

1. On your Mac running 10.14 or greater Mojave, launch the *App Store* application and search for *Apple Configurator 2*.
2. Download and install *Apple Configurator 2* to your Mac.
3. Plug in your iOS device to your Mac computer.
4. Launch *Apple Configurator 2*. You should see something that looks like the image below. Double-click the device.

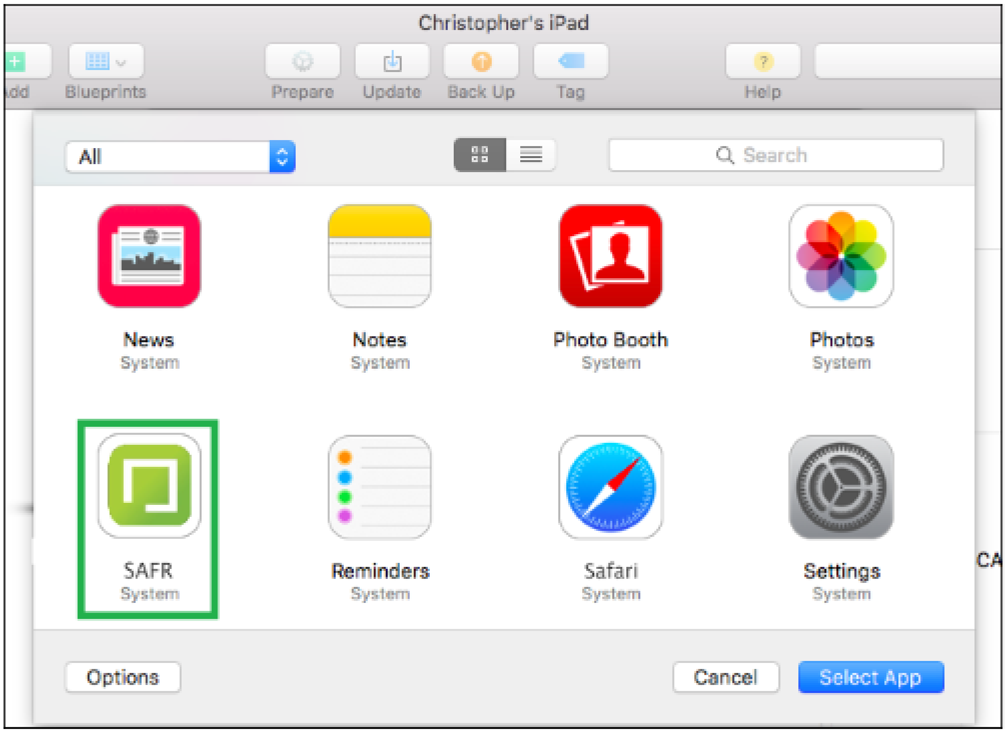


5. On the **Device Details** screen, from the **Actions** menu, click **Advanced > Start Single App Mode**.

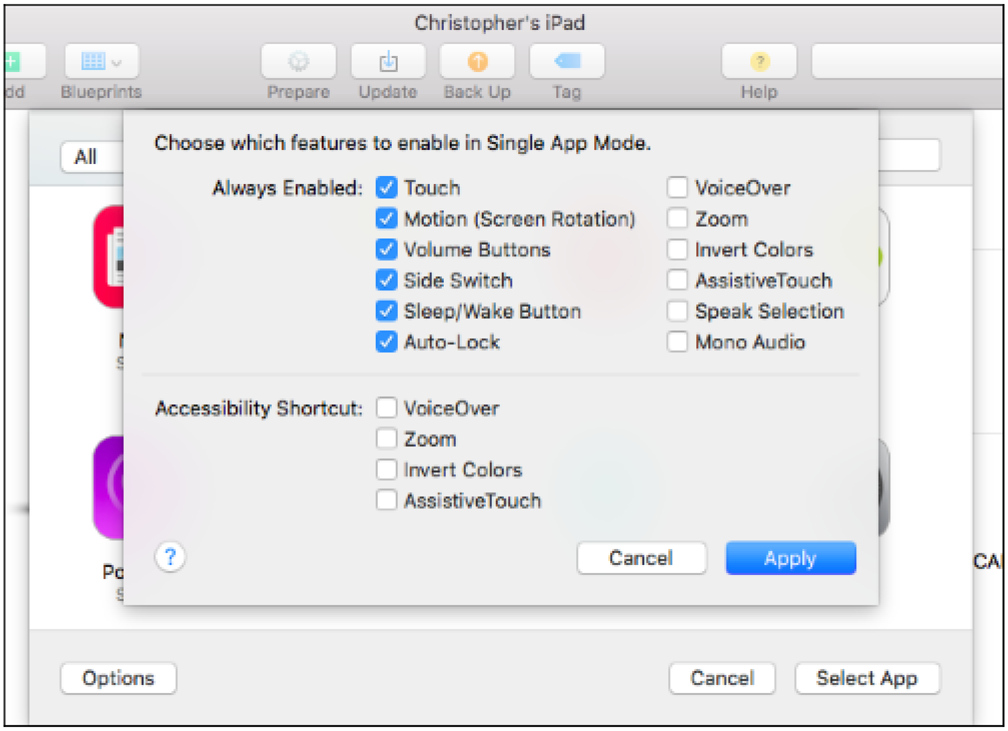


6. Select SAFR from the list of applications.

7. Click the **Select App** button when you're ready to launch SAFR. The iOS device is now locked in Single App Mode.



8. OPTIONAL: If you want to configure advanced options, click **Options**. From the dialog, select the options you want enabled, and click **Apply**. However, usually the defaults are sufficient.



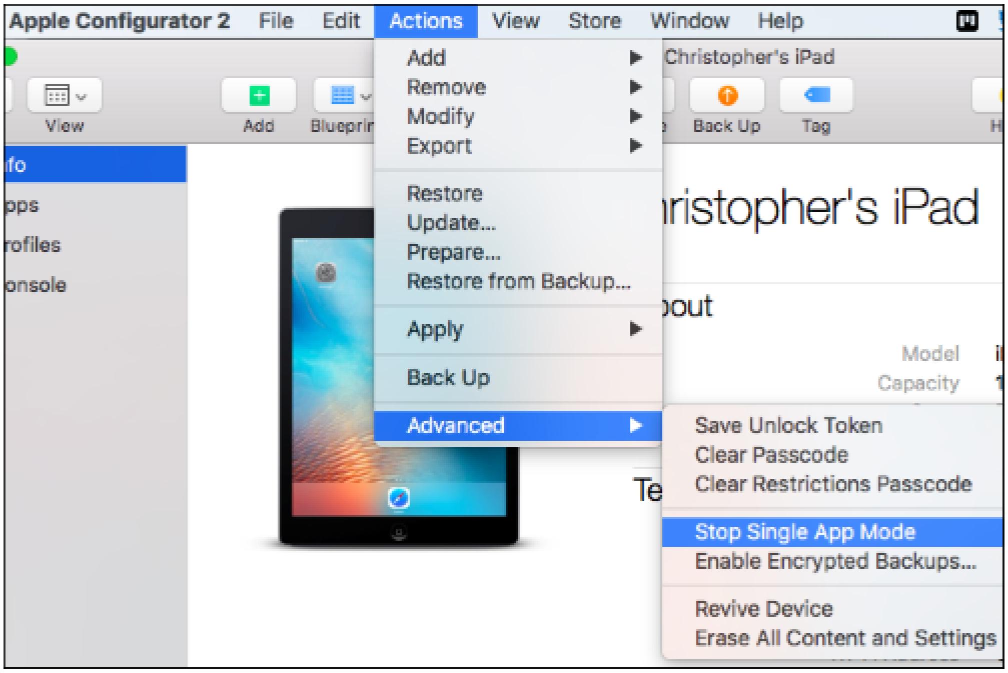
Choose which features to enable in Single App Mode.

Always Enabled: ☑ Touch ☑ Motion (Screen Rotation) ☑ Volume Buttons ☑ Side Switch ☑ Sleep/Wake Button ☑ Auto-Lock □ VoiceOver □ Zoom □ Invert Colors □ AssistiveTouch □ Speak Selection □ Mono Audio

Accessibility Shortcut: □ VoiceOver □ Zoom □ Invert Colors □ AssistiveTouch

? Cancel Apply

9. When you return to the applications screen, click the SAFR application and click Select **App**.
10. To disable Single App Mode, plug the iOS device into the computer. In the **Actions** menu, click **Advanced > Stop Single App Mode**.



74.3 Enable Kiosk Mode for Android

There are two kiosk modes available in the Android Mobile Client:

- Lock Task mode (LTM): A robust kiosk mode where only administrators are able to alter the configuration or access the data on the device. The device is locked into one application until the mode is explicitly disabled. You must install the Mobile Client using SAFR Beam to use this mode.

- Screen Pinning mode (SPM): A less secure kiosk mode without device administrator registration. When using the device you can exit the mode at any time. Available for any Android device with the Mobile Client installed.

Note: While this procedure explains how to manually set up a device using SAFR Beam, you can also use the Android Debug Bridge (ADB) command line tool.

To set up and enable Lock Task mode:

1. Go to the SAFR download portal and from the menu, select Android.
2. Install SAFR Beam on your primary device.
3. Set your target device in factory reset prior to use.
4. Follow the instructions on the primary device for installing the Mobile Client on a target device.
5. Once the Mobile Client is installed on the target machine, click the lock icon next to the settings gear icon. Follow the instructions for setting the device up for Lock Task mode.
   Note: In this mode, the client has full control over the device and only the client can request exiting the mode.
6. Exiting can be done by tapping the screen three times (3-taps gesture) which displays the system's security dialog. (assuming that one has been configured) In the dialog, you are prompted to confirm your identity by entering the device's credentials (PIN, gesture, or fingerprint). If the device does not have security settings in place or your identity is confirmed, the Mobile Client restarts in an unlocked state.

Important: You should configure device security either with a PIN, a gesture, or a fingerprint. That way, if a device is turned off while the Mobile Client is locked (either by the power button or as the result of drained battery), only a credible user is able to start the device and re-run the Mobile Client. When re-run, the Mobile Client enters the mode it was in prior to turning off the device.

Note: If you install the Mobile Client apart from SAFR Beam, you can still set up security by clicking the lock icon. However, because the Mobile Client has not been registered as a device administrator, its security is not as strong as the Lock Task mode.

The following scenarios occur when using the kiosk modes when the Mobile Client is or is not registered as a device manager:

Scenario	Action
No device security configured (not registered); you confirm to enter SPM on the security dialog	Exits via 3-taps gesture, or by holding the Recents and Back keys at the same time; the Mobile Client is restarted in unlocked state (Screen Pinning mode)
No device security configured (not registered); you deny entering SPM on the security dialog	The Mobile Client is in locked state but is restarted in unlocked state after approximately ten (10) seconds; a timer is triggered that queries for locked state and corrects it if needed
PIN device security configured (registered); you confirm to enter SPM on the security dialog	Exit by 3-taps gesture or by holding the Recents and Back keys at the same time; SAFR prompts you to confirm your identity by entering PIN and if successful, it is restarted in unlocked state

Note: On some devices, SPM can be explicitly enabled in system's setting with an option to ask for a PIN upon unlocking/PIN device security configured. If you confirm to enter SPM on the system dialog by exiting by holding the Recents and Back keys at the same time, you are prompted to confirm your identity by entering PIN. If successful, the device home screen is displayed. The next time, SAFR restarts in an unlocked state.

75 Install SAFR Beam

Install the SAFR Beam for Android utility onto one device and use this primary device to install the Mobile Client in a Lock Task kiosk mode on a second target device. Using SAFR Beam provides added security to the target device, locking it down in cases where added security is required. For example, using the device camera to identify employees and open secured door to them. For more information, see Configure Devices into Locked Mode section.

75.1 To Install and Use SAFR Beam

1. Secure two Android devices capable of running SAFR. One device serves as the primary and the other as the target.
2. Log into the SAFR Download Portal and install SAFR Beam on the primary device.
3. On the primary device, turn on Near Field Communication (NFC). Make sure the target device has NFC capabilities.
4. Reset the target to its factory settings.
5. Place the target device back to back with the primary device.
6. Once the target device is detected, tap the screen on the primary device to start the beam.
7. Follow the instructions on the target device to complete the installation.
8. Although not required, we highly recommend that you set up security access on the target device. (e.g. a PIN or gesture)
9. Run the Mobile Client on the target device. If prompted, set SAFR as the default launcher app.

76 Mobile Accounts Preferences

The Account preferences tab is where you configure your organization's SAFR accounts and related information, such as the directory for your facial recognition database.

- **Environment**: Determines which operating environment your client contacts. The possible values for this field are as follows:
  - *SAFR Developer Cloud*: Internal use only.
  - *SAFR Partner Cloud*: Internal use only.
  - *SAFR Cloud*: Used for cloud deployments. This is a general availability SAFR Server in the cloud maintained by RealNetworks. It is a stable, high availability environment intended for production use.
  - *SAFR Custom*: Used for on-premise deployments. If you select *SAFR Custom*, you will be asked to provide the URLs for your primary SAFR Server services.
- **User Identifier**: The account can have multiple user identifiers with different access privileges.
- **User Password**: The password for the user entered in the *User Identifier* field.
- **User Directory**: The directory in the account where the data used for facial recognition is stored.
- **User Source**: The *User Source* label for this mobile device. All SAFR event data is tagged by site and source labels. These labels are used to help filter and analyze collected recognition events, such as where a face was recognized.
- **User Site**: The *User Site* label for this mobile device. All SAFR event data is tagged by site and source labels. These labels are used to help filter and analyze collected recognition events, such as where a face was recognized.
- **Similar Directory**: The directory to use when using the *Similar* video processing mode. If this setting is left blank, then the User Directory is used when using Similar mode.
- **Report Status**: Enables a preview of the video stream in the video feed status window. The feed view is a simple low frame-rate stream (1 frame per second). It is only intended for inspecting camera orientation and lighting conditions. It is not intended for actively monitoring feeds for security purposes.
  - **Allow Remote Viewing**: Enables remote monitoring for your mobile device's video feed.
- **Version**: The SAFR version that's currently running.

77 Mobile Detection Preferences

Use detection preferences to enable and configure facial detection characteristics.

77.1 Enable Face Detector

The **Enable face detector** check box must be selected to enable face recognition.

- **Max Vertical Resolution**: Specifies the maximum supported vertical resolution.
- **Min Searched Face Size**: Defines the minimum face size that can be detected. A searched size of 80, for example, can still manage to detect faces as small as 60x60, but with lower certainty. Lowering this number enables SAFR to detect much smaller faces but also greatly increases CPU usage.
  Note: This setting does not impact face recognition accuracy.
- **Min Required Face Size**: Defines the minimum required size for a face to be detected. Any face smaller than the height or width is ignored.
- **Generate Recognizer Hint**: Optimizes facial recognition. It should be turned on for most cases. If it is turned off, recognition accuracy may be reduced if detection is performed at very low resolutions.
- **Detection Service**: Specifies which face detection service will be used.
  - *Standard*: Android only. The standard facial detection model that SAFR uses.
  - *High Sensitivity*: Android only. A high sensitivity facial detection model which has a lower latency and whose performance doesn't degrade when multiple faces are being analyzed simultaneously. This model consumes many more GPU resources than the *Standard* model.
  - *SAFR*: iOS only. The standard facial detection service that SAFR uses.
  - *SAFR Retina*: iOS only. A high sensitivity facial detection service which has a lower latency and whose performance doesn't degrade when multiple faces are being analyzed simultaneously. The SAFR Retina service consumes many more GPU resourcs than the SAFR service.
  - *Coreimage*: iOS only. A face detection service native to Apple. Its performance is comparable to SAFR's, but it's slightly less accurate.
  - *Vision*: iOS only. A face detection service native to Apple that's faster but much less accurate.
- **Input Size**: Android only. This setting is only available if you selected *High Sensitivity* for the **Detection service** setting above. The **Input Size** setting allows you to manage the trade-off between accuracy vs. speed. There are 4 possible values:
  - *Normal*: This is the standard against which the other 3 possible values are measured.
  - *Small*: This value has decreased accuracy but increased speed.
  - *Extra Small*: This value has greatly decreased accuracy but greatly increased speed.
  - *Large*: This value has increased accuracy but decreased speed.
- **Detection thresholds**: Allows you to configure the detection thresholds.
  - **Enable Custom Thresholds**: Allows you to customize the detection threshold. When this setting is checked, you can click on the *Configure* button to do the customization.
    - **Initial Candidate Threshold**: Initial face candidate threshold that is used during face detection.
    - **Middle Candidate Threshold**: Middle face candidate threshold that is used during face detection.
    - **Final Candidate Threshold**: Final face candidate threshold that is used during face detection.
  - **Detection Sensitivity Threshold**: The lower this value is, the more lenient the facial detection service will be when attempting to recognize a face, which can result in additional false positives.
- **Processors #**: Android only. Specifies how many processors will be used for face detection. Selecting more processors results in more face detections per second, but it also consumes more system resources, which can slow everything down resulting in ultimately fewer recognition attempts per second.
- **Validation service**: iOS only. Requires a mobile device with a TrueDepth camera. When set to *Real Face*, this setting uses the device's TrueDepth camera to do a liveness test on any faces in view of the camera. If the faces pass the test, then they are indeed real faces. Any faces that fail the test are probably spoofs. If this setting is set to *None*, or if the device doesn't have a TrueDepth camera, then no liveness test is performed.

77.2 Enable Liveness Detector

Android only. Enables RGB liveness detection. RGB liveness detection is only available if you selected *High Sensitivity* for the **Detection service** setting above. For a full description of how RGB liveness works, please see the RGB liveness detection topic.

- **Minimum required face size**: The minimum required height and width of a face, in number of pixels, for the *Texture Model* to be used.
- **Minimum required face context size**: The minimum required extra context around faces for the *Context Model* to be used.
- **Minimum required center pose quality**: The minimum face center pose quality for RGB liveness detection to be used.
- **Minimum required face sharpness quality**: The minimum face sharpness quality for RGB liveness detection to be used.
- **Minimum required face contrast quality**: The minimum face contrast quality for RGB liveness detection to be used.
- **Minimum preliminary liveness threshold**: For multimodal *detection schemes*, this is the liveness threshold which the first evaluated model (the *Texture Model*) must exceed before SAFR will bother evaluating the second model. If this threshold is not met, SAFR immediately returns NOTLIVE_CONFIRMED for the subject.
- **Liveness detection threshold**: Specifies how difficult it will be for a subject to be verified as LIVENESS_CONFIRMED.
- **Fake detection threshold**: Specifies how difficult it will be for a subject to be verified as NOTLIVE_CONFIRMED.
- **Detection scheme**: Specifies which RGB liveness model(s) should be used.
  - *Texture Unimodal*: Only the Texture model will be used.
  - *Context Unimodal*: Only the Context model will be used.
  - *Strict Multimodal*: Both the Texture and Context models will be used. For a subject to pass the RGB liveness test, both of the results of the models must meet or exceed the *Liveness detection threshold* value. This is the default option.
  - *Normal Multimodal*: Both the Texture and Context models will be used. For a subject to pass the RGB liveness test, the average of the results of the two models must meet or exceed the *Liveness detection threshold* value.
  - *Tolerant Multimodal*: Both the Texture and Context models will be used. Subjects pass the RGB liveness test when the result of either model meets or exceeds the *Liveness detection threshold* value.
- **Evaluate liveness over N frames**: The number of frames over which liveness should be evaluated.
- **Evaluate fake over N frames**: The number of frames over which fakeness should be evaluated.
- **Minimum confirmations required**: The percentage of frames that must meet the liveness or fake threshold for the subject to be declared either LIVENESS_CONFIRMED or NOTLIVE_CONFIRMED.

77.3 Enable Badge Detector

iOS only. The **Enable badge detector** check box must be selected to enable badge detection. Badges are visual representations of users that are quicker and easier to detect than faces. Compared to faces, they are easier to detect and recognize when rotated and when used in low light conditions.

- **Max Vertical Resolution**: Specifies the maximum supported vertical resolution.
- **Min searched badge size**: Defines the minimum badge size that can be detected. Lowering this value enables SAFR to detect very small badges (down to 15x15 pixels) at the cost of increasing the CPU usage. Conversely, increasing this value reduces CPU usage but requires larger badges for successful detection.
- **Min required badge size**: Use this setting to require a minimum badge size in pixels. Any badge smaller than this value in either height or width is ignored. This setting is mainly used when badge detection is only expected to occur when the badges are close to the camera. (When cameras are close

to the camera, the badge sizes are guaranteed to be larger.) If this value is set too small, SAFR could create many detection events for badges that are far from the camera and therefore not of interest.

- **Consecutive confirmations required**: This setting adds consecutive confirmations to the SAFR detection to create more reliable detections. Increase this setting for more reliable but slower badge detection. Decrease it for faster but slightly less reliable detection.
- **Detection service**: Specifies which badge detection method will be used. Depending on the capabilities of your cameras, lighting conditions, and other variables, certain options may work better with your environment than others.
  You can choose from the following options:
  - **apriltags**: Basic badge detection.
  - **rhinotagsLite**: The fastest badge detector, but it has a lower tolerance for motion blur. It requires cameras with a fast shutter speed.
  - **rhinotagsTeam**: Faster badge detector, but it has little resilience to motion blur.
  - **rhinotagsFlex**: Fast badge detector with moderate resilience to motion blur.
  - **rhinotagsFull**: Badge detector with robust handling under various conditions and a strong resilience to motion blur.
    rhinotagsFull is the recommended option.
    A full set of badge images supported by SAFR is available at https://github.com/anqixu/apriltag/tree/master/tag36h11.
  - It is recommended these images be re-sized to at least 2" x 2" size using the nearest neighbor algorithm (to maintain sharp edges) before use with SAFR.
  - Although a single badge of displayed format can express only 587 different IDs, multiple badges can be combined to increase the number of expressible IDs into the billions. For example, using 6 badges providesover 827 billion expressible IDs.

77.4 Enable Object Detector

iOS only. Internal use only.

78 Mobile Recognition Preferences

Use Recognition preferences to adjust the range for a variety of settings that determine whether or not SAFR detects, tracks, and recognizes faces and identities.

- **Detect Identity**: Select to enable identity detection.
- **Detect Gender**: Select to enable gender detection.
- **Detect Age**: Select to enable age detection.
- **Detect Mask**: Enables the detection of masks.
    - **Mask Model**: iOS only. Specifies the model to be used for mask detection.
        - *precise*: This model produces the least number of false positives (i.e. detecting that a person is wearing a mask but there is no mask), but it suffers from the lowest true positive rate. (i.e. detecting masks that are actually there)
        - *sensitive*: This model produces the highest true positive rate, but it suffers from the highest number of false positives.
        - *normal*: This model produces a moderate amount of both false positives and true positives.
    - **Mask Threshold**: iOS only. Specifies the threshold at and above which mask detection will conclude that *mask=true*.
    - **Consecutive confirmations required**: iOS only. Adds additional required consecutive confirmations for SAFR to recognize faces covered by masks. Increasing this setting creates more reliable but slower facial detection.
- **Detect Occlusion**: Obstructing the full view of the face by using, for example, a mask, glasses, or using a hand to block a part of the face.
- **Detect Sentiment**: Select to enable sentiment detection.
- **Detect RGB Liveness Action**: Android only. Enables the RGB liveness recognizer. When enabled, this recognizer creates events based on the RGB liveness feature when cameras view somebody enrolled in your SAFR Identity Database.
    - **Consecutive recognitions for live**: Android only. Number of consecutive recognition attempts that must be successful for a LIVENESS_CONFIRMED event.
    - **Consecutive recognitions for fake**: Android only. Number of consecutive recognition attempts that must be successful for a NOTLIVE_CONFIRMED event.
    - **Identity recognition threshold boost**: Android only. The amount to temporarily boost identity recognition attempts during RGB liveness actions.
- **Detect Smile Action**: Select to enable smile detection.
    - **Pre-smile Delay (seconds)**: The amount of time that there should be no smile.
    - **Smile Duration (seconds)**: The amount of time that the smile should last.
    - **Identity Threshold Boost**: The smile threshold to boost temporarily during the smile action.
    - **Transition Thresholds**: The top slider is the smile threshold, while the bottom slider is the nosmile threshold.
      When *Detect Sentiment* is enabled the server returns a sentiment value as well as a smile flag. The smile flag is set based on the sentiment value as calibrated by the server. If you want the Mobile Client to override the sentiment value that is used to indicate a smile then you can set the smile threshold value. If this value is set, it'ss used to override the smile flag returned from the server. If this value is set to nil then the smile flag returned from the server is used.
- **Detect pose liveness action**: iOS only. Enables the pose liveness action recognizer. See Pose Liveness Detection for more information. When enabled, you can click on **Configure** to expose the following configuration settings:
    - **Center pose quality**: Minimum center pose quality to use when detecting the initial center pose.
    - **Profile pose quality**: The maximum center pose quality to use when detecting the final profile pose.
    - **Max profile confidence at start**: Maximum profile pose confidence to allow during the initial center pose detection phase.
    - **Min profile confidence at end**: Minimum profile pose confidence to allow during the final profile pose detection phase.
    - **Min profile pose yaw**: The minimum profile pose yaw value that is required during the final

profile pose detection phase.
    - **Center pose consecutive confirmations**: Number of consecutive center pose confirmations required to enter the initial center pose detection phase
    - **Profile pose consecutive confirmations**: Number of consecutive profile pose confirmations required to enter the initial center pose detection phase.
    - **Min profile similarity**: Minimum similarity score required when verifying the final profile pose.
    - **Min detections per second**: Minimum number of frames per second required during the process.
    - **Min transition poses**: Minimum number of required center pose samples during the transition from center to profile pose.
    - **Max CPQ jump in continuous tracking**: Maximum change between samples while the pose is changing from center to profile.
    - **Max CPQ jump after tracking loss**: Maximum change between samples while the pose is changing from center to profile if lingering.
    - **Max profile pose roll**: The maximum roll threshold in either direction in which the face can rotate when determining whether the face is in profile pose.
- **Detect Mask Action**: iOS only. Enables the detection of mask event types. Mask event detection attempts can return 3 potential results: *mask=false*, *mask=indeterminate*, or *mask=true*. After the specified number of consecutive mask event detection results, (configured below) the mask event state is set to the appropriate value. The mask event state can only progress from false towards true; the state never regresses back towards false. For example, once the mask event state for a viewed person becomes set to *mask=true*, then that person's mask event state won't ever regress to *mask=indeterminate* or *mask=false*.

  Events are generated when the mask event state is set to either *mask=false* or *mask=true*.

  Note that enabling this setting automatically enables the **Detect Occlusion** setting above, since detecting mask actions inherently involves detecting occlusion at the same time.
    - **Min Face Size**: The smallest face size, in pixels, upon which the Mobile Client will attempt to detect a mask.
    - **Min Consecutive Mask Detections**: Specifies the minimum number of consecutive *mask=true* mask detection results that must occur before the Mobile Client will generate a *mask=true* event.
    - **Min Consecutive No Mask Detections**: Specifies the minimum number of consecutive *mask=false* mask detection results that must occur before the Mobile Client will generate a *mask=false* event.
    - **Min Consecutive Occluded No Mask Detections**: Specifies the minimum number of consecutive *mask=indeterminate* mask detection results that must occur before the Mobile Client will set the mask event state to *mask=indeterminate*.
    - **Face Edge Threshold**: How far a face must be from the edge of the screen before a mask event detection is attempted. For example, if a face is 100 pixels, and **Face Edge Threshold** is set to 3%, then the face must be 3 pixels from the edge of the screen before SAFR will attempt a mask event detection.
- **Face quality settings**:
    - **Minimum required center pose quality** iOS only.
        - **For recognition**: Defines the minimum required quality for a face posed directly in front of the camera (center posed) to attempt recognition. Center pose quality (CPQ) ranges from one to zero. A score of 1 is given to a face looking straight into the camera. Any deviation from this position diminishes center pose quality. Center pose quality of a face in full profile position is given a score of zero. Recognition from any pose is possible, but accuracy is reduced for faces that are in extreme profile positions.
        - **For merging**: Defines the minimum required face center pose quality to attempt merging with existing reference images for a recognized identity.
        - **For learning/strangers**: Defines the minimum required face center pose quality to enable SAFR to store a reference image for a new identity.
    - **Minimum required face sharpness quality** iOS only.
        - **For recognition**: Indicates minimum required face sharpness quality to attempt recognition.

- **For merging**: Indicates minimum required face sharpness quality to attempt recognition.
- **For learning/strangers**: Indicates minimum required face sharpness quality to store as a reference for a new identity.
- **Minimum required face contrast quality** iOS only.
  Contrast quality defines the difference between the color of a subject's face and the background.
  - **For recognition**: This setting indicates the minimum amount of contrast quality (lower or higher contrast) for SAFR to attempt a recognition.
  - **For merging**: Defines the minimum required face contrast quality to attempt merging a captured face with its existing references in the SAFR system.
  - **For learning/strangers**: Indicates the minimum required face contrast quality to store as a reference for a new identity.
- **Minimum Recognition Face Size (pixels)**: Defines the minimum required face size in pixels to attempt recognition. It includes a 25% margin around the face.
- **Minimum Learning Face Size (pixels)**: Defines the minimum required face size in pixels to enable SAFR to store a reference image for a new identity. It includes a 25% margin around the face.
- **Minimum recognitions to lock on to identity**: iOS only. Minimum number of consecutive recognition attempts that must produce the same identity before SAFR locks onto the identity.
- **Re-recognition Delay (seconds)**: The number of seconds that must pass before SAFR attempts to reconfirm a tracked person.
- **Initial recognition attempts**: iOS only. Number of initial recognition attempts to make on an unrecognized face as quickly as possible.
- **Failed recognition back-off interval**: iOS only. After making the initial recognition attempts as quickly as possible, back up the amount specified by this setting for each subsequent recognition to slow down. This goes on until the retry interval is reached.
- **Retry failed recognitionS after every**: iOS only. The interval in which to run recognition requests if the face has not been recognized.
- **Lingering Timeout (frames)**: Specifies the number of additional frames that SAFR keeps a tracked face in active memory after SAFR has failed to detect it in the most recent frame. Increasing this value makes the tracker resilient against intermittent loss of face, but consumes system processing power.
- **Update identity every**: iOS only. Updates the identity when the currently saved identity is older than the updated identity.
- **Update identity with better image**: iOS only. Updates the identity when the currently saved identity is of lower quality (in all aspects) than the new image.
- **Maximum allowed occlusion**: iOS only.
  - **For learning/strangers**: Indicates the maximum occlusion value allowed for a face to be registered to the Person Directory. When this setting is set to 1, no occlusion filtering is applied, and the default configuration is ignored.
  - **Learn occluded faces**: Enables the learning of occluded faces.
- **Identity recognition threshold**: Determines the strictness of the face recognition when declaring identity matches between a face and stored identity image. You can independently set the Identity Recognition Thresholds for the following:
  - **Camera**: Sets the threshold for images. (e.g. photos)
  - **Similar**: Sets the threshold for *Similar* comparisons when running Similar video processing mode.
  - **Masked face threshold offset**: Sets the threshold when detecting masks.
  - **Proximity threshold allowance**: A boost value that is added to the Identity Recognition Threshold.
    For detailed information about Identity Recognition Threshold and Proximity Threshold Allowance, see Identity Recognition Thresholds.
- **Minimum Learning Pose Centerdness**: Android only. Specifies the minimum center pose quality that a face must have for it to be enrolled in SAFR's Person Directory.
- **Minimum Learning Contrast**: Android only. Specifies the minimum contrast a facial image must have for it to be enrolled in SAFR's Person Directory.
- **Minimum Learning Sharpness**: Android only. Specifies the minimum sharpness that a facial image

must have for it to be enrolled in SAFR's Person Directory.

- **Max Concurrent Recognitions**: iOS only. Specifies the maximum number of concurrent recognitions that the Mobile Client can attempt. We strongly recommend that you do not change this setting; doing so could greatly decrease your device's performance.

79 Mobile Events Preferences

Use the Events preferences tab to configure event reporting as well as how your client listens for event replies.

- **Report Events**: Enables event reporting. Event reporting enables SAFR to log and track events over time and gain additional insight into your SAFR system and usage patterns.
- **Report Action Events Only**: When enabled, only action events will be reported. "Action events" are any events of the following types:
  - *smileToActivate*: The event was triggered by a Smile action.
  - *poseLiveness*: The event was triggered by a successful Pose Liveness Detection.
  - *directionOfTravel*: The event was triggered by a Direction of Travel Recognition.
- **Include Unrecognizable Events**: Enable to report the appearance of unrecognizable people captured by camera feeds. Unrecognized people are people that the SAFR system can't see well enough to compare it to its Person Directory.
- **Include Stranger Events**: Enable this option to report when the appearance ofstrangers. Strangers are people that the SAFR system can see well enough to compare to individuals stored in the People Directory, but for whom there isn't a match.
  - **Min Age**: The minimum age of strangers that will trigger stranger events. If a stranger younger than the specified minimum age is detected, no stranger event is generated.
  - **Max Age**: The maximum age of strangers that will trigger stranger events. If a stranger older than the specified maximum age is detected, no stranger event is generated.
  - **Only if occluded**: iOS only. When enabled stranger events are only reported when the stranger is occluded. This can be useful if you want to catch people who are attempting to bypass your security system by intentionally occluding their faces.
- **Include Speculated Identity Events**: Enables reporting events for speculated people. A "Speculated Identity" is a face that isn't a 100% match with a face in the Person Directory, but is close.
- **Preserve Event Face Image**: Enable if you want the images that trigger an event to be saved with the event report.
- **Preserve Event Scene Thumbnail Image**: Enable if you want a thumbnail of the scene image in which the event occurred to be saved with the event report.
- **Reporting Delay**: The number of seconds an event report is delayed in order to properly assess the nature of the event. For example, a person who may at first seem unknown may become known after a second observation.
- **Update in-progress event attributes**: iOS only. If this is enabled then any event properties that change will be updated at the specified update interval. Many properties do change periodically, such as images or other averages that are continually computed.
  - **Update interval**: iOS only. Specifies the interval time in which to update event properties that change.
- **Update with higher quality image**: iOS only. Update the thumbnail images with higher quality images during the course of the event if possible.
- **Min Identified Event Duration**: The minimum duration required for an event representing a known person to be recorded as an event.

  This setting helps filter out noise or brief appearances that may not be worth reporting as a system event.

  If this setting and **Reporting Delay** have different settings, the greater number is used.
- **Min Unidentified Event Duration**: iOS only. The minimum duration of an event representing an unrecognizable person to be recorded as an event.

  If this setting and **Reporting Delay** have different settings, the greater number is used.
- **Min Stranger Event Duration**: iOS only. The minimum duration of an event representing an unrecognized person to be recorded as an event.

  If this setting and **Reporting Delay** have different settings, the greater number is used.
- Min Unrecognizable/Stranger Event Duration**: Android only. The minimum duration of an event representing an unrecognized or unrecognizable person to be recorded as an event. If this setting and** Reporting Delay** have different settings, the greater number is used.
- **Min Masked Event Portion**: iOS only. Indicates the portion of the qualifying mask detection

samples during the event that are needed to indicate the presence of a mask for mask presence to be attributed to an event.

- **Anonymous event data sharing**: iOS only. When enabled, the Mobile Client automatically collects and aggregates data such as age and gender so that statistics can be gathered about the location where events occurred. No personally identifiable information (including photos) is shared.
- **Listen For Event Replies**: Select to enable listening for event replies. Listening for event replies enables the client to display reply messages on the screen.
- **Display Reply Message**: Select to enable the display of reply messages on the screen.
- **Display until end of event**: iOS only. When enabled, the event reply is continuously displayed until the event ends. When this isn't enabled, the event reply is only shown for a couple seconds.
- **Auto-replies**: iOS only. Internal use only. Do not enable.
- **Sound on Positive Reply**: iOS only. Specifies which sound, if any, is played when a positive event occurs.
- **Sound on Neutral Reply**: iOS only. Specifies which sound, if any, is played when a neutral event occurs.
- **Sound on Negative Reply**: iOS only. Specifies which sound, if any, is played when a negative event occurs.
- **Voice on Positive Reply**: iOS only. Specifies which voice, if any, is played when a positive event occurs.
- **Voice on Neutral Reply**: iOS only. Specifies which voice, if any, is played when a neutral event occurs.
- **Voice on Negative Reply**: iOS only. Specifies which voice, if any, is played when a negative event occurs.
- **Reaction Delay**: Delays the event reporting to the server by the specified number of seconds.

80 Mobile User Interface Preferences

The User Interface preferences tab is where you can customize your user interface.

- **Enable Registration**: Select to enable unknown users to register their faces.
- **Min Age**: The minimum age for unknown users to register their own faces.
- **Person Type**: The default *Person Type* to assign to users who enroll in the Person Directory using this Mobile Client.
- **Home Location**: A string specifying the location of this Mobile Client.
- **Prompt**: A text string that appears on the screen when a user taps the screen.
- **Enable Expiration**: When enabled, registrations done at this Mobile Client will expire after a period of time. For example, you may want to have a bunch of people register in the morning as way of taking attendance, and then all those registrations could expire at the end of the day.
  - **Expiration**: Specifies when registrations will expire.
- **Form**: A customizable form that users will be prompted to fill out when they use the mobile device to register.
- **Disable Reregistration**: iOS only. When enabled, users aren't able to register themselves into SAFR if they're already registered. This means that users are unable to change their information within SAFR. (e.g. their *Person Type*, how their name is spelled, their email address, etc.) When this setting is disabled, users can revise any of their information by reregistering themselves and then re-entering their information, which will overwrite their previously saved information.
  - **Grace period (seconds)**: iOS only. When **Disable Reregistration** is enabled, the **Grace period** setting specifies a number of seconds after registration where users are still able to revise their information. For example, a user could use this time to correct the spelling of their name.
- **Show Video**: When enabled, the mobile device shows live video from its camera on its screen. When disabled, the video is turned off and the screen is black.
- **Show Tracking Frames**: Enables colored indicator frames overlaid around detected faces and objects. See the color codes section here for a description of what each color indicates.
- **Show Landmarks**: Displays the five face landmarks (eyes, nose tip, and the corners of the mouth) on faces viewed by the mobile device's camera. See Interpret Video Feed Overlays for more information about face landmarks.
- **Show Depth Profile**: iOS only. Internal use only.
- **Show Attributes**: When enabled, people's attributes are displayed above their faces on the mobile devioe's screen.
- **Show Names**: When enabled, people's names are displayed as part of their displayed attributes.
- **Flash Names**: When enabled, people's names will flash on the screen when the person is first detected.
- **Average Age & Gender**: During its normal operation, SAFR estimates the age and gender of people in every frame of a video stream independently. Thus, a person's displayed age can fluctuate by 10 years frame to frame. When the *Average Age & Gender* setting is enabled, ages and gender for detected persons are averaged over time, which creates a much smoother and more accurate experience. This setting has no effect on recognized people whose age or gender are specified within the Person Directory; SAFR will always display the stored values.
- **Enable Mirroring**: iOS only. When enabled, the video image is mirrored before detection and recognition operations are executed.
- **Idle seconds to turn off recognition**: iOS only. As long as your mobile device is performing recognition attempts, the Mobile Client won't put your device to sleep. When the device stops performing recogntion attempts and becomes idle, the Mobile Client waits the number of seconds specified by this setting before putting the mobile device to sleep. Anybody can wake the device up by tapping on its screen.
- **Disable Auto-Lock**: iOS only. When enabled, your iOS device's auto-lock feature will be disabled. This setting is disabled by default to help with your device's battery life. However, you should always enable this setting before using your mobile device in a production environment, since your device wouldn't be of much use as a registration kiosk or face recognition panel if it's in locked mode.
- **Highlight Border Thickness**: Use the slider to set the thickness (in pixels) of the frame displayed around faces.

- **Overlay Text Size**: Specifies the size of the text in the video feed overlay.
- **Sentiment Thresholds**: Specifies what range of sentiment values are classified as unhappy, neutral, and happy.
- **Name display message (e.g. #N = Full name, #F = First name, #U = Last name)**: iOS only. Use this option to display a custom message to registered and recognized entrants.
  Use #N as a placeholder for the name of any recognized person. For example, "Welcome, #N" would display "Welcome, <recognized person's name>." The message is only displayed to registered persons.
- **Speak name display message**: iOS only. When enabled, the *Name display message* will be spoken aloud.
- **Name Flash Time**: Specifies the number of seconds a person's name should remain on the screen when the person is first detected by the mobile device.
- **Minimum Name Refresh Time**: Specifies the amount of time that must pass before people's names are flashed on-screen again.
- **Similar Overlay Zoom**: iOS only. Sets the size of the comparison faces.
- **Similar maximum faces to select**: iOS only. Sets the maximum number of faces to compare to the target face.

81 Web Console

The Web Console provides administrators and operators web-based access to the SAFR system. It allows you to make changes to your account, manage the Person Directory, view events in the Events Archive, manage video feeds, and generate reports.

81.1 Access the Web Console with a Cloud Deployment

To access the Web Console with a cloud deployment, do the following:

1. Go to the **Products** tab of the SAFR Download Portal.
2. Click on the **System Console** link located under the first listed product, **SAFR Cloud**.
3. Log in using your SAFR Cloud Account credentials.

It's also possible to go straight to the Web Console login page located at https://safr.real.com/console.

81.2 Access the Web Console with an On-Premise Deployment

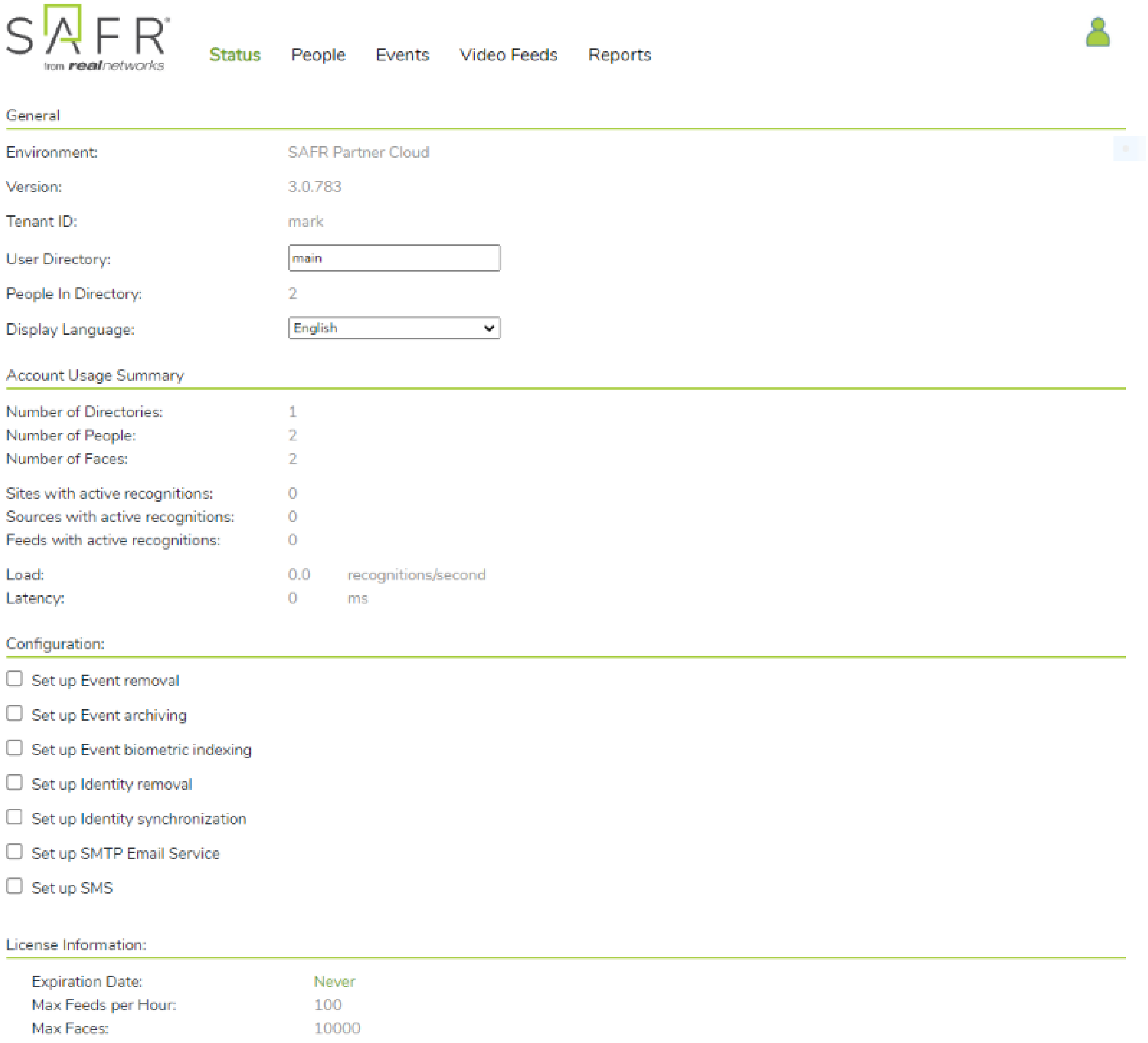
To access the Web Console with an on-premise deployment, you can click on the `SAFR Admin Console` icon that's automatically installed on your desktop on any machine where you've installed the SAFR Platform.

Alternately, you can also access the Web Console using a web browser:

- On your primary SAFR Server, go to http://localhost:8090/.
- On any machine other than your primary SAFR Server, go to `http://<ServerIP>:8090`, where `<ServerIP>` = the IP address of your primary SAFR Server.

82 Status Page

The Status page includes general system, directory, and licensing information. It also allows you to configure some facets of your SAFR system.



General

Environment:	SAFR Partner Cloud
Version:	3.0.783
Tenant ID:	mark
User Directory:	main
People In Directory:	2
Display Language:	English

Account Usage Summary

Number of Directories:	1	
Number of People:	2	
Number of Faces:	2	
Sites with active recognitions:	0	
Sources with active recognitions:	0	
Feeds with active recognitions:	0	
Load:	0.0	recognitions/second
Latency:	0	ms

Configuration:

☐ Set up Event removal

☐ Set up Event archiving

☐ Set up Event biometric indexing

☐ Set up Identity removal

☐ Set up Identity synchronization

☐ Set up SMTP Email Service

☐ Set up SMS

License Information:

Expiration Date:	Never
Max Feeds per Hour:	100
Max Faces:	10000

82.1 General

- **Environment**: Environment associated with the user's account. There are two possible values for this field:
  - *SAFR Cloud*: A SAFR Server in the cloud maintained by RealNetworks. Cloud deployments use this environment.
  - *SAFR Local*: A locally installed SAFR Server that the user maintains. On-premise deployments use this environment.
- **Tenant ID**: The name of the person currently logged in.
- **User Directory**: User directory where the user's data is stored. The default value for this is `main`.
- **People in Directory**: Number of people enrolled in your Person Directory.
- **Display Language**: Language used by SAFR.

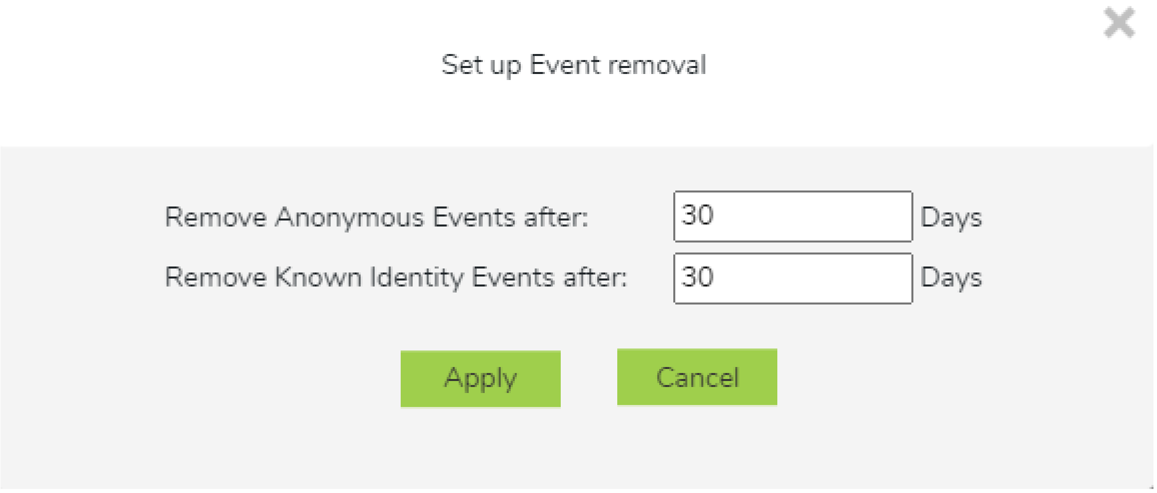
82.2 Account Usage Summary

- **Number of Directories**: Number of directories in your SAFR account.
- **Number of People**: Number of people currently registered.
- **Number of Faces**: Number of faces currently stored in SAFR's database.
- **Number of Sites**: Number of defined sites. A site can consist of one or more cameras, although usually it consists of multiple cameras.
- **Number of Sources**: Number of defined sources. A source can consist of one or more cameras, although usually it consists of a single camera.
- **Number of Feeds**: Number of feeds currently running across the SAFR system.
- **Load**: Number of recognition attempts every second across all video feeds that are currently active in your SAFR system.
- **Latency**: Number of milliseconds it takes for your SAFR Server to generate a response after it receives a recognition request from a client.

82.3 Configuration

82.3.1 Set up Event removal

Enables the automatic removal of events after the specified time interval.



Set up Event removal

Remove Anonymous Events after: 30 Days

Remove Known Identity Events after: 30 Days

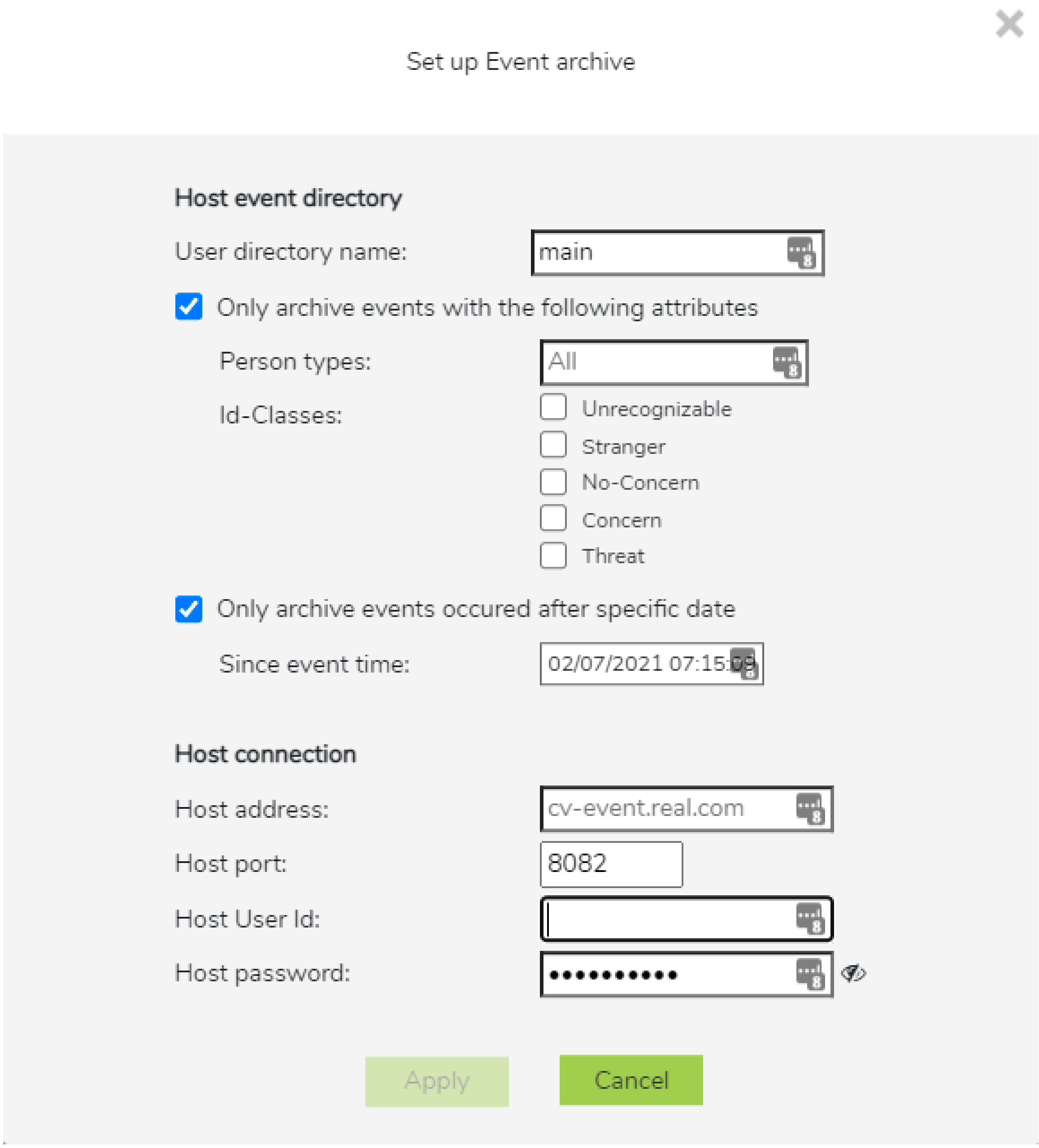
- **Remove Anonymous Events after**: Determines how many days to wait before removing events triggered by people without a *name* attribute. Floating point numbers are valid. If this value is set to zero, then anonymous events won't be automatically removed.
- **Remove Known Identity Events after**: Determines how many days to wait before removing non-anonymous events. Floating point numbers are valid. If this value is set to zero, then non-anonymous events won't be automatically removed.

82.3.2 Set up Event archive

Enables your event archive to automatically sync with another SAFR account's event archive. The target SAFR account can be on the same SAFR Server, or on a different SAFR Server. Your event archive will sync with the target Event Archive once every 10 minutes. Events are guaranteed to be synced; if an error is encountered while attempting to sync, the sync will be repeatedly retried untl it's successful.

Note: This sync does not sync event deletions. Thus, if an event is deleted in the target event archive, that event will NOT automatically be deleted in your event archive as well.

Selecting the *Set up Event archive* box causes the following dialogue to appear:



Set up Event archive

Host event directory

User directory name: main

☑ Only archive events with the following attributes

Person types: All

Id-Classes:
□ Unrecognizable
□ Stranger
□ No-Concern
□ Concern
□ Threat

☑ Only archive events occured after specific date

Since event time: 02/07/2021 07:15:08

Host connection

Host address: cv-event.real.com

Host port: 8082

Host User Id:

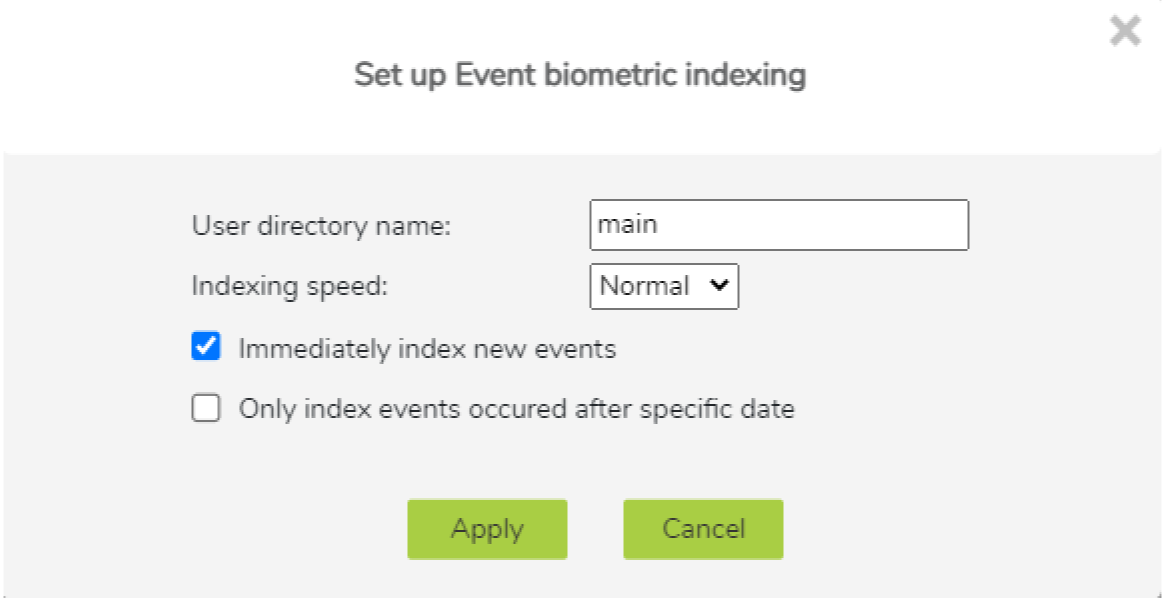
Host password: ●●●●●●●●●●●

- **User directory name**: The name of the user directory whose event archive you're trying to sync with.
- **Only archive events with the following attributes**:
  - **Person types**: Only events generated by the specified *Person types* will be synced. If this field is left blank, then events generated by any *Person type* will be synced.
  - **Id-Classes**: Only events generated by the specified *Id Classes* will be synced. If no *Id Classes* are selected, then events generated by any *Id Class* will be synced.
- **Only archive events occured after specific date**: If selected, then only events generated after the specified date will be synced.
- **Host address**: The IP address or the hostname of the target host machine.
- **Host port**: The port number that the target machine's CVEV server listens on. This field defaults to CVEV's default port number, 8082.
- **Host User Id**: The *User Id* of somebody who has the credentials to log into the host machine.
- **Host password**: The *password* of somebody who has the credentials to log into the host machine.

82.3.3 Set up Event biometric indexing

Sets up biometric indexing on events. Biometric indexing is required to allow event searching by image on the Web Console's Events Page or the Desktop Client's Search by Image Window.

This setting is only visible to users with CONFIG_PRIVILEGE or SUPER_CONFIG_PRIVILEGE privelege levels. See the Desktop Client's Manage Users Preferences documentation for information about user privilege levels.



Set up Event biometric indexing

User directory name: main

Indexing speed: Normal

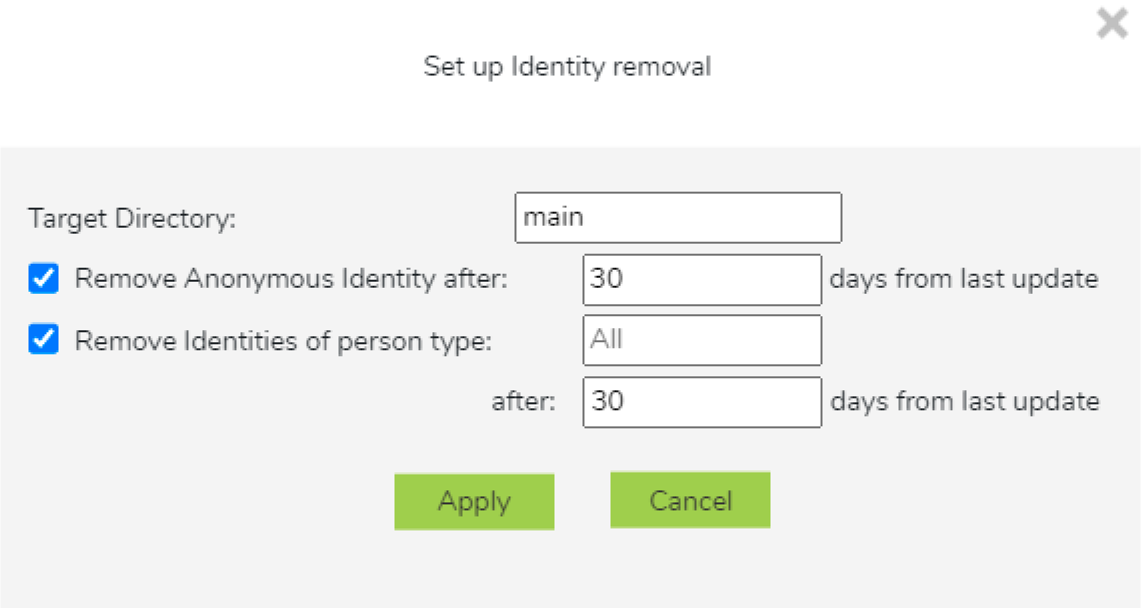
☑ Immediately index new events

□ Only index events occured after specific date

- **User directory name**: The name of the user directory whose event archive you want to biometrically index.
- **Indexing speed**: The speed at which the event can be located when search by image is executed. Faster indexing speeds can lower system performance.
- **Immediately index new events**: Specifies if events should be biometrically indexed as soon as they're created. Enabling this option can affect your system performance when events are created.
- **Only index events occurred after specific date**: When checked, specifies the date after which events should be biometrically indexed. If this checkbox isn't checked, then all the events in the event archive are indexed.

82.3.4 Set up Identity removal

Enables the automatic removal of identities after the specified time interval has passed since the identity was last updated (i.e. the last time the person was seen).



Set up Identity removal

Target Directory: main

☑ Remove Anonymous Identity after: 30 days from last update

☑ Remove Identities of person type: All

after: 30 days from last update

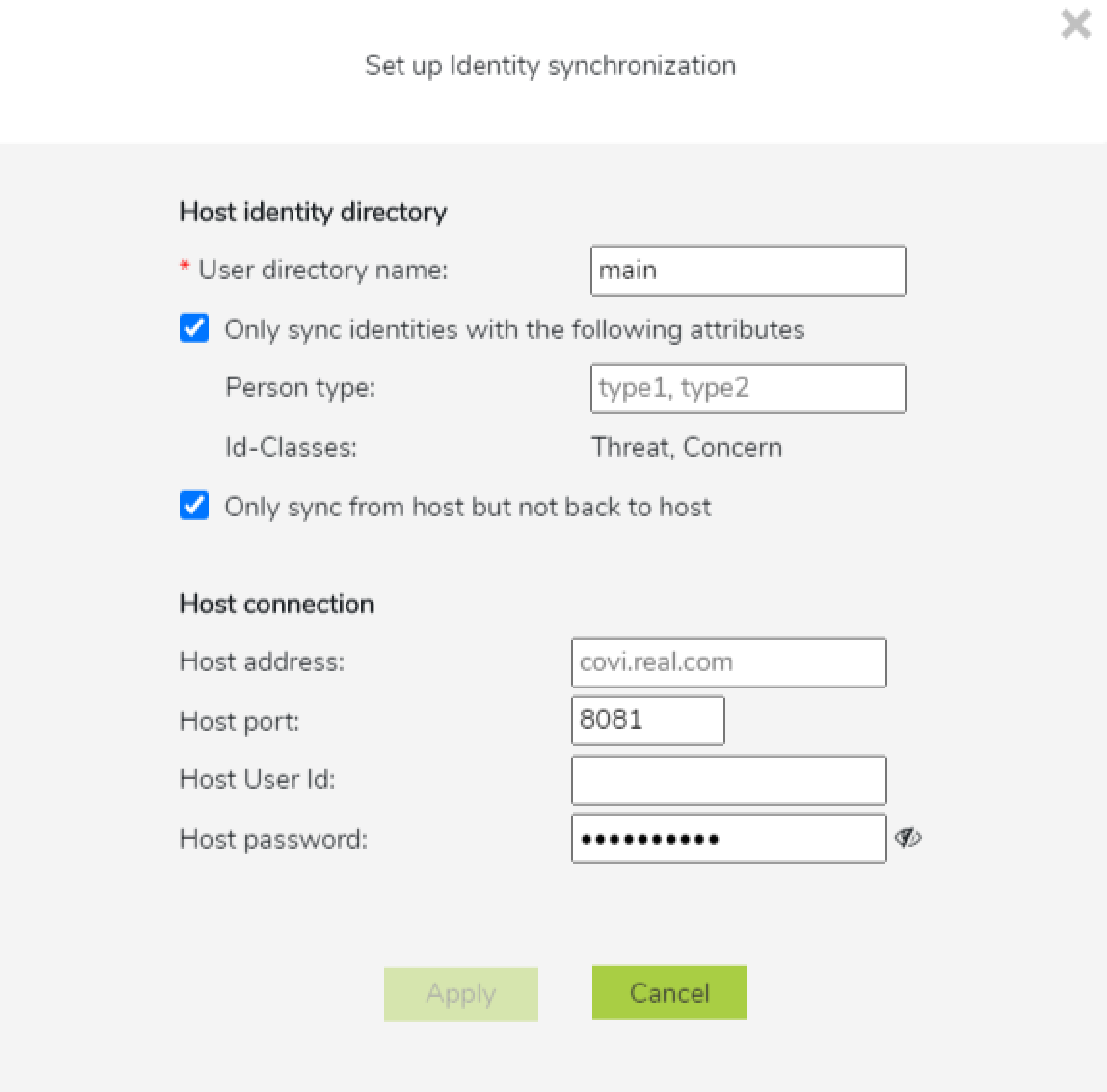
- **Target Directory**: Determines the directory whose identities are to be automatically removed.
- **Remove Anonymous Identity after**: Determines how many days to wait before removing identities that don't have a *name* attribute. Floating point numbers are valid. If this value is set to zero, then anonymous identities won't be automatically removed.
- **Remove Identities of person type**: Select the *Person Type* of the identities you'd like removed. If you don't modify this field, then identities of all *Person Types* will be removed.
- **after**: Determines how many days to wait before removing identities of the specified *Person Type*. Floating point numbers are valid. If this value is set to zero, then identities with *Person Types* won't be automatically removed.

82.3.5 Set up Identity synchronization

Enables the identity synchronization feature. This is only available over HTTPS, so an SSL certificate is required. See SSL Certificate Installation for more information.

When enabled and configured correctly, your Person Directory will sync with another Person Directory. The Person Directory that you're syncing with can belong to another SAFR system, or it can belong to a different user directory within your own SAFR system. See Identity Synchronization Configuration for more information about configuring identity synchronization, including more advanced directory configurations.

Selecting the *Set up Identity synchronization* box causes the following dialogue to appear:



Set up Identity synchronization

Host identity directory

* User directory name: main

☑ Only sync identities with the following attributes

Person type: type1, type2

Id-Classes: Threat, Concern

☑ Only sync from host but not back to host

Host connection

Host address: covi.real.com

Host port: 8081

Host User Id:

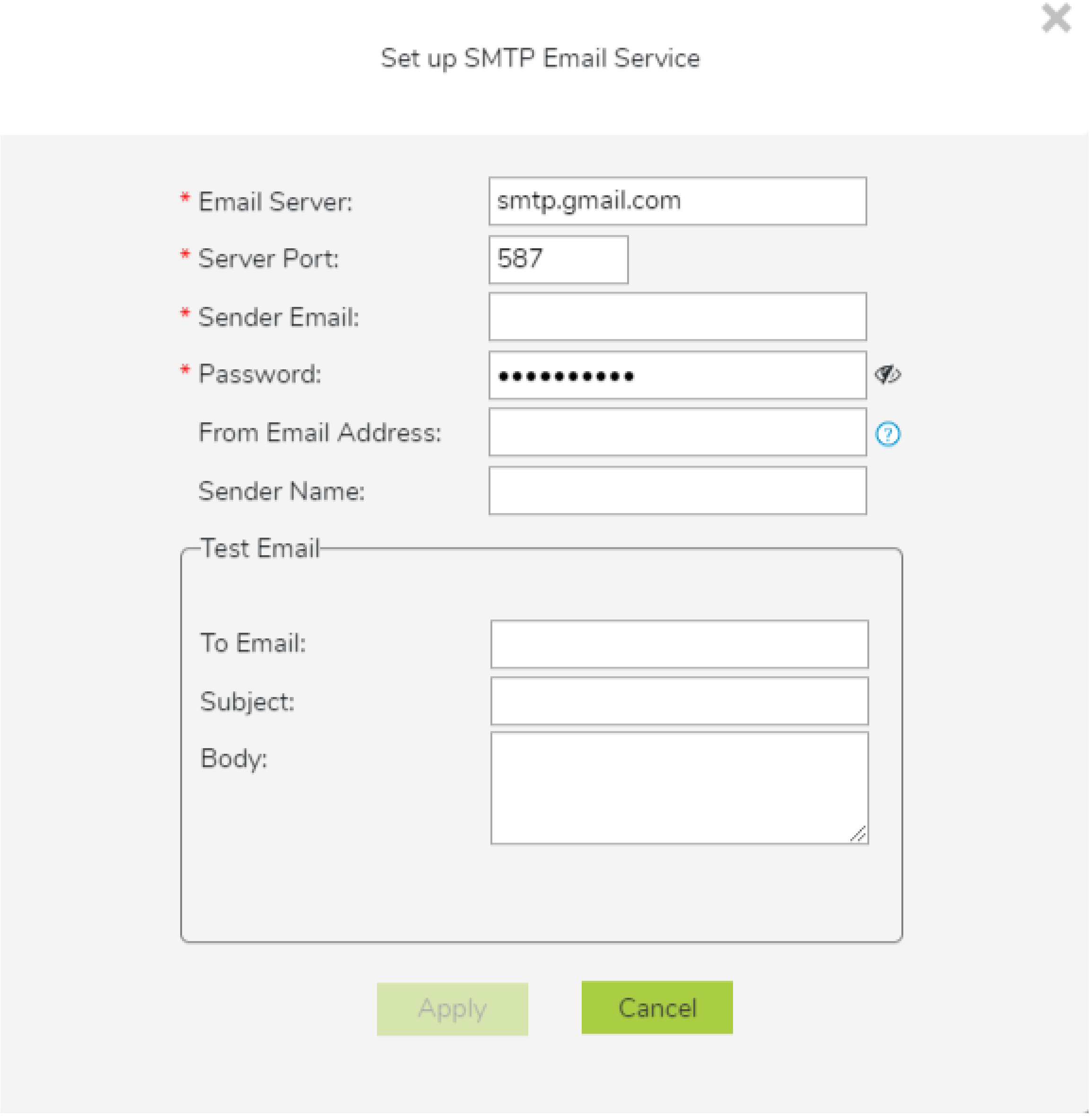
Host password: ●●●●●●●●●●●

- **User directory name**: The name of the user directory that you're trying to sync identities with.
- **Only sync identities with the following attributes**: When selected, it causes only identities with the specified attributes to be synced.
  - **Person type**: The *Person types* that identities must have to be synced.
  - **Id-Classes**: The *Id Classes* that identities must have to be synced.
- **Only sync from host but not back to host**: When enabled, the identity synchronization is uni-directional. When this is not enabled, then any identity within either Person Directory that isn't registered in the other Person Directory will be copied so that both directories will end up having identical sets of registered identities.
- **Host address**: The IP address or the hostname of the target host machine.
- **Host port**: The port number that the target machine's CoVi server listens on.
- **Host User Id**: The *User Id* of somebody who has the credentials to log into the host machine.
- **Host password**: The *password* of somebody who has the credentials to log into the host machine.

82.3.6 Set up SMTP Email Service

Enables SAFR's actions to send emails. Before you can configure SAFR to send emails, make sure you obtain an SMTP server account that you can use to send emails.

When you click on *Set up SMTP Email Service*, a dialogue will pop up requesting configuration information.



Set up SMTP Email Service

* Email Server: smtp.gmail.com

* Server Port: 587

* Sender Email:

* Password: ●●●●●●●●●●●

From Email Address:

Sender Name:

Test Email

To Email:

Subject:

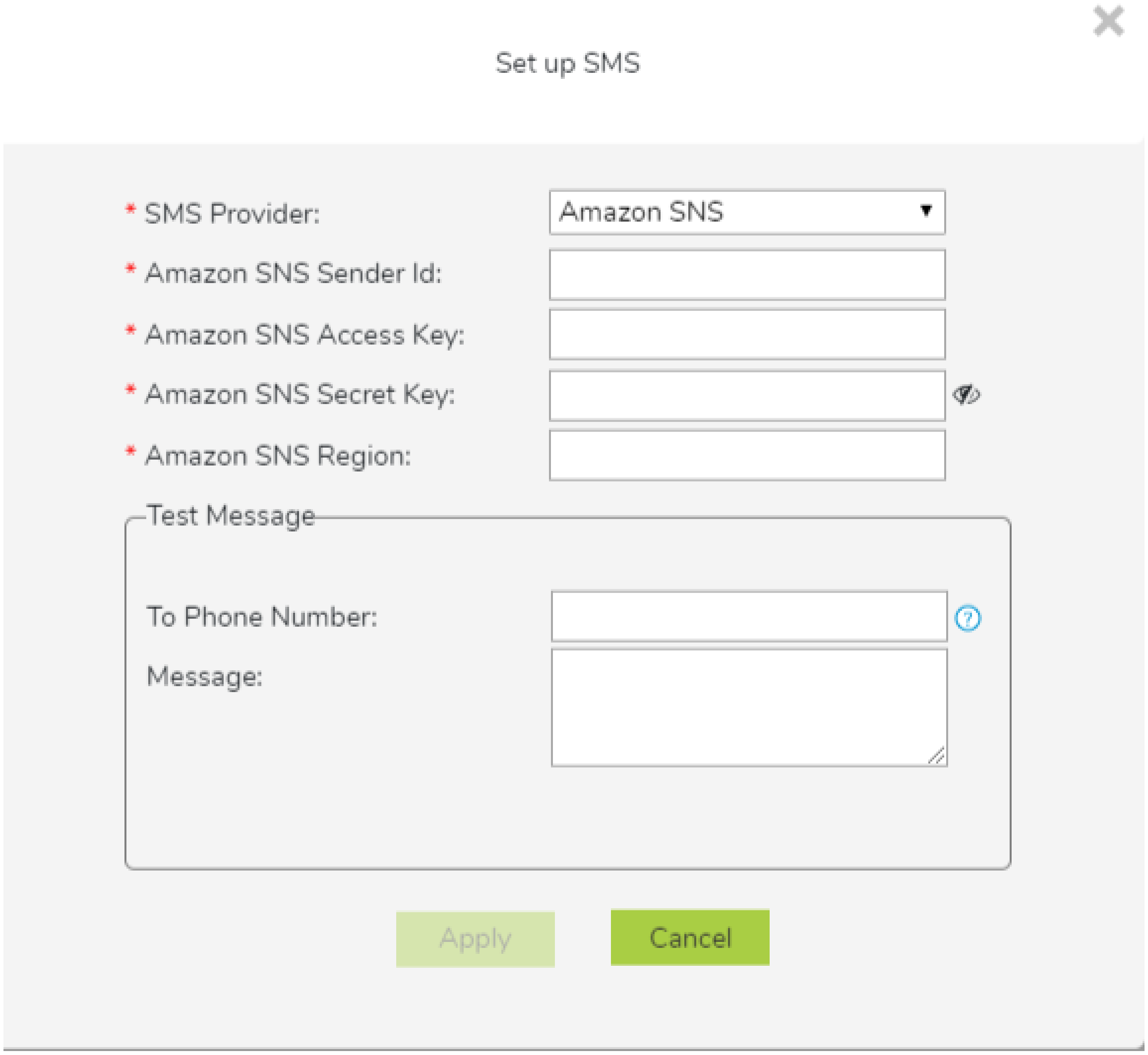
Body:

- **Email Server**: The address of the SMTP email server.
- **Server Port**: The email server port. The default port for SMTP is 587.
- **Sender Email**: The email username of the SMTP account. (e.g. me@gmail.com)
- **Password**: The password for the SMTP account.
- **From Email Address**: The email address that will appear on the "From" line. This feature isn't supported by all email servers; if this field isn't used then the *Sender Email* value is used for the "From" line.
- **Test Email**: Configure the test email that will be sent after you finish setting up the SMTP email service.
  - **To Email**: The email address to which the test email will be sent.
  - **Subject**: The test email's subject.
  - **Body**: The test email's body.

82.3.7 Set up SMS

Enables SAFR's actions to send short message service (SMS) messages. Before you can set up SMS, you must first set up an AWS account which is configured for your region so it can send SMS messages.

When you click on *Set up SMS*, a dialogue will pop up requesting configuration information.



Set up SMS

* SMS Provider: Amazon SNS

* Amazon SNS Sender Id:

* Amazon SNS Access Key:

* Amazon SNS Secret Key:

* Amazon SNS Region:

Test Message

To Phone Number:

Message:

- **SMS Provider**: The SMS provider that you're using. This value will always be `Amazon SNS`.
- **Amazon SNS Sender Id**: The name that will be used to send the SMS notifications.
- **Amazon SNS Access Key**: Your Amazon SNS Access Key.
- **Amazon SNS Secret Key**: Your Amazon SNS Secret Key.
- **Amazon SNS Region**: The region of your Amazon SNS.
- **Test Message**: Configure the test message that will be sent after you finish setting up SMS.
  - **To Phone Number**: The phone number to which the test message will be sent.
  - **Message**: The text message that will be sent to the phone number specified above.

82.4 License Information

Shows the operating limits of your SAFR license. See On-Premise Licensing for additional information about on-premise SAFR licenses, or Cloud Licensing for additional information on cloud SAFR licenses.

- **Expiration date**: The date when the SAFR license expires. After this date, SAFR software discontinues operation.

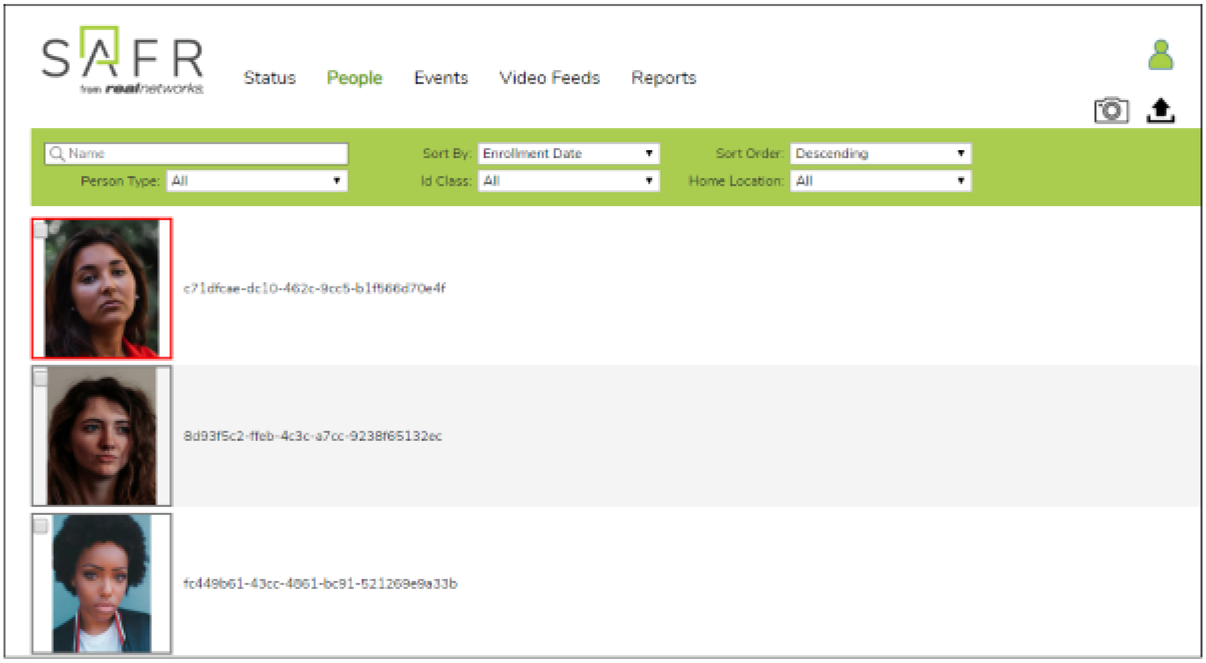
- **Max Feeds per Hour**: Maximum number of video feeds that can be used at one time by the SAFR system. If you attempt to connect more video feeds than your license allows, the excess video feed connection attempts will all fail. Existing video feeds must be disconnected for a period of 1 hour before new video feeds are allowed to re-use the license.

  Note: If a single camera is providing video feeds to 2 different Desktop Client instances, that counts as 2 video feeds for licensing purposes.
- **Max Faces**: Maximum number of faces that can be stored in SAFR's database. Attempting to save more faces than this limit allows results in an error.
- **Max Days Between Reports**: The maximum elapsed time that can pass before the SAFR system must report its status to a SAFR License Server. To communicate with the SAFR License Server, your SAFR Server must be able to make connections to cv-instam.real on port 443. Your SAFR Server will discontinue operation if it's unable to reach the SAFR License Server after the specified time has elapsed. If you need to operate your SAFR system on a private network that isn't connected to the Internet, contact your SAFR account manager to acquire a special offline license.

  Note: This metric is only applicable for on-premise deployments, and won't appear on the Web Consoles of cloud deployments.

83 People Page

The People page provides the ability to view and edit information about all the registered people in the Person Directory. For more information, see Manage People in the Person Directory.



Person Type: All

Sort By: Enrollment Date

Id Class: All

Sort Order: Descending

Home Location: All

c71dfcae-dc10-462c-9cc5-b1f666d70e4f

8d93f5c2-ffeb-4c3c-a7cc-9238f65132ec

fc449b61-43cc-4861-bc91-521269e9a33b

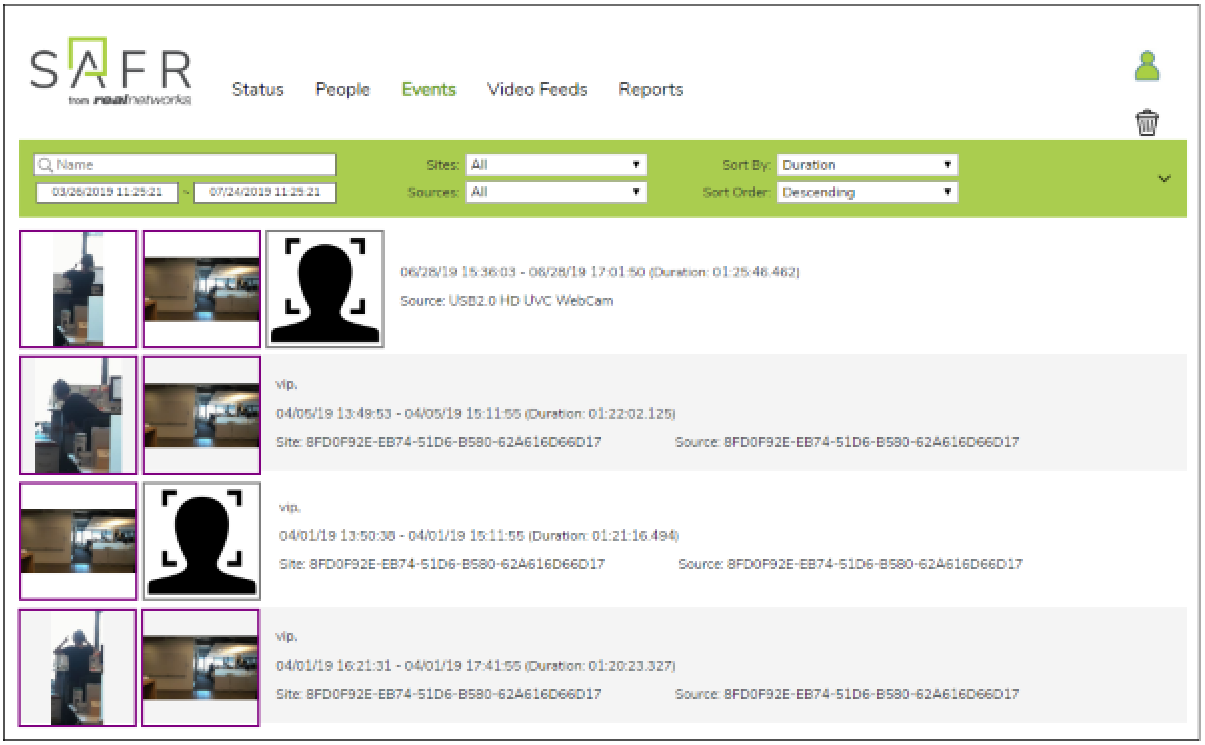
In addition, you can:

- Click the camera icon to take pictures of faces using your integrated camera to register people to the Person Directory.
- Click the upload icon to import images from files. Click the setting icon to adjust the acceptable lower limits of the center pose, contrast, and sharpness image quality metrics.

See Importing and Registering People for more information.

84 Events Page

The Events page lists all reported events stored in your Event Archive.



Name

03/26/2019 11:25:21 ~ 07/24/2019 11:25:21

Sites: All

Sources: All

Sort By: Duration

Sort Order: Descending

06/28/19 15:36:03 - 06/28/19 17:01:50 (Duration: 01:25:46.462)

Source: USB2.0 HD UVC WebCam

vip.

04/05/19 13:49:53 - 04/05/19 15:11:55 (Duration: 01:22:02.125)

Site: 8FD0F92E-EB74-51D6-B580-62A616D66D17 Source: 8FD0F92E-EB74-51D6-B580-62A616D66D17

vip.

04/01/19 13:50:38 - 04/01/19 15:11:55 (Duration: 01:21:16.494)

Site: 8FD0F92E-EB74-51D6-B580-62A616D66D17 Source: 8FD0F92E-EB74-51D6-B580-62A616D66D17

vip.

04/01/19 16:21:31 - 04/01/19 17:41:55 (Duration: 01:20:23.327)

Site: 8FD0F92E-EB74-51D6-B580-62A616D66D17 Source: 8FD0F92E-EB74-51D6-B580-62A616D66D17

85 Video Feeds Pages

The Video Feeds pages provide processor status and tenant configuration capabilities for all your connected video feeds. Root Config provides a list of all SAFR global default processor and feed properties.

The system is organized as follows:

- Tenants can have directories.
- Users and user IDs are security principals. They have privileges and map to a tenant. They have access to all directories within the tenant.
- If you have super privileges, you're also able to read, write, or config other tenants' properties for APIs that allow for those changes.
- A user ID can be restricted to particular directories within a tenant using white-listing.

Note: For cloud deployments, the Root Config properties are read-only. For local deployments, the Root Config property defaults can be changed by users with super config privileges. However, you are advised to make Root Config changes only when necessary.

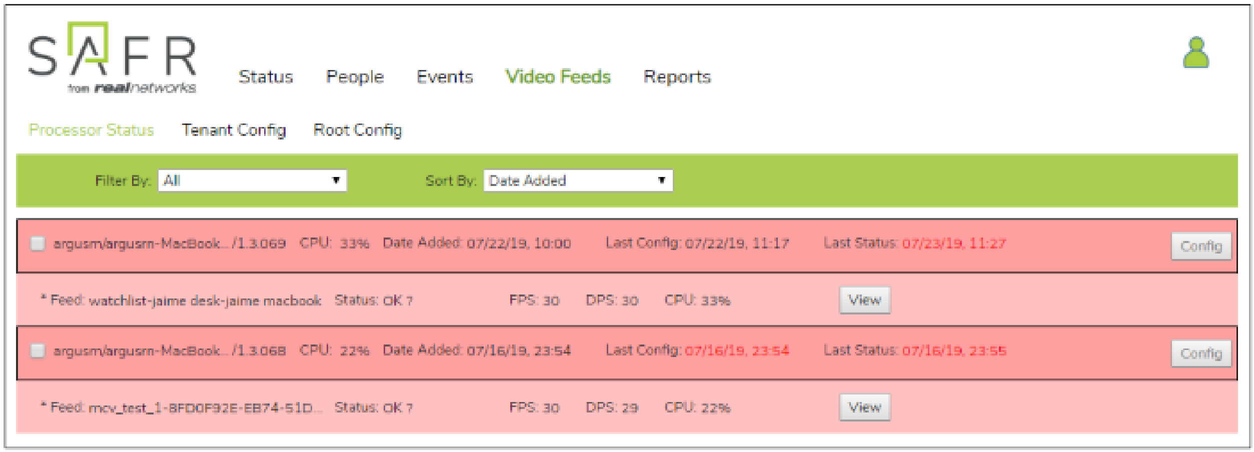
The Root and Tenant configs and modes are set on the tabs. Worker config is set by clicking the **Config** button on the **Processor Status** page.

- Tenant Config properties override Root Config properties or Feed properties for your account.
- Root mode overrides settings set on the Root and Tenant Config pages. Tenant mode overrides settings on other pages.
- Like the source URL, the Worker Config sets instance properties, although you can override any settings. This is useful to override settings for an individual device if, for example, there are unique lighting conditions for one feed.

85.1 Processor Status Page

This page provides a list of Desktop Client instances and video feeds associated with the account. Each row represents a separate computer running the Desktop Client that has a video feed associated with it. Inactive video feeds are identified by a red date-time status. Feeds are made inactive by either having status reporting disabled or shutting down the associated Desktop Client.

If the video feed is active, click **View** to access a streaming video window. Depending on your privileges, click **Config** to view, edit, or add attributes to override Root and Tenant global configuration settings for a single video feed. To make changes to global account settings, go to the Tenant Config page.



Filter By: All

Sort By: Date Added

□ argusm/argusm-MacBook... /1.3.069 CPU: 33% Date Added: 07/22/19, 10:00 Last Config: 07/22/19, 11:17 Last Status: 07/23/19, 11:27 Config

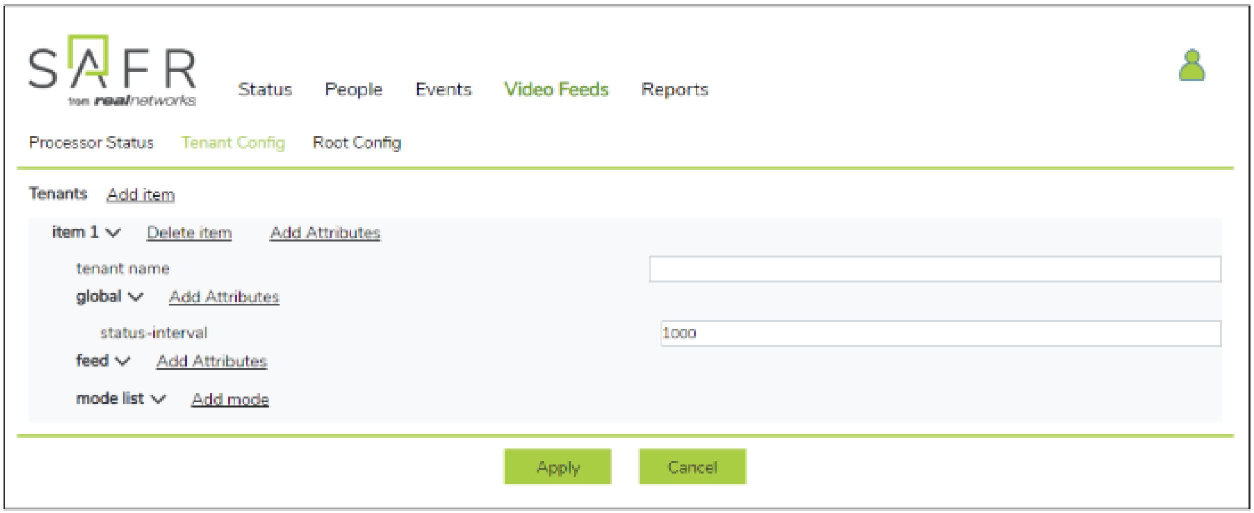
* Feed: watchlist-jaime desk-jaime macbook Status: OK ? FPS: 30 DPS: 30 CPU: 33% View

□ argusm/argusm-MacBook... /1.3.068 CPU: 22% Date Added: 07/16/19, 23:54 Last Config: 07/16/19, 23:54 Last Status: 07/16/19, 23:55 Config

* Feed: mcv_test_1-8FD0F92E-EB74-51D... Status: OK ? FPS: 30 DPS: 29 CPU: 22% View

85.2 Tenant Config Page

The Tenant is the primary account. Use this page to add and edit attributes of global settings at the account level to override the Root configurations. Directories can be added at this level by clicking the **Add Item** link. To make changes to individual video feeds, go to the Processor Status page.



Tenants

item 1

tenant name

global

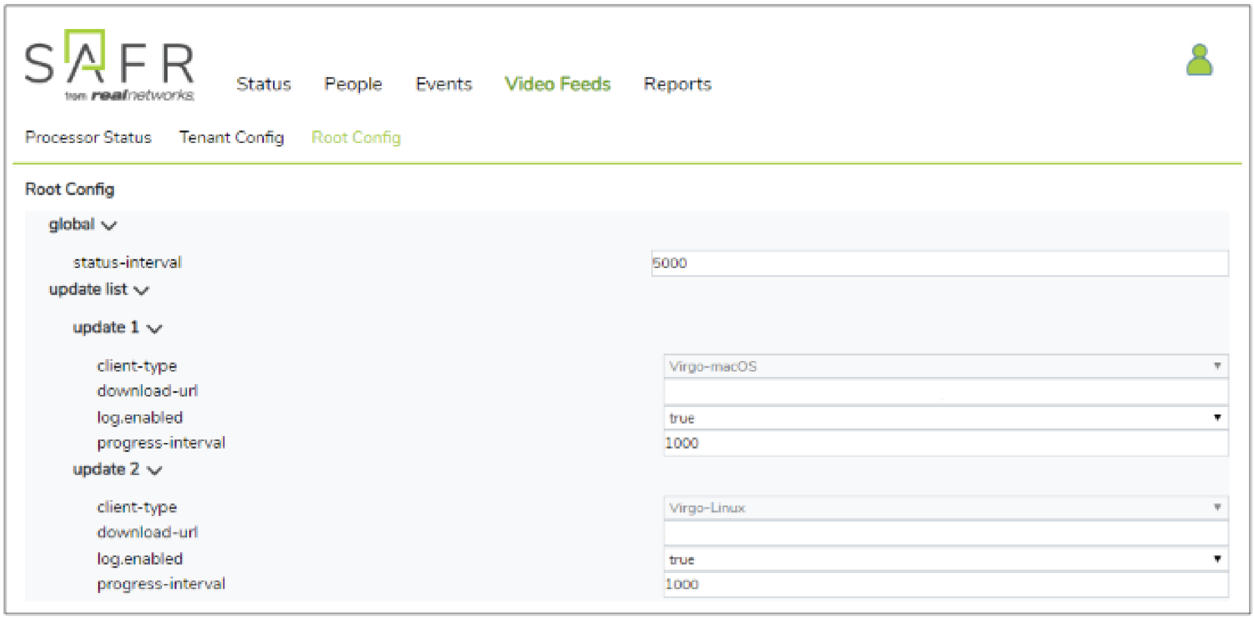
status-interval 1000

feed

mode list

85.3 Root Config Page

The Root Config page displays all the properties set in VIRGO by RealNetworks. These global settings are read-only for cloud deployments, but they can be changed for local deploymentts. To override these settings for your deployment, go to the Tenant Config and Processor Status pages.



Root Config

global

status-interval 5000

update list

update 1

client-type Virgo-macOS

download-url

log.enabled true

progress-interval 1000

update 2

client-type Virgo-Linux

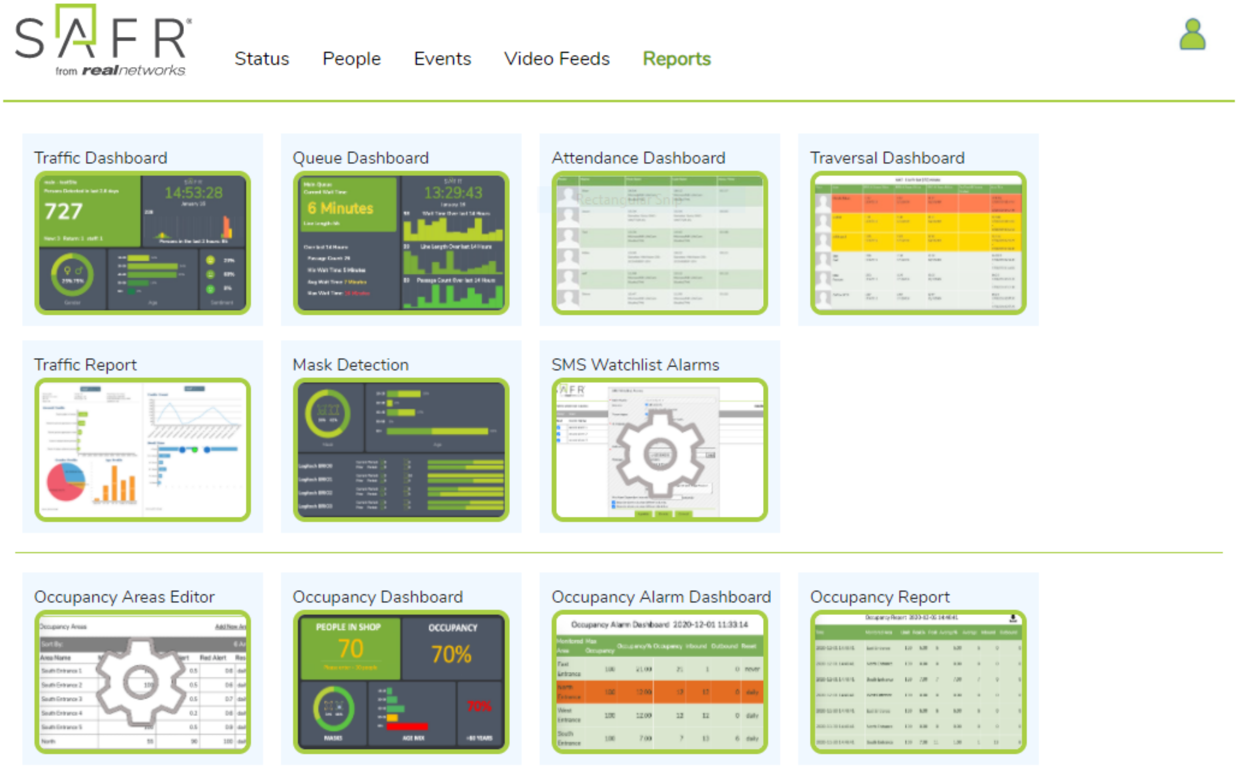
download-url

log.enabled true

progress-interval 1000

86 Reports Page

Click on the report that you're interested in to set the report's parameters and generate the report.



86.1 Save and Share Reports

The URLs of the generated reports contain all of the report's parameters, so you can save reports by bookmarking them and revisiting them at a later date.

Similarly, you can share reports with other people by emailing them reports' URLs. Note, however, that the link recipient will need to meet the following criteria to access the reports:

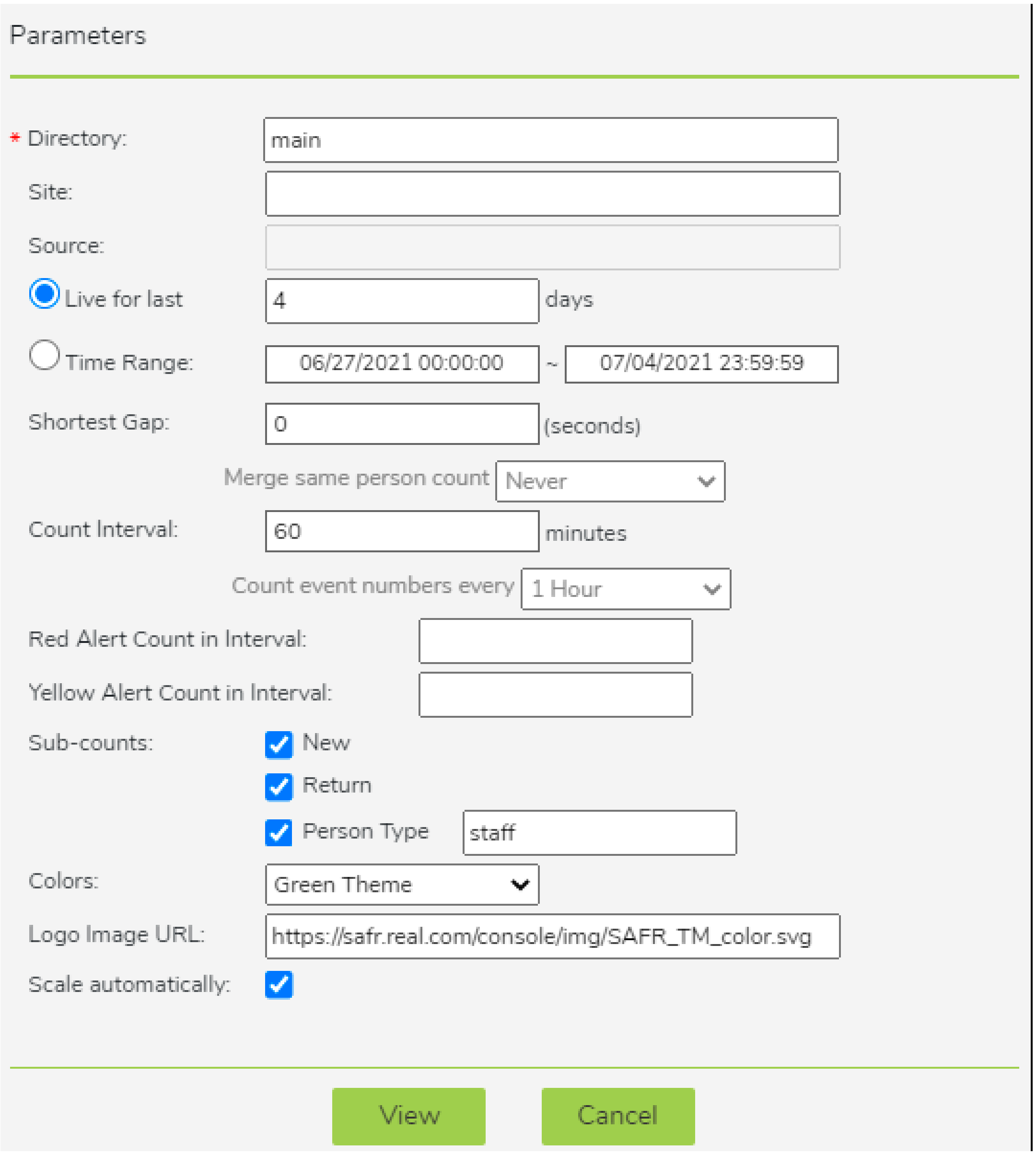
- They must have valid credentials for your SAFR system.
- They must have a user role other than Analyst. (i.e. Analysts are unable to view reports, but all other roles can view them.)

87 Traffic Dashboard

The Traffic Dashboard provides in-depth information about recognized and unrecognized people at your site, including:

- Total number of people viewed.
- Percentage of male and female faces.
- Age and sentiment percentages.
- Sentiment scores.

87.1 Input Parameters



Parameters

* Directory: main

Site:

Source:

● Live for last 4 days

○ Time Range: 06/27/2021 00:00:00 ~ 07/04/2021 23:59:59

Shortest Gap: 0 (seconds)

Merge same person count Never

Count Interval: 60 minutes

Count event numbers every 1 Hour

Red Alert Count in Interval:

Yellow Alert Count in Interval:

Sub-counts: ☑ New

☑ Return

☑ Person Type staff

Colors: Green Theme

Logo Image URL: https://safr.real.com/console/img/SAFR_TM_color.svg

Scale automatically: ☑

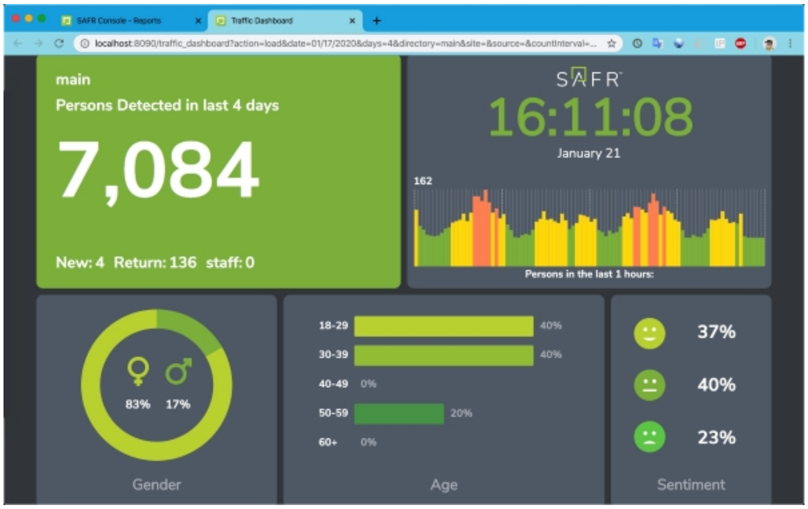
- **Directory**: User directory from which to run the dashboard.
- **Site**: Filter that allows you to limit the report to cameras with the specified site value. Site values can be set using the Account Preferences tab within the Desktop Client.
- **Source**: Filter that allows you to limit the report to a single source. A source is typically a camera

but may also be the source ID assigned when processing video from a file or making REST API calls. Source values can be set using the Camera Preferences tab within the Desktop Client.

- **Live for last**: Number of previous days to include in the dashboard. When this parameter is used, the Traffic Dashboard is dynamically re-generated every 30 seconds using the most recent time frame. For example, if you were to set this parameter to “2” and then leave the dashboard open for a week, it would always display data from the most recent two days. This parameter is mutually exclusive with *Time Range* below.
- **Time Range**: Dates to include in the dashboard. This parameter is mutually exclusive with *Life for last* above.
- **Shortest Gap**: If a person is viewed by a camera (thus triggering an event), leaves the field of view of the camera, and is then seen by the camera again (thus triggering another event), the two events will be merged into a single event if the time between them is equal to or less than the number of seconds specified by the *Shortest Gap* parameter.
  - **Merge same person count**: This is a field to help you calculate the *Shortest Gap* parameter. You can select a value from the drop-down menu, and the correct number of seconds will be calculated and entered into the *Shortest Gap* field.
- **Count Interval**: Defines the time interval included in each data bar of the trend chart.
  - **Count event numbers every**: This is a field to help you calculate the *Count Interval* parameter. You can select a value from the drop-down menu, and the correct number of minutes will be calculated and entered into the *Count Interval* field.
- **Red Alert Count in Interval**: When the count within a count interval is greater than this number, the trend chart bar is shown in red. Set this value to zero if you don’t want any bars shown in red.
- **Yellow Alert Count in Interval**: When the count within a count interval is greater than this number, the trend chart bar is shown in yellow. Set this value to zero if you don’t want any bars shown in yellow.
- **Sub-counts**: Specifies which sub-counts, if any, you want displayed on your dashboard. You can choose one or more of the following sub-counts:
  - **New**: Number of unique registered people that appear.
  - **Return**: Total number of registered people that appear. Note that multiple appearances by the same number are counted multiple times for the purpose of this sub-count.
  - **Person Type**: Number of people who appeared with the specified *Person Type*. Multiple *Person Types* can be specified by separating the *Person Types* with a comma. (e.g. “Staff,Teacher,Student”)
- **Colors**: Specifies which color scheme will be used for the dashboard. There are three options: *Blue Theme*, *Green Theme*, and *Contrast*.
- **Logo Image URL**: Use this to use a custom logo in place of the SAFR logo at the top of the trend chart.
- **Scale automatically**: When enabled, the dashboard will automatically fill the user’s browser window. This facilitates reading the dashboard even when the window is very small. (e.g. on mobile devices)

87.2 Generated Dashboard

Below is a sample Traffic Dashboard.


SAFR Console - Reports
Traffic Dashboard
localhost:8090/traffic_dashboard?action=load&date=01/17/2020&days=4&directory=main&site=&source=&countInterval=...
main
Persons Detected in last 4 days
7,084
New: 4 Return: 136 staff: 0
SAFR
16:11:08
January 21
162
Persons in the last 1 hours:
83%
17%
Gender
18-29
40%
30-39
40%
40-49
0%
50-59
20%
60+
0%
Age
37%
40%
23%
Sentiment

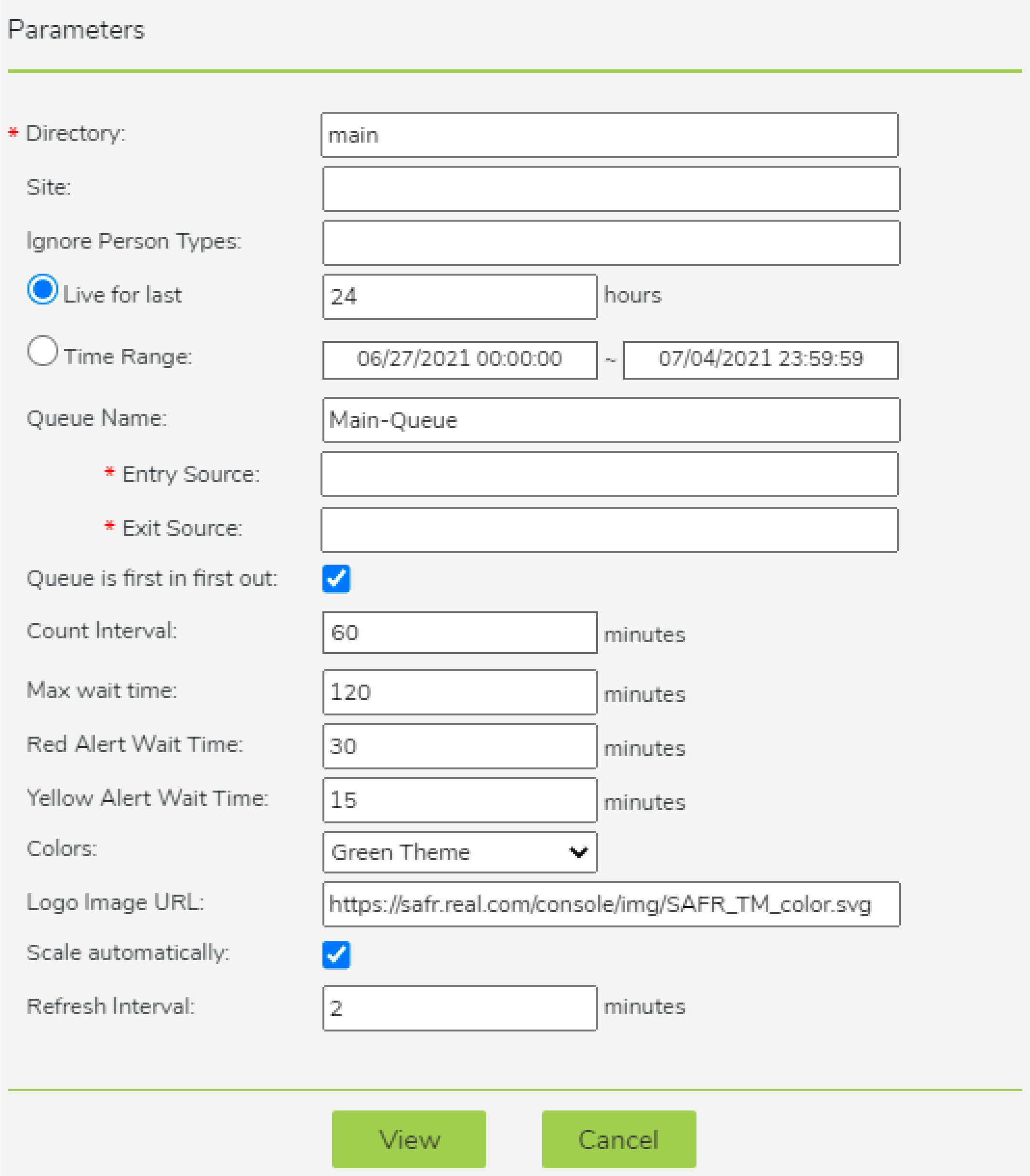
The trend chart is the chart in the upper right corner of the dashboard.

Note that the dashboard can have "Unknown" entries for both gender and age if some of your video feeds didn't have gender and/or age detection enabled during the time frame in question. Both gender and age detection can be enabled or disabled on the Recognition Preferences tab in the Desktop Client.

88 Queue Dashboard

The Queue Dashboard is used to monitor wait times in a queue. In order to use the Queue Dashboard you'll need 2 cameras: one for the entrance, and one for the exit.

88.1 Input Parameters



Parameters

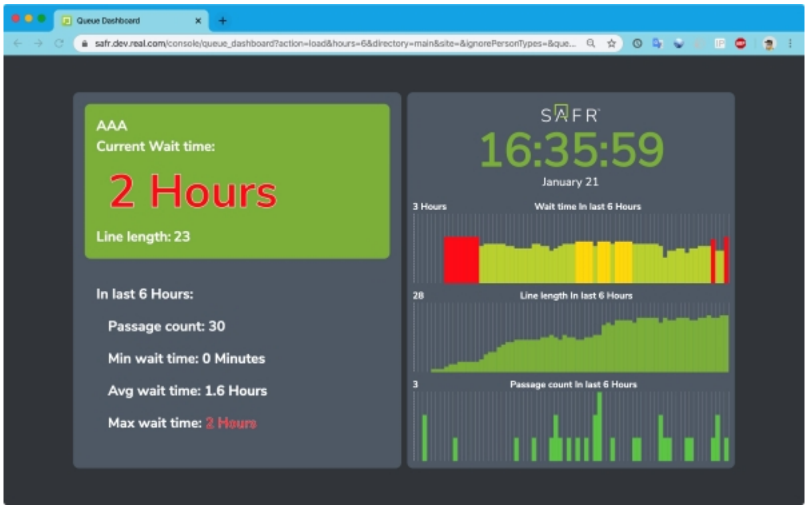
* Directory:	main	
Site:		
Ignore Person Types:		
● Live for last	24	hours
○ Time Range:	06/27/2021 00:00:00 ~ 07/04/2021 23:59:59	
Queue Name:	Main-Queue	
* Entry Source:		
* Exit Source:		
Queue is first in first out:	☑	
Count Interval:	60	minutes
Max wait time:	120	minutes
Red Alert Wait Time:	30	minutes
Yellow Alert Wait Time:	15	minutes
Colors:	Green Theme	
Logo Image URL:	https://safr.real.com/console/img/SAFR_TM_color.svg	
Scale automatically:	☑	
Refresh Interval:	2	minutes

- **Directory**: User directory from which to run the dashboard.
- **Site**: Specifies the camera(s) to use. Cameras' default site values can be set using the Account Preferences tab within the Desktop Client.
- **Ignore Person Types**: The Person Types that should not be included in the dashboard, if any.
- **Live for last**: Number of previous hours to include in the dashboard. Every time the dashboard refreshes, the most recent *Live for last* hours are used to re-generate the Queue Dashboard. The dashboard's refresh rate is defined by the *Refresh Interval* parameter below. This parameter is mutually exclusive with *Time Range* below.

- **Time Range**: Time range to include in the dashboard. This parameter is mutually exclusive with *Live for last* above.
- **Queue Name**: Title of the queue that appears at the top of the dashboard.
  - **Entry Source**: The camera at the beginning of the queue.
  - **Exit Source**: The camera at the exit of the queue.
- **Queue is first in first out**: When this is enabled, (i.e. FIFO behavior is enabled) when a person exits the queue everybody who entered the queue before that person is assumed to also be gone from the queue. When this is disabled, people who haven't been seen to leave the queue are assumed to still be waiting, until that person's **maxWaitTime** is exceeded.
- **Count Interval**: The amount of time each bar on the wait time chart in the Queue Dashboard represents.
- **Max wait time**: Any individual whose wait time exceeds this value is assumed to be a false data point and is discarded. It's assumed that the person left the queue without waiting within it to get to the end.
- **Red Alert Wait Time**: When the wait for somebody is greater than this number, the bar in the wait time chart is shown in red. Set this parameter to zero if you don't want any bars shown in red.
- **Yellow Alert Wait Time**: When the wait for somebody is greater than this number, the bar in the wait time chart is shown in yellow. Set this parameter to zero if you don't want any bars shown in yellow.
- **Colors**: Specifies which color scheme will be used for the dashboard. There are three options: *Blue Theme*, *Green Theme*, and *Contrast.*
- **Logo Image URL**: Use this to use a custom logo in place of the SAFR logo at the top of the wait time chart.
- **Scale automatically**: When enabled, the dashboard will automatically fill the user's browser window. This facilitates reading the dashboard even when the window is very small. (e.g. on mobile devices)
- **Refresh Interval**: Specifies how frequently the data on the dashboard is refreshed. If "0" is entered, the dashboard won't work. If you want a very quick refresh time, enter a very small non-zero number such as 0.1.

88.2 Generated Dashboard

Below is a sample Queue Dashboard.



safr.dev.real.com/console/queue_dashboard?action=load&hours=6&directory=main&site=&ignorePersonTypes=&que...

AAA
Current Wait time:
2 Hours
Line length: 23

In last 6 Hours:
Passage count: 30
Min wait time: 0 Minutes
Avg wait time: 1.6 Hours
Max wait time: 2 Hours

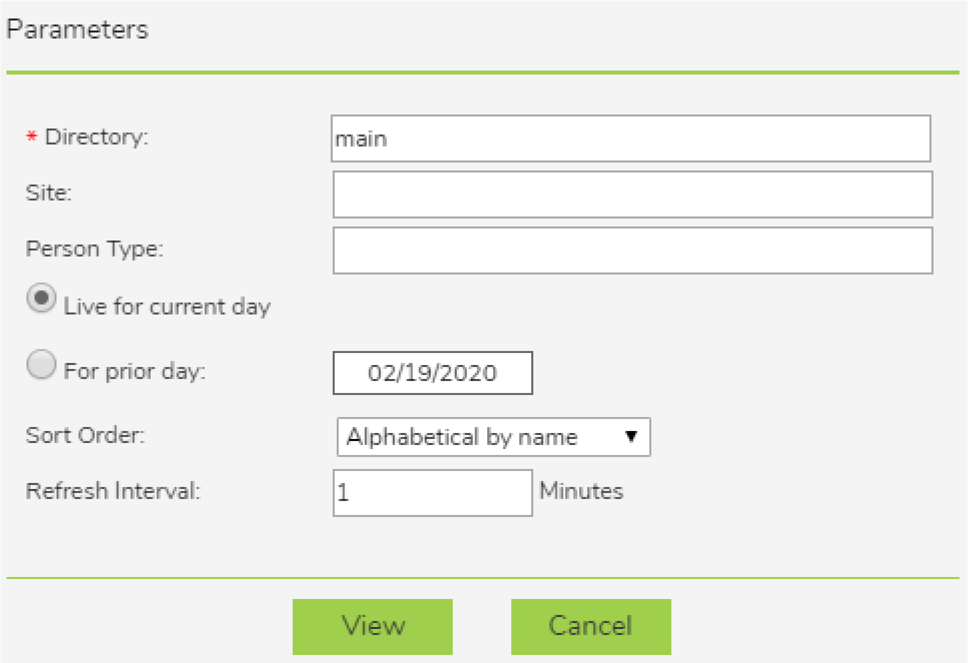
16:35:59
January 21

Queue Dashboard
SAFR
3 Hours
Wait time in last 6 Hours
28
Line length in last 6 Hours
3
Passage count in last 6 Hours

89 Attendance Dashboard

This report allows you to monitor the attendance record of a group of people (e.g. employees or students) on a given day. Although somebody might be seen multiple time in a day, this dashboard only reports the first time in a day they're seen and the last time in day they're seen, which allows the report to calculate how long a person was at the location. Note that this dashboard doesn't recognize periods in the middle of the day where the person might leave and then later come back to the location. (e.g. during lunch hour)

89.1 Input Parameters



Parameters

* Directory: main

Site:

Person Type:

● Live for current day

○ For prior day: 02/19/2020

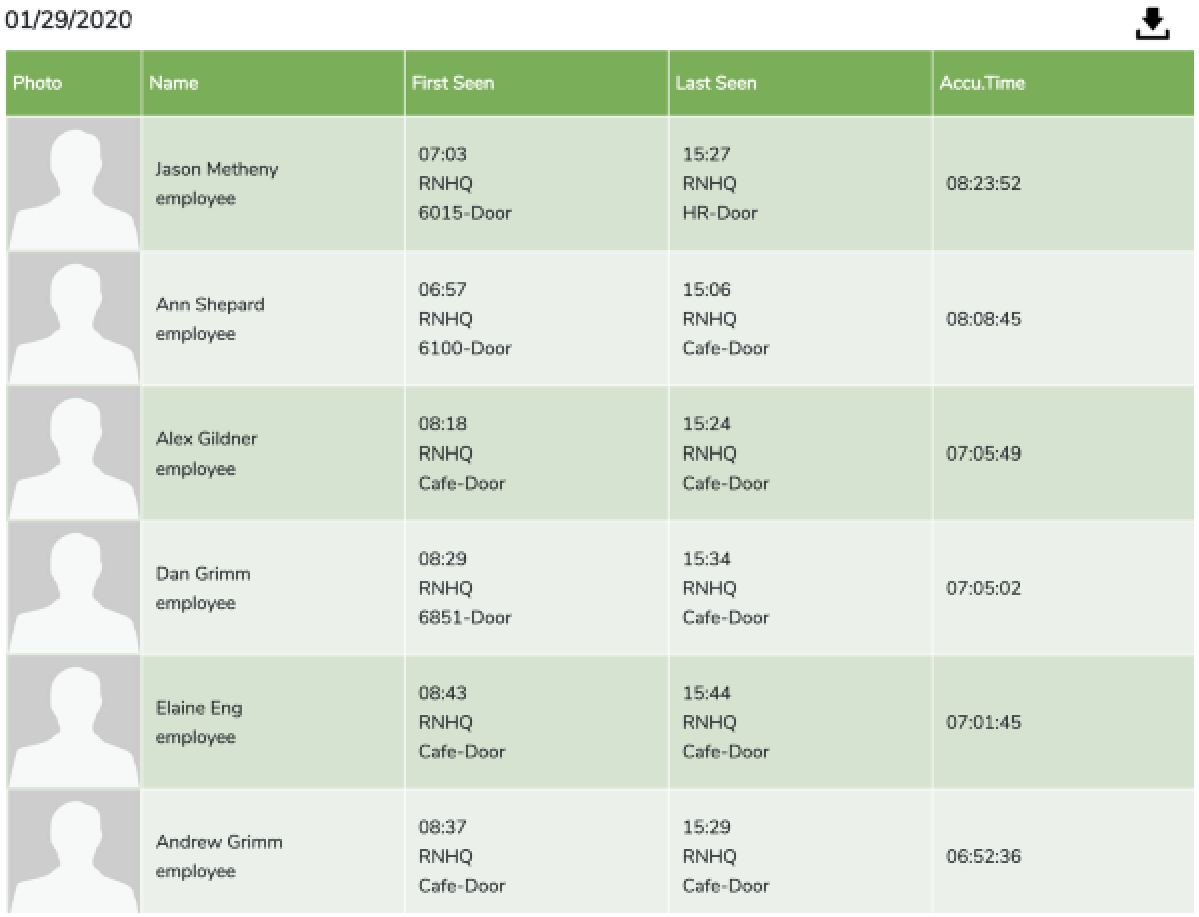
Sort Order: Alphabetical by name ▼

Refresh Interval: 1 Minutes

- **Directory**: User directory from which to run the dashboard.
- **Site**: Specifies the camera(s) to use. Cameras' site values can be set using the Account Preferences tab within the Desktop Client.
- **Person Type**: The Person Type(s) to be included in the dashboard. If this parameter is left blank, then all Person Types are included.
- **Live for current day**: Causes the current day to be used for the dashboard. Selecting this parameter is mutually exclusive with the *For prior day* parameter below.
- **For prior day**: The day which you want to appear in the dashboard. Selecting this parameter is mutually exclusive with the *Live for current day* parameter above.
- **Sort Order**: Specifies the criteria by which the people are sorted. There are 4 options:
  - **Alphabetical by name** - Sorts based on the alphabetical order of their names.
  - **In order of arrival** - Sorts based on the order of people's arrival times, with people who arrived first being displayed first.
  - **Shortest attendance first** - Sorts based on how long each person has attended, with the shortest attendances appearing first.
  - **Longest attendance first** - Sorts based on how long each person has attended, with the longest attendances appearing first.
- **Refresh Interval**: Specifies how frequently the data on the dashboard is refreshed. If "0" is entered, the dashboard won't work. If you want a very quick refresh time, enter a very small non-zero number such as 0.1.

89.2 Generated Dashboard

Below is a sample attendance dashboard. Note that you can download the dashboard as an *.xslx file by clicking on the download symbol in the upper right corner.



01/29/2020

Photo	Name	First Seen	Last Seen	Accu.Time
	Jason Metheny employee	07:03 RNHQ 6015-Door	15:27 RNHQ HR-Door	08:23:52
	Ann Shepard employee	06:57 RNHQ 6100-Door	15:06 RNHQ Cafe-Door	08:08:45
	Alex Gildner employee	08:18 RNHQ Cafe-Door	15:24 RNHQ Cafe-Door	07:05:49
	Dan Grimm employee	08:29 RNHQ 6851-Door	15:34 RNHQ Cafe-Door	07:05:02
	Elaine Eng employee	08:43 RNHQ Cafe-Door	15:44 RNHQ Cafe-Door	07:01:45
	Andrew Grimm employee	08:37 RNHQ Cafe-Door	15:29 RNHQ Cafe-Door	06:52:36

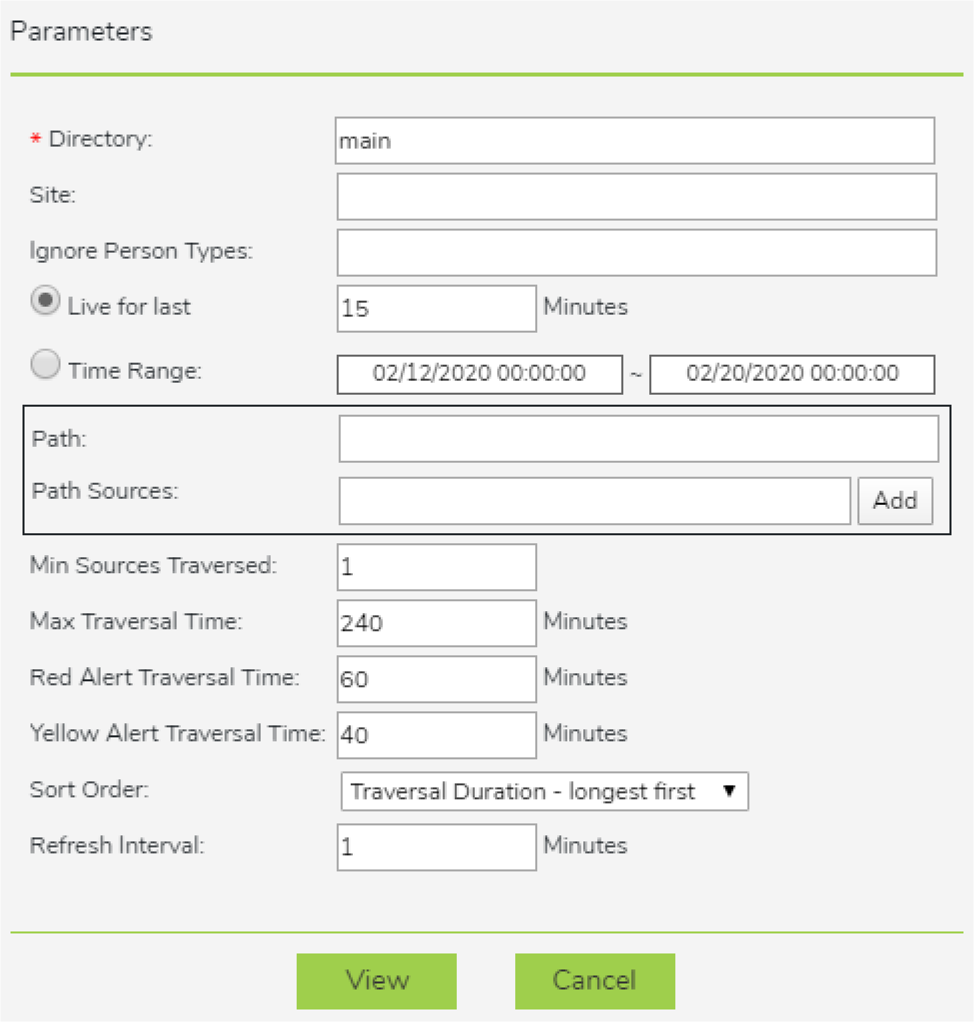
90 Traversal Dashboard

Displays traversal durations of individuals along a defined set of cameras. This dashboard highlights individuals exceeding expected traversal times and can be used to identify suspicious activity or general slow-downs (i.e. congestion) in real-time or time-frames in the past.

To use this report, you will either define a path when you input parameters or you can use an already defined path. A path is a list of 2 or more cameras. Thus, a path might consist of a set of cameras monitoring a causeway or a set of cameras monitoring all the entrances to a warehouse.

The dashboard requires that either SAFR is set to auto-register people or that viewed people are already registered in the database. Auto-registration is typically done by setting cameras to the `Learn and Monitor` video processing mode in the Camera Feed Analyzer window in the Desktop Client. In order for auto-registration to be sucessful, the facial images should be high quality and at least 220 pixels wide. This can be achieved by using high resolution cameras with sufficient zoom to capture faces.

90.1 Input Parameters



Parameters

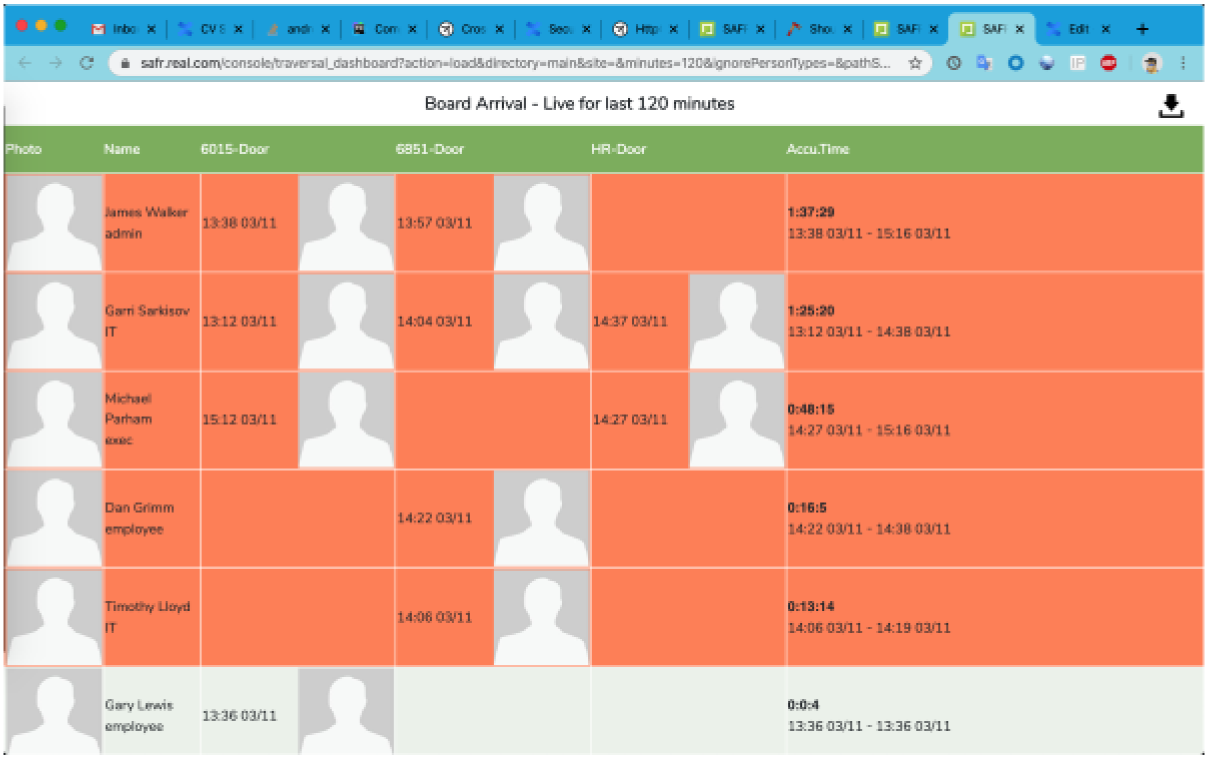
* Directory:	main	
Site:		
Ignore Person Types:		
● Live for last	15	Minutes
○ Time Range:	02/12/2020 00:00:00 ~ 02/20/2020 00:00:00	
Path:		
Path Sources:		Add
Min Sources Traversed:	1	
Max Traversal Time:	240	Minutes
Red Alert Traversal Time:	60	Minutes
Yellow Alert Traversal Time:	40	Minutes
Sort Order:	Traversal Duration - longest first ▾	
Refresh Interval:	1	Minutes

- **Directory**: User directory from which to run the dashboard.
- **Site**: Specifies the camera(s) to use. Cameras' site values can be set using the Account Preferences tab within the Desktop Client.
- **Ignore Person Types**: The Person Types the dashboard should ignore, if any.

- **Live for last**: Number of previous minutes to include in the dashboard. Every time the dashboard refreshes, the most recent *Live for last* hours are used to re-generate the Traversal Dashboard. The dashboard's refresh rate is defined by the *Refresh Interval* parameter below. This parameter is mutually exclusive with *Time Range* below.
- **Time Range**: Dates to include in the dashboard. This parameter is mutually exclusive with *Live for last* above.
- **Path**: The path that you want to use for this Traversal Dashboard. **Note**: If you have already defined one or more paths, then you have the option to use one of them by selecting an already defined path from a drop-down menu that this field will offer you.
- **Path Sources**: All the cameras that make up this traversal route. **Note**: The order you add cameras to this field doesn't matter.
- **Min Sources Traversed**: The minimum number of cameras that a person must pass in front of before the traversal dashboard will include them in its data.
- **Max Traversal Time**: Any individual whose traversal time exceeds this value is assumed to be a false data point and is discarded. It's assumed that the person left the traversal area without completing the path.
- **Red Alert Traversal Time**: When a person's traversal time exceeds this value, their data is shown in red on the Traversal Dashboard. Set this value to zero if you don't want any data shown in red.
- **Yellow Alert Traversal Time**: When a person's traversal time exceeds this value, their data is shown in yellow on the Traversal Dashboard. Set this value to zero if you don't want any data shown in yellow. The *Red Alert Traversal Time* parameter takes precedence over this parameter.
- **Sort Order**: Specifies the criteria by which the people are sorted. There are three values you can choose from:
  - Traversal Duration - Longest first.
  - Traversal Start - In order of arrival.
  - Traversal Start - Most recent first.
- **Refresh Interval**: Specifies how frequently the data on the dashboard is refreshed. If "0" is entered, the dashboard won't work. If you want a very quick refresh time, enter a very small non-zero number such as 0.1.

90.2 Generated Dashboard

Below is a sample Traversal Dashboard. Note that you can download the dashboard as an *.xslx file by clicking on the download symbol in the upper right corner.



safr.real.com/console/traversal_dashboard?action=load&directory=main&site=&minutes=120&ignorePersonTypes=&pathS...

Board Arrival - Live for last 120 minutes

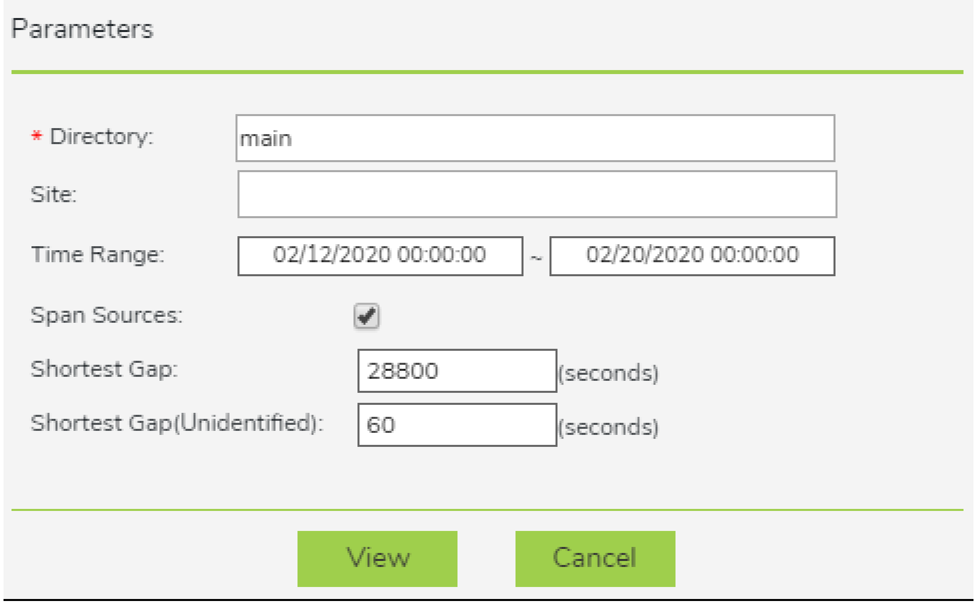
Photo	Name	6015-Door		6851-Door		HR-Door		Accu.Time
	James Walker admin	13:38 03/11		13:57 03/11				**1:37:29** 13:38 03/11 - 15:16 03/11
	Garri Sarkisov IT	13:12 03/11		14:04 03/11		14:37 03/11		**1:25:20** 13:12 03/11 - 14:38 03/11
	Michael Parham exec	15:12 03/11				14:27 03/11		**0:48:15** 14:27 03/11 - 15:16 03/11
	Dan Grimm employee			14:22 03/11				**0:16:5** 14:22 03/11 - 14:38 03/11
	Timothy Lloyd IT			14:06 03/11				**0:13:14** 14:06 03/11 - 14:19 03/11
	Gary Lewis employee	13:36 03/11						**0:0:4** 13:36 03/11 - 13:36 03/11

91 Traffic Report

Provides in-depth information about recognized and unrecognized people at your site, including:

- Total number of events.
- Counts for unknown and known persons.
- Gender and age profiles.
- Traffic trends per day.
- Dwell time: The amount of time a person remains on camera per event.

91.1 Input Parameters



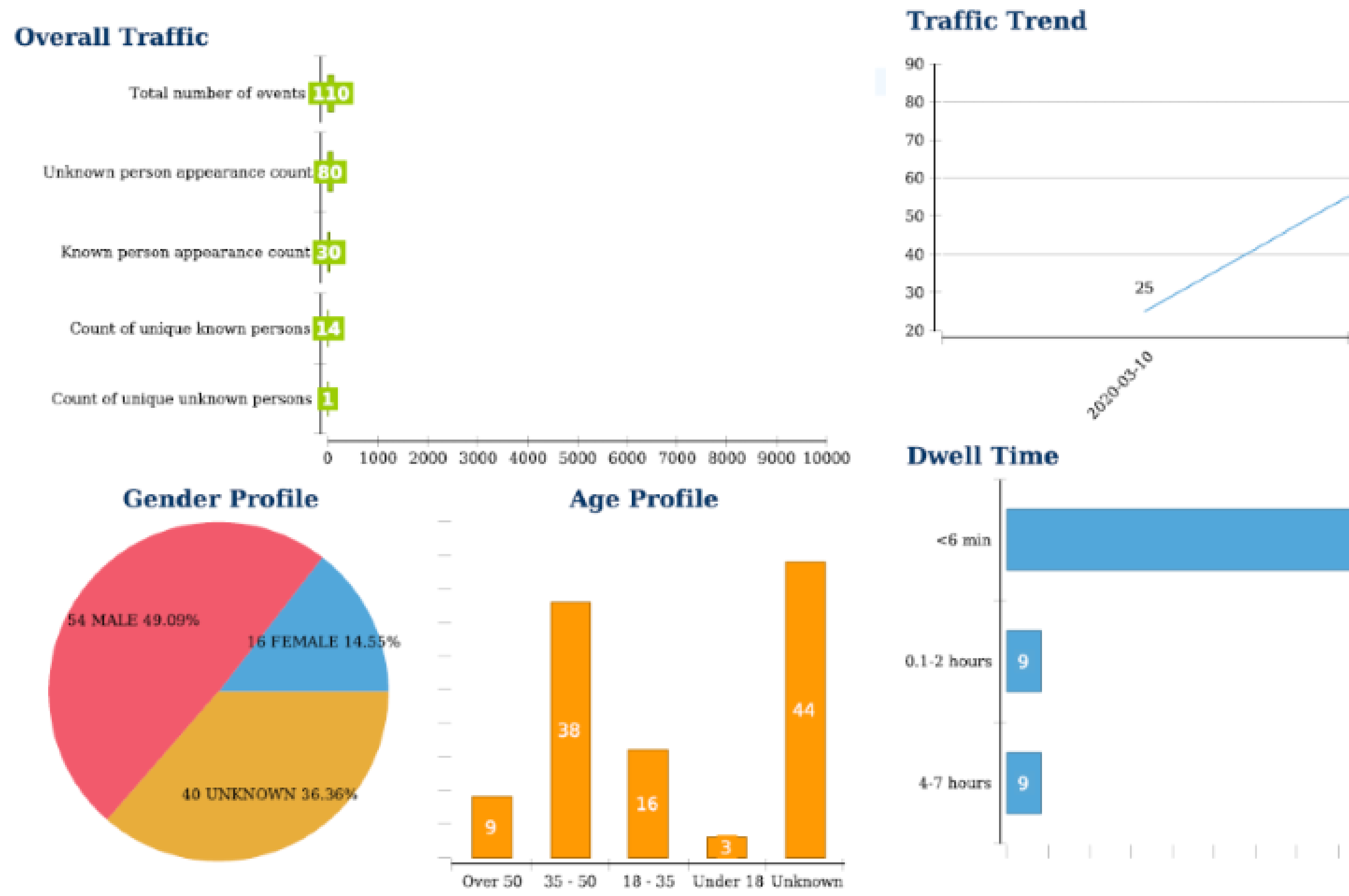
Parameters

* Directory:	main
Site:	
Time Range:	02/12/2020 00:00:00 ~ 02/20/2020 00:00:00
Span Sources:	☑
Shortest Gap:	28800 (seconds)
Shortest Gap(Unidentified):	60 (seconds)

- **Directory**: User directory from which to run the report.
- **Site**: Specifies the camera(s) to use. Cameras' site values can be set using the Account Preferences tab within the Desktop Client.
- **Time Range**: Dates and times to include in the report.
- **Span Sources**: Specifies whether or not events triggered in multiple cameras at the same time (plus or minus the shortest gap time) by the same person should be combined into a single event.
- **Shortest Gap**: If an identified person is viewed (thus triggering an event), leaves the field of view of the camera, and is then seen by the camera again (thus triggering another event) the two events will be merged into a single event if the time between them is equal to or less than the number of seconds specified by the *Shortest Gap* field.
- **Shortest Gap(Unidentified)**: If an unidentified person is viewed (thus triggering an event), leaves the field of view of the camera, and is then seen by the camera again (thus triggering another event) the two events will be merged into a single event if the time between them is equal to or less than the number of seconds specified by the *Shortest Gap(Unidentified)* field.

91.2 Generated Report

Below are screenshots from a sample Traffic Report:



The Overall Traffic graph exposes the following data:

- **Total number of events**: Total number of events generated over the time period covered by the report.
- **Unknown person appearance count**: Number of apppearances of registered people who don't have a name assigned to them in the Identity Database.
- **Known person appearance count**: The number of apppearances of named registered people.
- **Count of unique known persons**: Number of named registered people who were seen. Note that if a person was seen multiple times, they're only counted once for the purpose of this value.
- **Count of unique unknown persons**: Number of registered people who don't have a name assigned to them in the Identity Database who were seen. Note that if a person was seen multiple times, they're only counted once for the purpose of this value.

Both the gender and age profiles can have "Unknown" entries if some of your video feeds didn't have gender and/or age detection enabled during the time frame covered by the report. Both gender and age detection can be enabled or disabled on the Recognition Preferences tab in the Desktop Client.

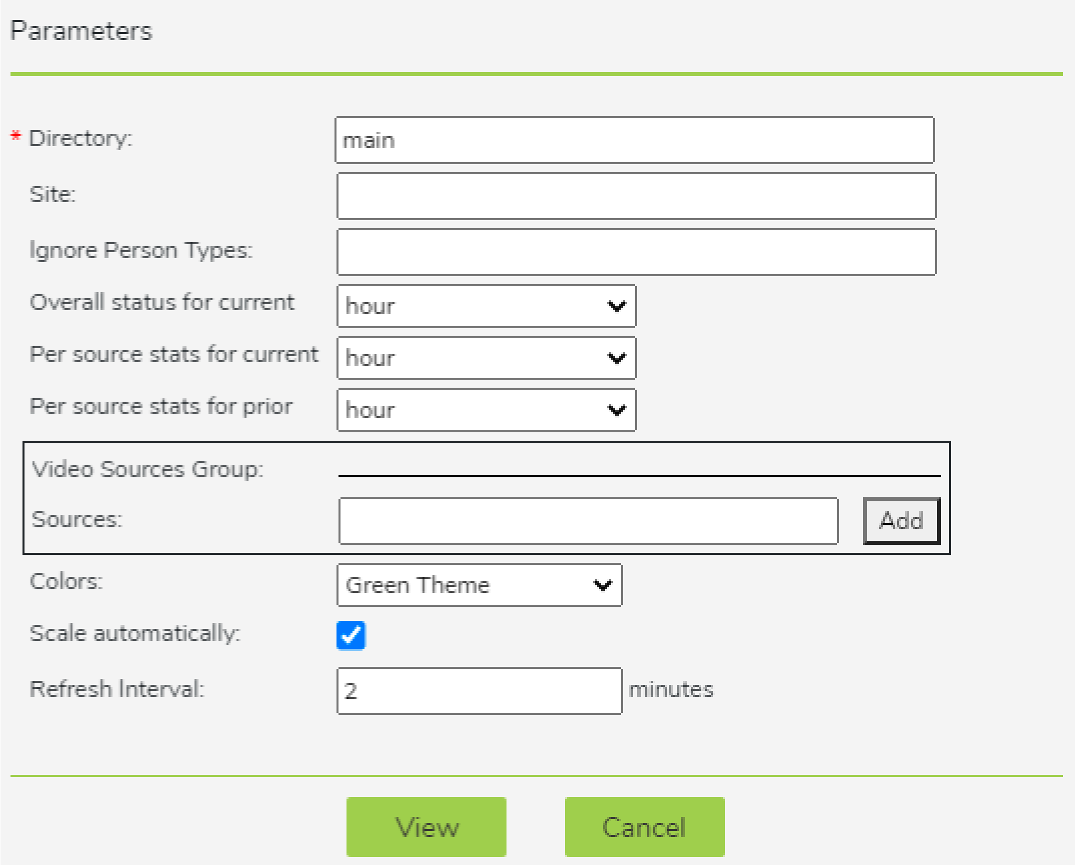
Dwell time is the amount of time a person remains on camera per event.

You can download the sample Traffic Report here.

92 Mask Detection Dashboard

The Mask Detection Dashboard is used to monitor the percentage of events that are mask events.

92.1 Input Parameters



Parameters

* Directory: main

Site:

Ignore Person Types:

Overall status for current: hour

Per source stats for current: hour

Per source stats for prior: hour

Video Sources Group:

Sources:

Colors: Green Theme

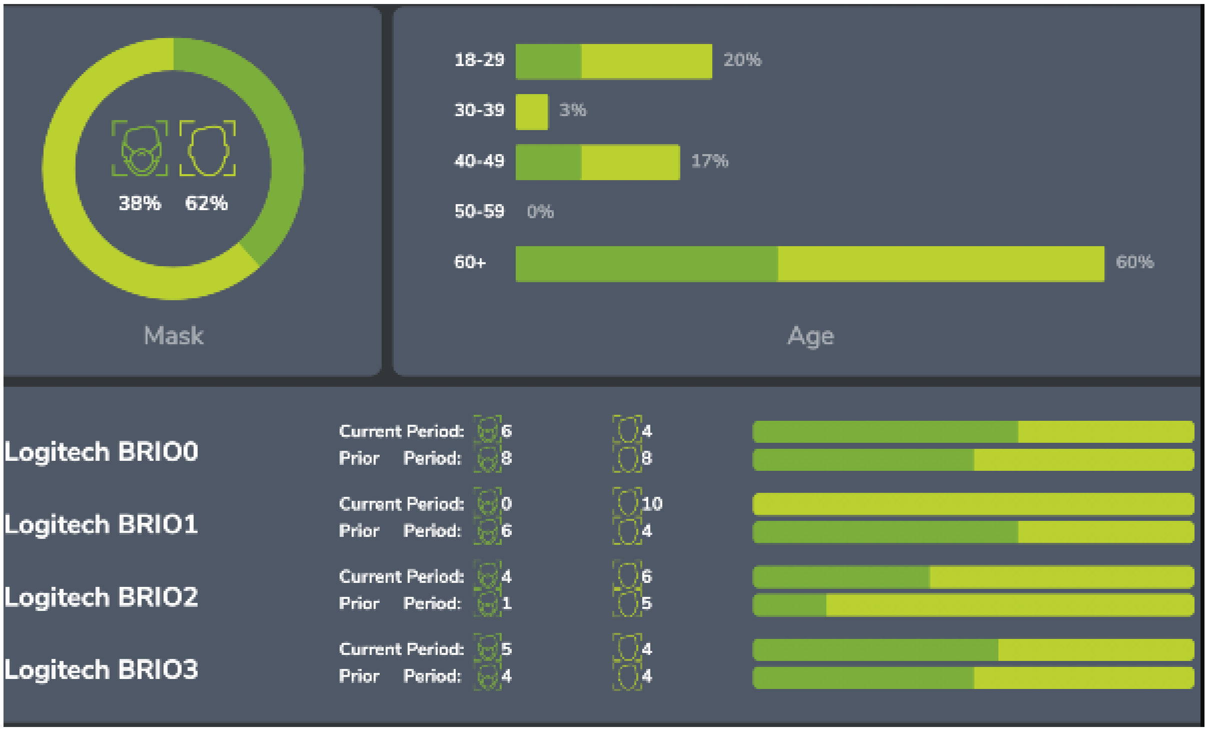
Scale automatically: ☑

Refresh Interval: 2 minutes

- **Directory**: User directory from which to run the dashboard.
- **Site**: Filter that allows you to limit the report to cameras with the specified site value. Site values can be set using the Account Preferences tab within the Desktop Client.
- **Ignore Person Types**: Specifies the Person types, if any, that the Mask Detection Dashboard should ignore and not count.
- **Overall status for current**: Specifies the amount of time that will be monitored for mask events in the upper left corner of the dashboard. You may select from: *hour*, *day*, *week*, *month*, or *quarter*.
- **Per source stats for current**: Specifies the amount of time that will be monitored as the "Current Period" for each video source being monitored. You may select from: *hour*, *day*, *week*, *month*, or *quarter*.
- **Per source stats for prior**: Specifies the amount of time that will be monitored as the "Prior Period" for each video source being monitored. You may select from: *hour*, *day*, *week*, *month*, or *quarter*.
- **Video Sources Group**
  - **Sources**: The cameras that you want to include in the dashboard.
- **Colors**: Specifies which color scheme will be used for the dashboard. There are three options: *Green Theme*, *Blue Theme*, and *Contrast*.
- **Scale automatically**: When enabled, the dashboard will automatically fill the user's browser window. This facilitates reading the dashboard even when the window is very small. (e.g. on mobile devices)
- **Refresh Interval**: Specifies how frequently the data on the dashboard is refreshed. If "0" is entered, the dashboard won't work. If you want a very quick refresh time, enter a very small non-zero number such as 0.1.

92.2 Generated Dashboard

Below is a sample Mask Detection Dashboard.



93 SMS Watchlist Alarms Dashboard

The SMS Watchlist Alarms Dashboard allows you to create and manage one or more alarms that, when triggered, will send a Short Message Service (SMS) message to one or more specified recipients.

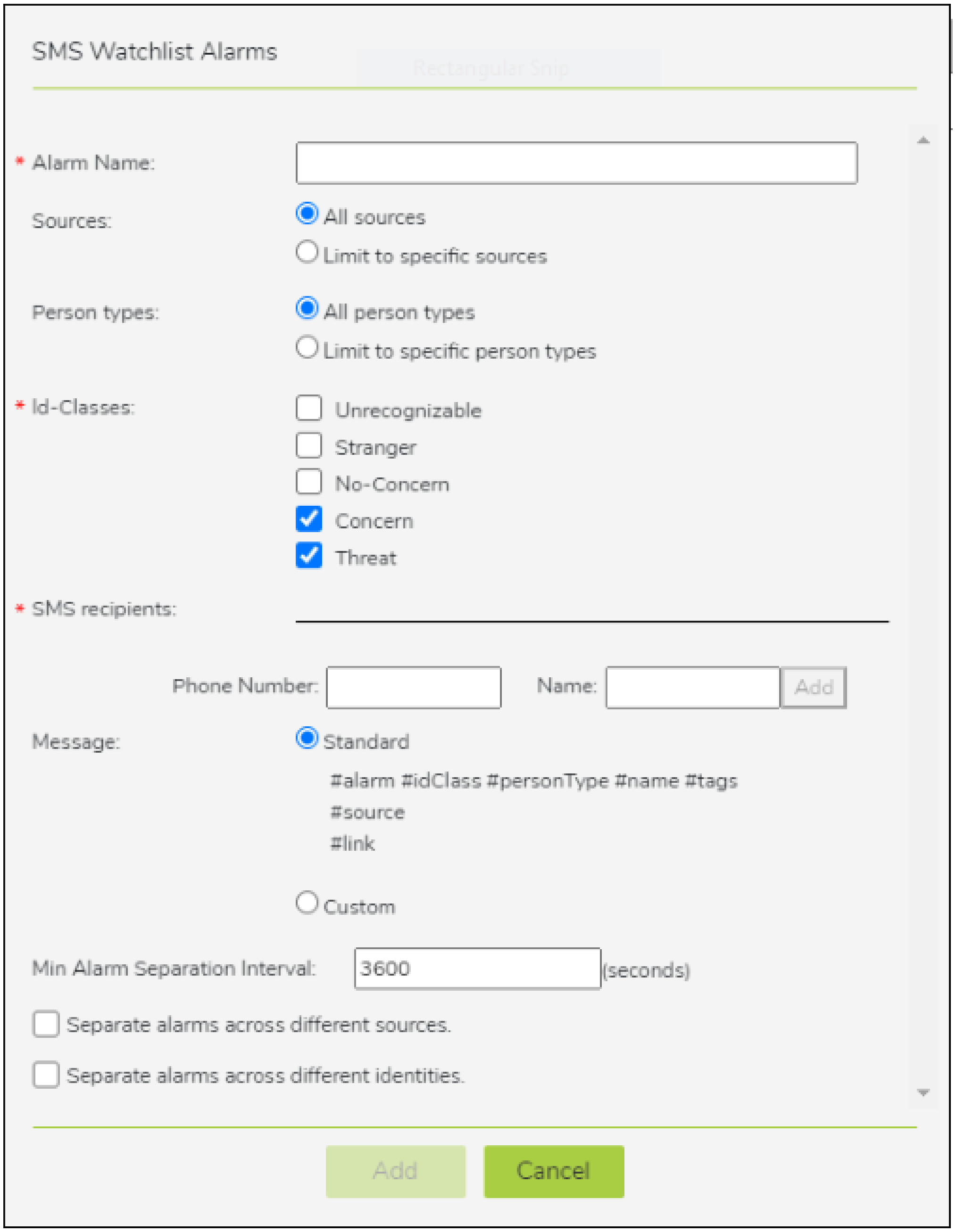
Before this dashboard can be used, you must first configure SAFR to send SMS messages. This is done on the Web Console's Status Page.

93.1 The Dashboard

When you click on the SMS Watchlist Alarms Dashboard on the Web Console's Reports Page, you're immediately shown the dashboard. The dashboard consists of a list of all the SMS Watchlist Alarms that you've defined. You're able to individually enable or disable the listed alarms. Clicking on a listed alarm will allow you to edit its definition. To define an entirely new alarm, click on the **Add New Alarm** button.

93.2 SMS Watchlist Alarm Definition

When you create a new alarm or edit an existing alarm, you'll use the alarm definition dialogue shown below to configure the alarm.



SMS Watchlist Alarms

* Alarm Name:

Sources: ● All sources ○ Limit to specific sources

Person types: ● All person types ○ Limit to specific person types

* Id-Classes:
□ Unrecognizable
□ Stranger
□ No-Concern
☑ Concern
☑ Threat

* SMS recipients:

Phone Number: Name:

Message: ● Standard

#alarm #idClass #personType #name #tags
#source
#link

○ Custom

Min Alarm Separation Interval: 3600 (seconds)

□ Separate alarms across different sources.

□ Separate alarms across different identities.

- **Alarm Name**: Name of this alarm. The name must be unique across all SMS Watchlist Alarms in the specified directory.
- **Sources**: Specifies which sources (i.e. which cameras) can be used to trigger this alarm.
- **Person types**: The *Person types* that can trigger this alarm. Note that in order to trigger this alarm, a person must also be one of the *Id Classes* specified below.
- **Id Classes**: The *Id Classes* that will trigger this alarm. Note that at least one *Id Class* must be selected.
- **SMS recipients**: The list of people (defined by their name and phone number) that will receive an SMS message if this alarm is triggered.
- **Message**: The text of the SMS message. You can select a standard message, or you can create a

custom message of your own. Note that the SMS messages support macros, as shown in the table below. (Indeed, the standard message consists entirely of macros.)

Macro	Description
#alarm	Will be replaced with the name of the alarm.
#source	Will be replaced with the event.source. If no event.source is set for the event, this macro will instead be replaced with an empty string.
#site	Will be replaced with the event.site. If no event.site is set for the event, this macro will instead be replaced with an empty string.
#idClass	Will be replaced with the event.idClass.
#personType	Will be replaced with the event.personType. If no event.personType is set for the event, this macro will instead be replaced with an empty string.
#name	Will be replaced with the event.name. If no event.name is set for the event, this macro will instead be replaced with an empty string.
#tags	Will be replaced with the event.tags. If no event.tags is set for the event, this macro will instead be replaced with an empty string.
#link	Will be replaced with URL to the event that triggered the alarm. When using an on-premise license, this URL will take the form of: https://<server>/e/<eventGUID> When using a cloud license, this URL will take the form of: https://safr.real.com/console/e/<eventId>

- **Min Alarm Separation Interval**: The amount of time, in seconds, that must pass before this alarm can be triggered again.
    - **Separate alarms across different sources**: When enabled, alarms triggered by different sources can ignore the *Min Alarm Separation Interval* value. (i.e. Multiple alarms can be triggered at the same time, provided each alarm is being triggered by an event viewed by different sources.)
    - **Separate alarms across different identities**: When enabled, alarms triggered by different people can ignore the *Min Alarm Separation Interval* value. (i.e. Multiple alarms can be triggered at the same time, provided each alarm is being triggered by a different person.)

94 Occupancy Areas Editor

The Occupancy Areas Editor is used to manage occupancy monitoring areas. An occupancy monitoring area represents a physical space where SAFR is monitoring how many people are in it.

Occupancy monitoring areas rely on SAFR's *direction of travel* recognition feature, so the camera(s) that the occupancy monitoring areas use should be positioned such that people are either moving directly away, directly towards, directly to the left, or directly to the right of the camera. See the Desktop Client Camera Preferences documentation for additional information about the *direction of travel* feature.

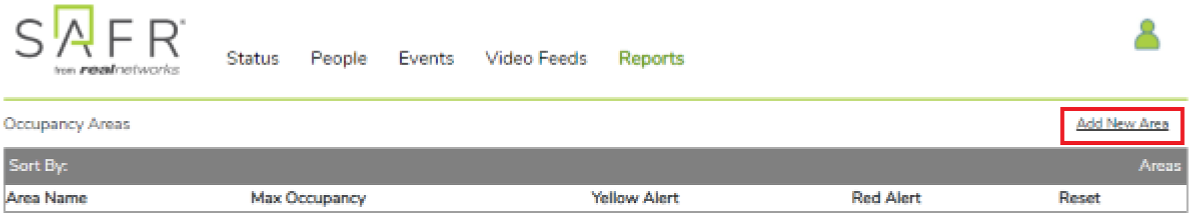
In addition, the following preferences must be set for the occupancy monitoring areas to work:

- Every participating camera must be using a Video Processing Mode that can generate events. Specifically, each camera must be using one of the following Video Processing Modes:
  - Learn and Monitor
  - Anonymous Traffic Monitoring
  - Enrolled and Anonymous Traffic Monitoring
  - Enrolled and Unique Traffic Monitoring
  - Enrolled and Stranger Monitoring
- Events must be enabled in the Desktop Client's Events Preferences menu.
- Face detection and person detection must be enabled in the Desktop Client's Detection Preferences menu. If person detection isn't enabled, then anybody whose face can't be seen (e.g. only the back of their head is visible) won't be counted.

Occupancy Reports and Occupancy Alarm Dashboard use occupancy monitoring areas to generate their output, so you must define at least one area to use them.

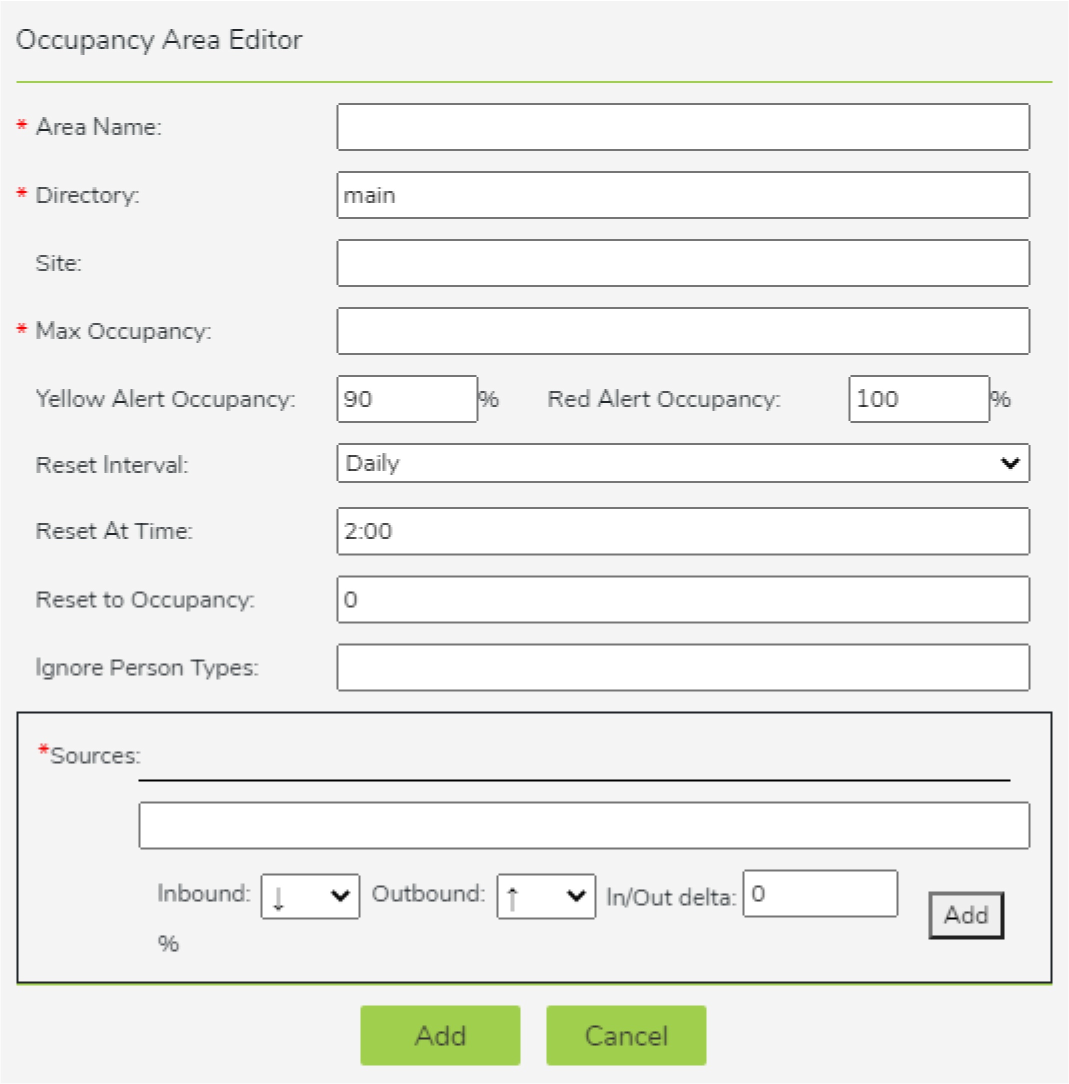
94.1 Define New Occupancy Monitoring Areas

To define a new occupancy monitoring area, click on **Add New Area** in the upper right corner:



Area Name	Max Occupancy	Yellow Alert	Red Alert	Reset

You will be prompted to enter the following information:



Occupancy Area Editor

* Area Name:

* Directory: main

Site:

* Max Occupancy:

Yellow Alert Occupancy: 90 % Red Alert Occupancy: 100 %

Reset Interval: Daily

Reset At Time: 2:00

Reset to Occupancy: 0

Ignore Person Types:

*Sources:

Inbound: ↓ Outbound: ↑ In/Out delta: 0 %

- **Area Name**: Name of the occupancy monitoring area you're defining.
- **Directory**: User directory for which the occupancy monitoring area will be defined.
- **Site**: Filter that allows you to limit the area to cameras with the specified site value. Site values can be set using the Account Preferences menu within the Desktop Client.
- **Max Occupancy**: Maximum number of allowed people in the occupancy monitoring area.
- **Yellow Alert Occupancy**: Percentage occupancy at which the area is flagged with a yellow alert.
- **Red Alert Occupancy**: Percentage occupancy at which the area is flagged with a red alert.
- **Reset Interval**: Specifies how frequently you want the occupancy level of your occupancy monitor area to reset to 0. This setting can be useful if, for example, you know that your store will always be empty at 2 AM and you want to make sure the occupancy level resets back to 0 every morning. This field has 3 possible values: *Daily*, *Weekly*, and *Never*.
- **Reset At Time**: The time of day when you want the occupancy level of your occupancy monitoring area to reset to zero. **Note**: This field is only available when the *Reset Interval* field is set to *Daily* or *Weekly*.
- **Reset to Occupancy**: The occupancy to which you want to set the occupancy monitoring area when the reset interval occurs.
- **Ignore Person Types**: Specifies the *Person types*, if any, that should be ignored and not counted.
- **Sources**: Specifies which cameras to use to monitor occupancy. The cameras use SAFR's *direction of travel* feature (described in detail in the Camera Preferences documentation). The arrows next

to *Inbound* and *Outbound* represent the camera direction that corresponds to entering or leaving the monitored space. Note that although direction of travel recognition defaults to 10%, we recommend that you go to your Desktop Client's Camera Preferences menu and change the values to 40%.

95 Occupancy Dashboard

Not available for Linux and macOS on-premise deployments.

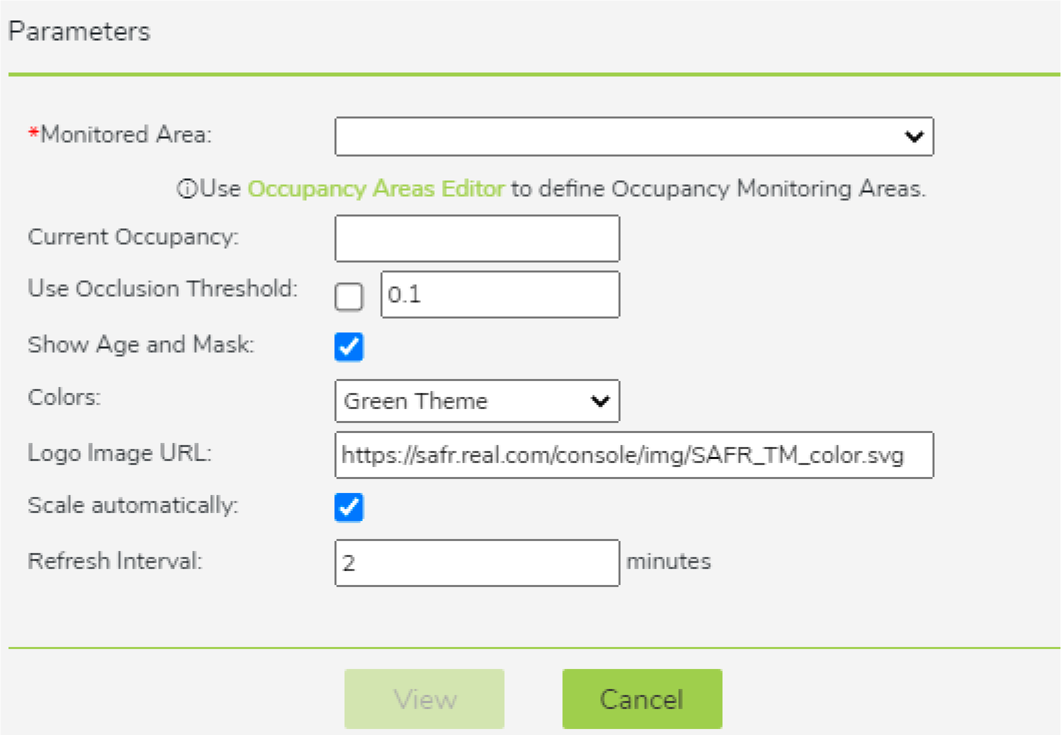
The Occupancy Dashboard is used to monitor how many people are within a defined physical space. The dashboard can also display mask and age information.

The dashboard relies on SAFR's *direction of travel* feature, so the camera(s) that the dashboard uses should be positioned such that people are either moving directly away, directly towards, directly to the left, or directly to the right of the camera. See the Desktop Client Camera Preferences documentation for additional information about the *direction of travel* feature.

In addition, the following preferences must be set for the Occupancy Dashboard to work:

- Every participating camera must be using a Video Processing Mode that can generate events. Specifically, each camera must be using one of the following Video Processing Modes:
  - Learn and Monitor
  - Anonymous Traffic Monitoring
  - Enrolled and Anonymous Traffic Monitoring
  - Enrolled and Unique Traffic Monitoring
  - Enrolled and Stranger Monitoring
- Events must be enabled in the Desktop Client's Events Preferences menu.
- Face detection and person detection must be enabled in the Desktop Client's Detection Preferences menu. If person detection isn't enabled, then anybody whose face can't be seen (e.g. only the back of their head is visible) won't be counted.
- Age and/or mask detection must be enabled in the `Detect` section of the Desktop Client's Recognition Preferences menu for those sections of the Occupancy Dashboard to be populated.

95.1 Input Parameters



Parameters

*Monitored Area:

ⓘUse Occupancy Areas Editor to define Occupancy Monitoring Areas.

Current Occupancy:

Use Occlusion Threshold: □ 0.1

Show Age and Mask: ☑

Colors: Green Theme

Logo Image URL: https://safr.real.com/console/img/SAFR_TM_color.svg

Scale automatically: ☑

Refresh Interval: 2 minutes

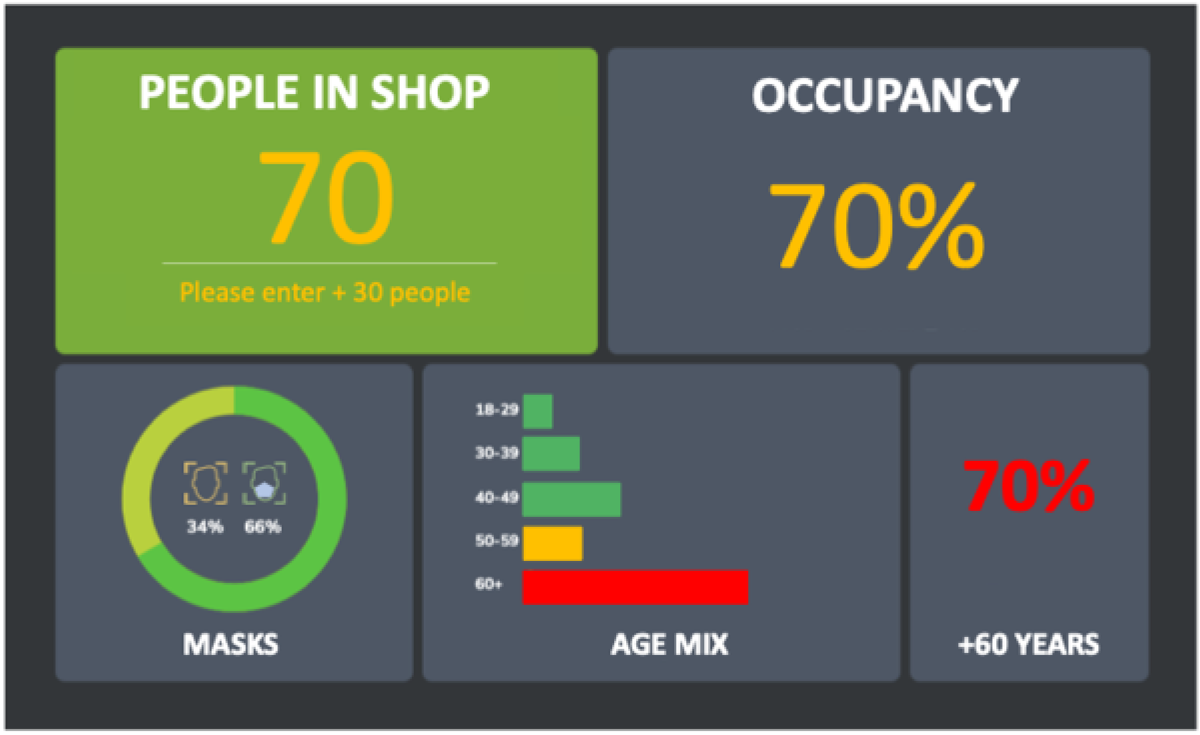
- **Monitored Area**: Specifies the monitored areas to be included in this dashboard. You'll need to define one or more Occupancy Monitoring Areas by using the Occupancy Areas Editor on the Web

Console's Reports Page.

- **Current Occupancy**: The current number of people in the space you want to monitor.
- **Use Occlusion Threshold**: Specifies if occlusion detection is enabled for the Occupancy Dashboard. If enabled, the numerical value indicates the maximum occlusion value allowed for a face to be recognized. When this setting is set to 1, no occlusion filtering is applied.
  Note: Occlusion detection must also be enabled in the `Detect` section on the Desktop Client's Recognition Preferences menu.
- **Show Age and Mask**: Specifies if age and mask information will be included on the dashboard.
  Note: Age and/or mask detection must also be enabled in the `Detect` section on the Desktop Client's Recognition Preferences menu. In addition, the previous parameter, **Use Occlusion Threshold**, must be enabled for mask information to be included in the Occupancy Dashboard.
- **Colors**: Specifies which color scheme will be used for the dashboard. There are three options: *Blue Theme*, *Green Theme*, and *Contrast*.
- **Logo Image URL**: Use this to use a custom logo in place of the SAFR logo at the top of the trend chart.
- **Scale automatically**: When enabled, the dashboard will automatically fill the user's browser window. This facilitates reading the dashboard even when the window is very small. (e.g. on mobile devices)
- **Refresh Interval**: Specifies how frequently the data on the dashboard is refreshed. If "0" is entered, the dashboard won't work. If you want a very quick refresh time, enter a very small non-zero number such as 0.1.

95.2 Generated Dashboard

Below is a sample Occupancy Dashboard.



96 Occupancy Alarm Dashboard

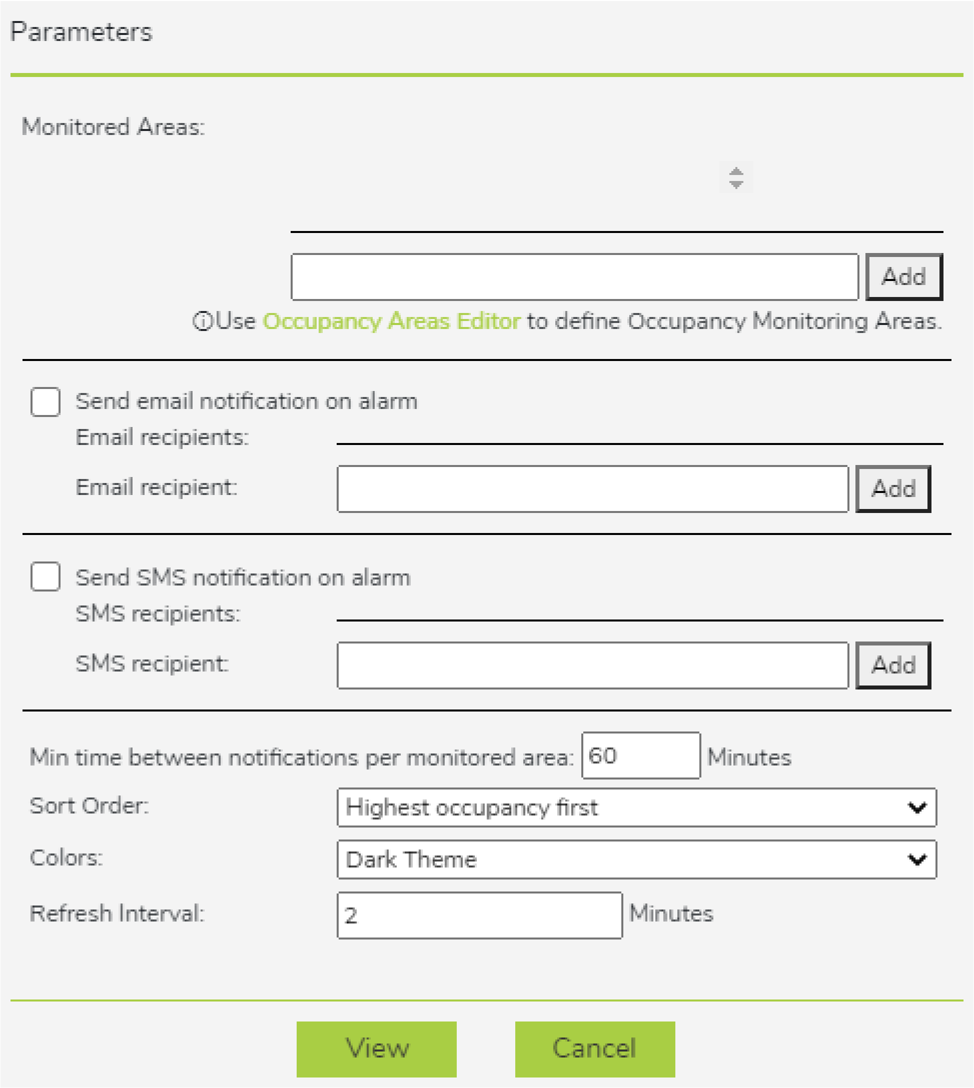
The Occupancy Alarm Dashboard is used to monitor how many people are within one or more physical spaces. When too many people enter one of the areas, notification emails and/or SMS messages are sent to specified recipients.

The dashboard relies on SAFR's *direction of travel* feature, so the camera(s) that your Occupancy Alarm Dashboard uses should be positioned such that people are either moving directly away, directly towards, directly to the left, or directly to the right of the camera. See the Desktop Client Camera Preferences documentation for additional information about the *direction of travel* feature.

In addition, the following preferences must be set for the dashboard to work:

- Every participating camera must be using a Video Processing Mode that can generate events. Specifically, each camera must be using one of the following Video Processing Modes:
  - Learn and Monitor
  - Anonymous Traffic Monitoring
  - Enrolled and Anonymous Traffic Monitoring
  - Enrolled and Unique Traffic Monitoring
  - Enrolled and Stranger Monitoring
- Events must be enabled in the Desktop Client's Events Preferences menu.
- Face detection and person detection must be enabled in the Desktop Client's Detection Preferences menu. If person detection isn't enabled, then anybody whose face can't be seen (e.g. only the back of their head is visible) won't be counted.

96.1 Input Parameters


Parameters

Monitored Areas:

ⓘUse Occupancy Areas Editor to define Occupancy Monitoring Areas.

□ Send email notification on alarm

Email recipients:

Email recipient:

□ Send SMS notification on alarm

SMS recipients:

SMS recipient:

Min time between notifications per monitored area: 60 Minutes

Sort Order: Highest occupancy first

Colors: Dark Theme

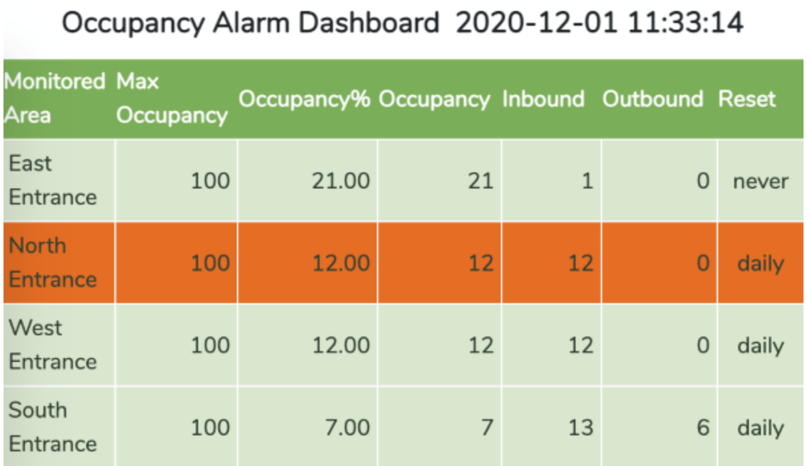
Refresh Interval: 2 Minutes

- **Monitored Areas**: Specifies the monitored areas to be included in this dashboard. You'll need to define one or more Occupancy Monitoring Areas by using the Occupancy Areas Editor on the Web Console's Reports Page.
- **Send email notification on alarm**: Enables sending email notifications when alarms trigger. Note that you first need to configure an email service on the Status Page of the Web Console.
  - **Email recipient**: The list of email recipients.
- **Send SMS notification on alarm**: Enables sending SMS message notifications when alarms trigger. Note that you first need to configure SMS notifications on the Status Page of the Web Console.
  - **SMS recipient**: The list of SMS recipients.
- **Min time between notifications per monitored area**: Specifies the minimum amount of time that must pass before the Occupancy Alarm Dashboard will sent another notification from the same monitored area. Note that this applies to both red and yellow alerts. For example, if you set this parameter to 60 minutes, and an occupancy monitoring area reaches the yellow alert occupancy level, the dashboard won't send out another alert of any type (yellow or red) for that monitoring area for 60 minutes.
- **Sort Order**: Specifies the criteria by which the occupancy areas are sorted. There are 3 options:
  - **Highest occupancy first**: The areas with the greatest number of people appear at the top of

    the dashboard.
    - **Alarm Sources - as configured**: The areas will be listed in the same order that they appear in the *Monitored Areas* parameter above.
    - **Alphabetical by name**: The occupancy areas are sorted based on the alphebetization of their names.
- **Colors**: Specifies which color scheme will be used for the dashboard. There are two options: *Dark Theme* and *Light Theme*.
- **Refresh Interval**: Specifies how frequently the data on the dashboard is refreshed. If “0” is entered, the dashboard won’t work. If you want a very quick refresh time, enter a very small non-zero number such as 0.1.

96.2 Generated Dashboard

Below is a sample Occupancy Alarm Dashboard.



Occupancy Alarm Dashboard 2020-12-01 11:33:14

Monitored Area	Max Occupancy	Occupancy%	Occupancy	Inbound	Outbound	Reset
East Entrance	100	21.00	21	1	0	never
North Entrance	100	12.00	12	12	0	daily
West Entrance	100	12.00	12	12	0	daily
South Entrance	100	7.00	7	13	6	daily

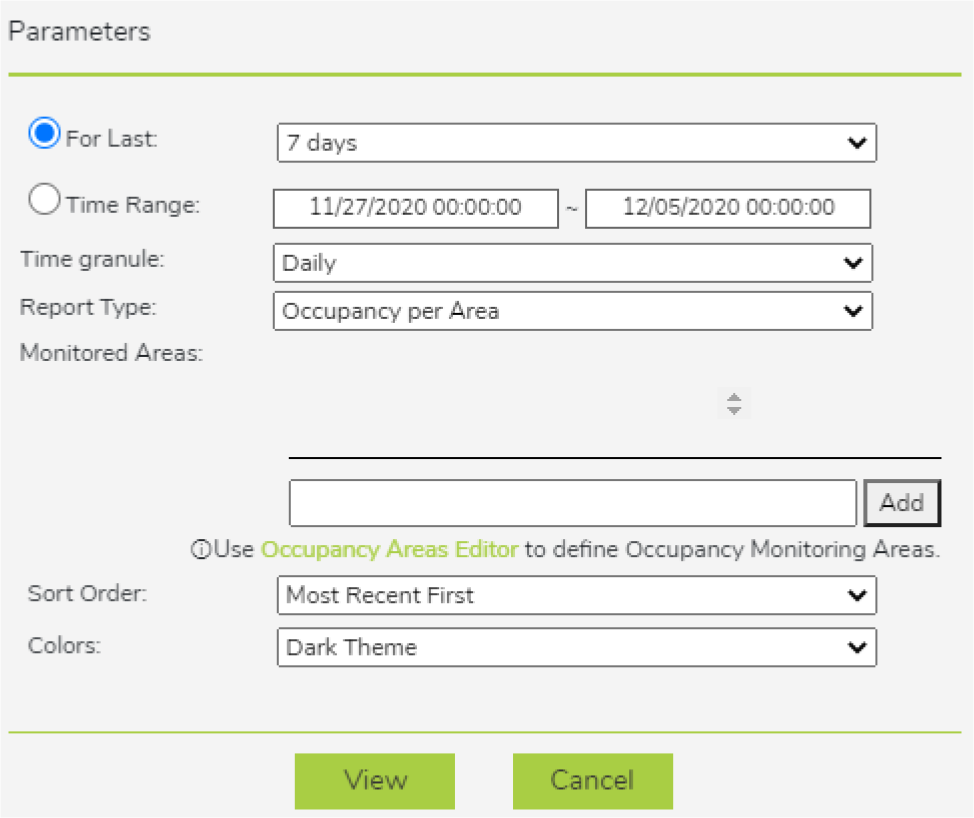
97 Occupancy Report

The Occupancy Report describes how many people were in one or more monitored areas during one or more discrete time periods. The report relies on SAFR's *direction of travel* feature, so the camera(s) that the Occupancy Report uses should be positioned such that people are either moving directly away, directly. towards, directly to the left, or directly to the right of the camera. See the Desktop Client Camera Preferences documentation for additional information about the *direction of travel* feature.

In addition, the following preferences must be set for the report to work:

- Every participating camera must be using a Video Processing Mode that can generate events. Specifically, each camera must be using one of the following Video Processing Modes:
  - Learn and Monitor
  - Anonymous Traffic Monitoring
  - Enrolled and Anonymous Traffic Monitoring
  - Enrolled and Unique Traffic Monitoring
  - Enrolled and Stranger Monitoring
- Events must be enabled in the Desktop Client's Events Preferences menu.
- Face detection and person detection must be enabled in the Desktop Client's Detection Preferences menu. If person detection isn't enabled, then anybody whose face can't be seen (e.g. only the back of their head is visible) won't be counted.

97.1 Input Parameters



Parameters

● For Last: 7 days

○ Time Range: 11/27/2020 00:00:00 ~ 12/05/2020 00:00:00

Time granule: Daily

Report Type: Occupancy per Area

Monitored Areas:

ⓘUse Occupancy Areas Editor to define Occupancy Monitoring Areas.

Sort Order: Most Recent First

Colors: Dark Theme

- **For Last**: Number of previous days to include in the report. There are 5 possible values: *24 hours*, *7 days*, *30 days*, *90 days*, and *365 days*. If you want the report to cover a time period other than these 5 values, use the *Time Range* parameter instead.
- **Time Range**: The dates to include in the report. Use this parameter if you want a custom time range that isn't available in the *For Last* parameter.

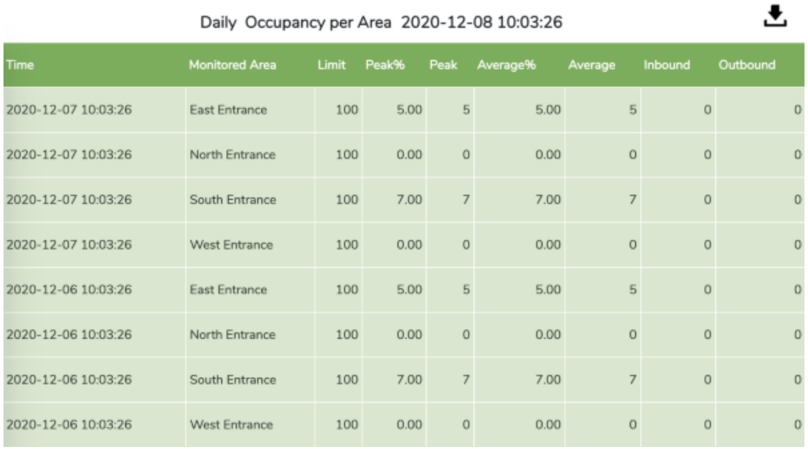
- **Time granule**: Specifies how much time each row of the report covers. There are 4 possible values: *Hourly*, *Daily*, *Weekly*, and *Monthly*.
- **Report Type**: Specifies the type of report to generate. There are 2 types of reports available:
  - **Occupancy per Area**: Each row of the report represents a single occupancy monitoring area.
  - **Traffic per Source**: Each row of the report represents a single camera
- **Monitored Areas**: Specifies the occupancy monitoring areas to be included in the report. You'll need to define one or more areas by using the Occupancy Areas Editor on the Web Console's Reports Page.
- **Sort Order**: Specifies the criteria by which the report rows are ordered. There are 2 options: *Most Recent First* and *Oldest First*.
- **Colors**: Specifies which color scheme will be used for the report. There are two options: *Dark Theme* and *Light Theme*.

97.2 Generated Report

You can download the generated Occupancy Report as an *.xslx file by clicking on the download symbol in the upper right corner.

97.2.1 Occupancy Per Area Report

Below is a sample "Occupancy per Area" Occupancy Report. Entries where the occupancy levels reached yellow or red alert levels will be highlighted in the appropriate color.

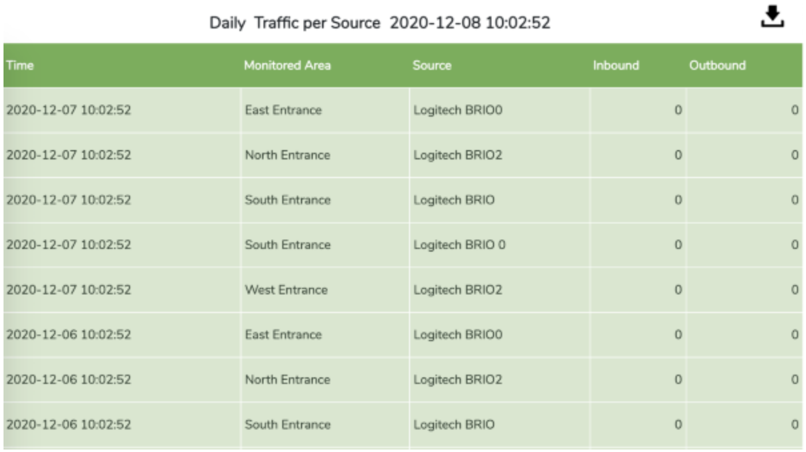


Daily Occupancy per Area 2020-12-08 10:03:26

Time	Monitored Area	Limit	Peak%	Peak	Average%	Average	Inbound	Outbound
2020-12-07 10:03:26	East Entrance	100	5.00	5	5.00	5	0	0
2020-12-07 10:03:26	North Entrance	100	0.00	0	0.00	0	0	0
2020-12-07 10:03:26	South Entrance	100	7.00	7	7.00	7	0	0
2020-12-07 10:03:26	West Entrance	100	0.00	0	0.00	0	0	0
2020-12-06 10:03:26	East Entrance	100	5.00	5	5.00	5	0	0
2020-12-06 10:03:26	North Entrance	100	0.00	0	0.00	0	0	0
2020-12-06 10:03:26	South Entrance	100	7.00	7	7.00	7	0	0
2020-12-06 10:03:26	West Entrance	100	0.00	0	0.00	0	0	0

97.2.2 Traffic per Source Report

Below is a sample "Traffic per Source" Occupancy Report, where each row of the report represents a single video feed. Note that overall occupancy isn't a meaningful concept when looking at single cameras; instead the report only shows the number of people that traveled into or out of the occupany monitoring area.



Daily Traffic per Source 2020-12-08 10:02:52

Time	Monitored Area	Source	Inbound	Outbound
2020-12-07 10:02:52	East Entrance	Logitech BRIO0	0	0
2020-12-07 10:02:52	North Entrance	Logitech BRIO2	0	0
2020-12-07 10:02:52	South Entrance	Logitech BRIO	0	0
2020-12-07 10:02:52	South Entrance	Logitech BRIO 0	0	0
2020-12-07 10:02:52	West Entrance	Logitech BRIO2	0	0
2020-12-06 10:02:52	East Entrance	Logitech BRIO0	0	0
2020-12-06 10:02:52	North Entrance	Logitech BRIO2	0	0
2020-12-06 10:02:52	South Entrance	Logitech BRIO	0	0

98 Face Detection-Person Detection Tie-In

When face detection and person detection are enabled at the same time, face objects will be associated with the appropriate person objects. This allows SAFR to continue tracking people even if they turn their faces away from the camera. In addition, face recognition metadata will be used to automatically enhance the metadata of the associated person object. Each face object can be associated with at most one person object. Similarly, each person object can be associated with at most one face object.

When a face object and person object have become associated with each other, events that are generated by the person object are called "parent events" or "root events", while events generated by the face object are called "children events" or "secondary events". You can choose if secondary events are included in the Event Archive by enabling or disabling the **Include Secondary Events** setting on the Events Preferences tab in the Desktop Client. SAFR's default behavior is to not include secondary events, as they can create too much useless "noise" in the Event Archive.

Face detection and person detection can be enabled and configured on the Detection Preferences tab on Windows machines.

98.1 Shared Event Attributes

When a face object and a person object become associated, they will share the following event attributes:

- age
- avgSentiment
- company
- directGazeDuration
- expDate
- externalId
- gender
- homeLocation
- idClass
- imageTime
- maxSentiment
- minSentiment
- moniker
- name
- newId
- occlusion
- personId
- personTags
- personType
- region
- rootPersonAddDate
- similarityScore
- smileDuration
- tagId
- tagType
- validationEmail
- validationPhone

Whenever a face event is updated with any of the above properties, the associated person event will be updated as well.

Secondary events (i.e. associated face events) have their **rootEventId** attribute set to the eventId of their parent event. (i.e. the person event it's associated with) This enables all secondary events to be gathered and appropriately presented. Each secondary event has only one **rootEventId**.

Conversely, root events (i.e. associated person events) have their **hasSubEvents** attribute set to `true`. Root events aren't ended until all their child events are ended.

99 Identity Recognition Thresholds

SAFR measures the difference between a face image and the stored identity image by a difference measure known as “Identity Recognition”. The value of Identity Recognition can range from 0 to about 1.3. Values of 0.54 or lower represent are interpreted as certain matches, while values above 1.0 are interpreted as different faces.

SAFR has 2 configurable settings that define the acceptable difference between a reference image and the event face image. These are defined as follows:

- **Identity Recognition Threshold** - The largest difference between a face image and a stored identity image at which SAFR will report a 100% confidence match. As noted above, a value of 0.54 or lower represents a certain match. This is the default value for SAFR and usually shouldn’t be changed.
- **Proximity Threshold Allowance** - A boost value that is added to the Identity Recognition Threshold. The sum of Identity Recognition Threshold and Proximity Threshold Allowance is used as the largest difference between a face image and a stored identity image at which SAFR will report a reduced confidence match.

SAFR uses Proximity Threshold Allowance to report possible matches. It does this by reporting a match with a percentage confidence. Any value that is between the Identity Recognition Threshold (e.g. 0.54) and the Proximity Threshold Allowance-boosted value (e.g. 0.92) will be reported as a probable match using a percentage scale as follows:

Proximity Threshold Allowance Boost Confidence Table

Identity Recognition Threshold	Proximity Threshold Allowance	Combined Value	Confidence	Interpretation
0.54	0	0.54 or less	100%	Certain match (values > 100% are possible indicating even greater certainty).
0.54	0.14	0.68	93%	Close match but not certain enough to unlock the door in Secure Access scenarios.
0.54	0.3	0.84	86%	Possible match with low confidence.
0.54	0.38	0.92	82%	Similar face with no confidence of match.
0.54	0.51	1.05	79% or less	Different faces.

The confidence match is overlaid on the videos or images just below the face (alongside the name) and is also reported in the SAFR Events returned through REST APIs in the `confidence` field.

99.1 Typical Uses of Proximity Threshold Allowance

Proximity Threshold Allowance is frequently used in the following scenarios:

- Watchlist monitoring - If you wish to have confidence reported in the user interface, it’s generally recommended you set the Proximity Threshold Allowance to 0.38. With this value SAFR will begin to report faces that are 82% confidence or greater. You can adjust the value of Proximity Threshold Allowance to limit matches to higher or lower confidences, as desired. This is typical for a watchlist scenario where you wish to see possible matches. Operators should be educated so they understand the **Proximity Threshold Allowance Boost Confidence Table** above and can use that information accordingly with possible matched individuals.

- Secure Access - If you do not wish to have confidence reported, then set Proximity Threshold Allowance to 0. In this case only certain matches are reported. This is typically used in Secure Access video processing mode where only certain matches are desired.
- User verification - In some cases, you may wish to validate a user based on a document they present such as a photo ID. While a 100% confidence is desired, this may not always be possible because of the poor condition of the photo ID or because of the age difference between the photo ID and the subject. In these cases, it may be acceptable to use the confidence value returned by SAFR to gauge a close match. For example, a value of 93% confidence could be considered sufficiently close to authorize registration.

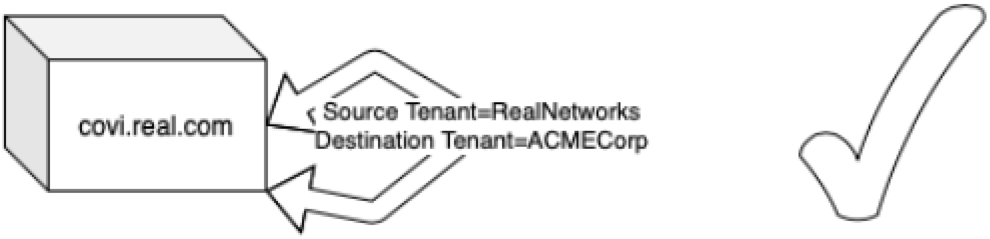
100 Identity Synchronization Configuration

People, faces, and users all belong to tenants. Within a tenant, directories divide persons, faces, and face data. Identity synchronization can be used to copy a directory from one tenant to another but it can't be used to copy data within a tenant.

100.1 Basic Configuration Rules

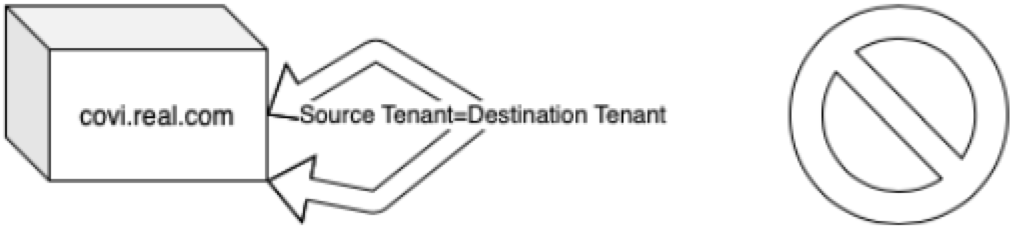
100.1.1 Same Server/Different Tenant

A different user in the cloud can be used as a sync partner as long as the tenant is different. The directory will be duplicated.



100.1.2 Same Server/Same Tenant

A different user in the cloud can be used as a sync partner but not if they both belong to same the tenant. If the tenant is the same there will be conflicts.

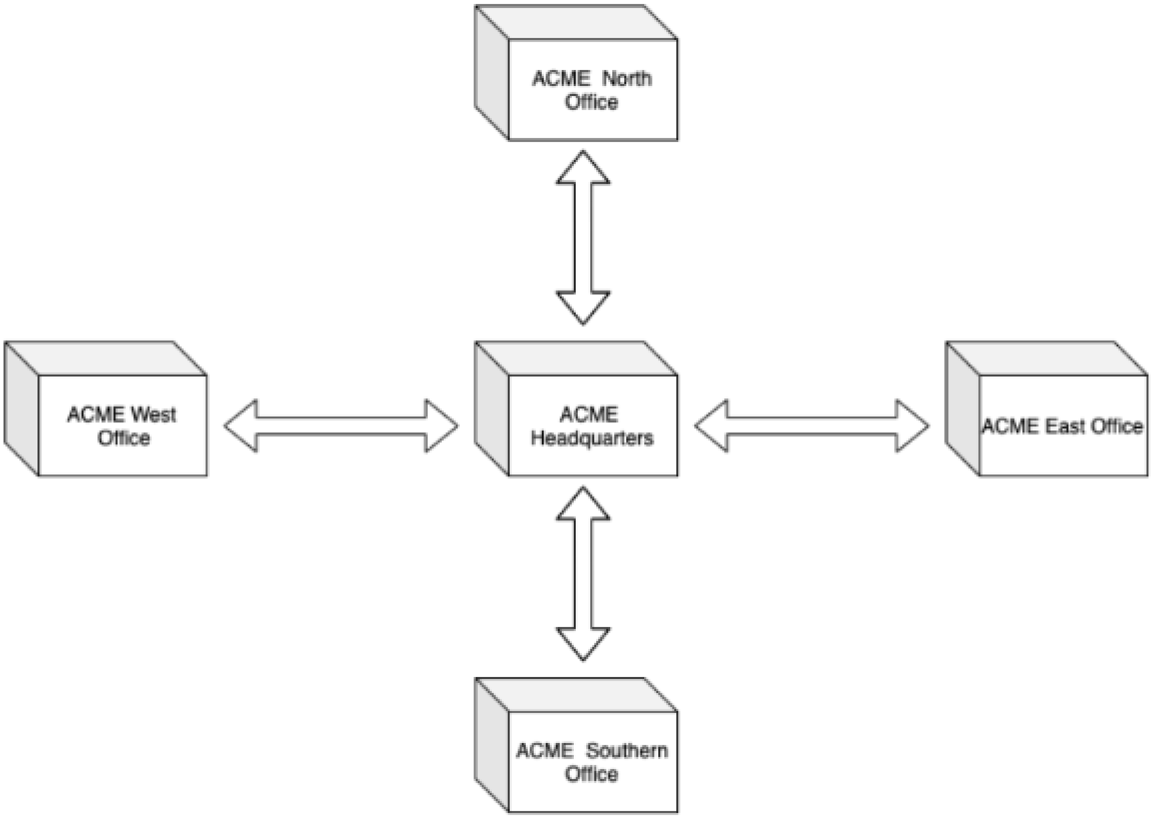


100.2 Multi Node Configuration Examples

More complex configurations can be designed to sync data across multiple locations using combinations of cloud and local platform services.

100.2.1 Central Server Configuration (Spokes on a Wheel)

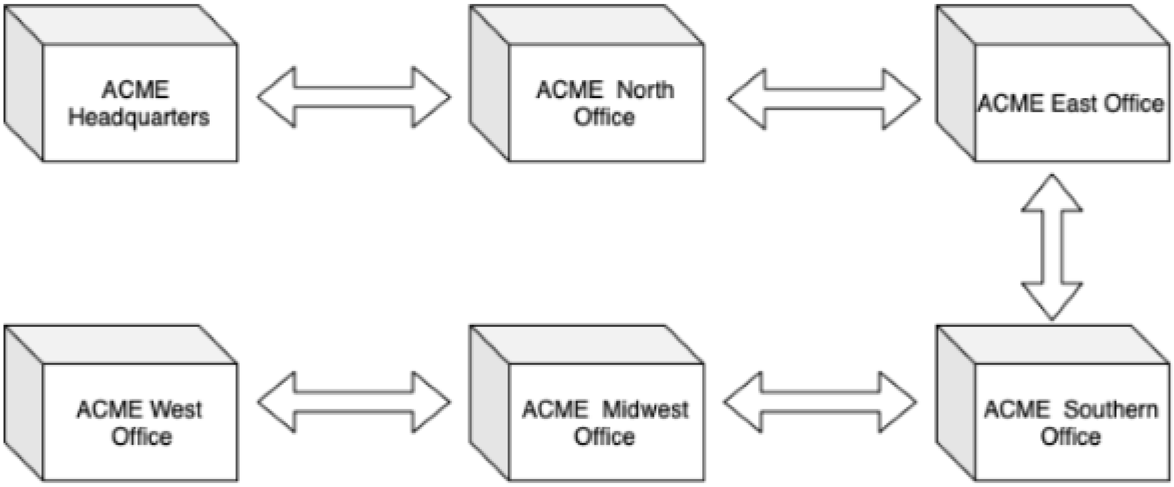
Satellite offices or sites are configured to sync with a central server. This could be a combination of cloud and local platform services.



100.2.2 Daisy Chain Sync Configuration

Local platform installations could also be configured in a daisy-chain type configuration:

Satellite offices or sites are confiured to sync with a nearest node or different account. This could be a mix of cloud and local platform installs.



101 Pose Liveness Detection

Liveness detection is the process whereby facial recognition software attempts to differentiate between genuine live faces and spoofed fake faces. (e.g. a photo of a face) With pose liveness detection, SAFR uses a face's center pose quality to attempt to detect liveness.

Pose liveness detection operates as follows:

1. State A: An unrecognized face needs to be recognized.
2. State B: Proof of liveness will pursued as follows:
    1. The recognized face is tracked at the rate of at least 25 frames per second.
    2. Any loss of tracking (occurrence of lingering for more than 1 frame) or a detection gap > 40 ms in frame capture time results in the need to re-recognize the face and thus a return to State A.
    3. Pose quality must maintain a score of 0.5 or higher for 3 consecutive frames and at least one of the samples must have a profile pose confidence of 35% or less to trigger the transition to the next state, State C.
3. State C: A smooth transition to profile pose will be pursued as follows:
    1. The face is tracked at the rate of at least 25 detections per second.
    2. Any loss of tracking (occurrence of lingering for more than 10 frames) or a detection gap > 40ms in frame capture time results in the need for re-recognition and thus return to State A.
    3. A momentary loss of tracking (recovered in less than 10 frames) will require a center pose quality difference from the prior frame of no less than 0.15.
    4. If change in identity is detected as part of prescribed re-recognition, State B will restarted.
    5. Pose quality must be observed to transition to score of 0.26 or lower for at least 3 consecutive frames and with 66% of at least 3 images but no more than 30 images immediately proceeding with scores observed > 0.26 and < 0.5 and in decreasing sequence to trigger transition to State D.
        - For example: `0.45, 0.37, 0.23, 0.12, 0.24`
        - This algorithm can interpreted as requiring presence of descending strand of samples being at least 66% of the number of samples with min number being specified in preferences and max number being 30 (~1 second).
4. State D: After the profile pose state has changed, a verification call is issued to obtain a similarity score to the identity obtained in State A.
    1. The verification call must indicate at least a 86% match.
    2. A response from recognition must also indicate that the face is in profile pose, based on profile pose confidence returned and threshold set.
    3. If both of above are met, liveness detection will conclude.
    4. If both aren't true, re-recognition will continue immediately for as long as the pose quality score remains at 0.26 or lower until successful confirmation of pose and 86% identity match is confirmed.
    5. If pose score exceeds value of 0.26 for 3 consecutive frames, transition back to state B will occur.
    6. Any loss of tracking (occurrence of lingering) will result in need for re-recognition and thus return to State A.

102 RGB Liveness Detection

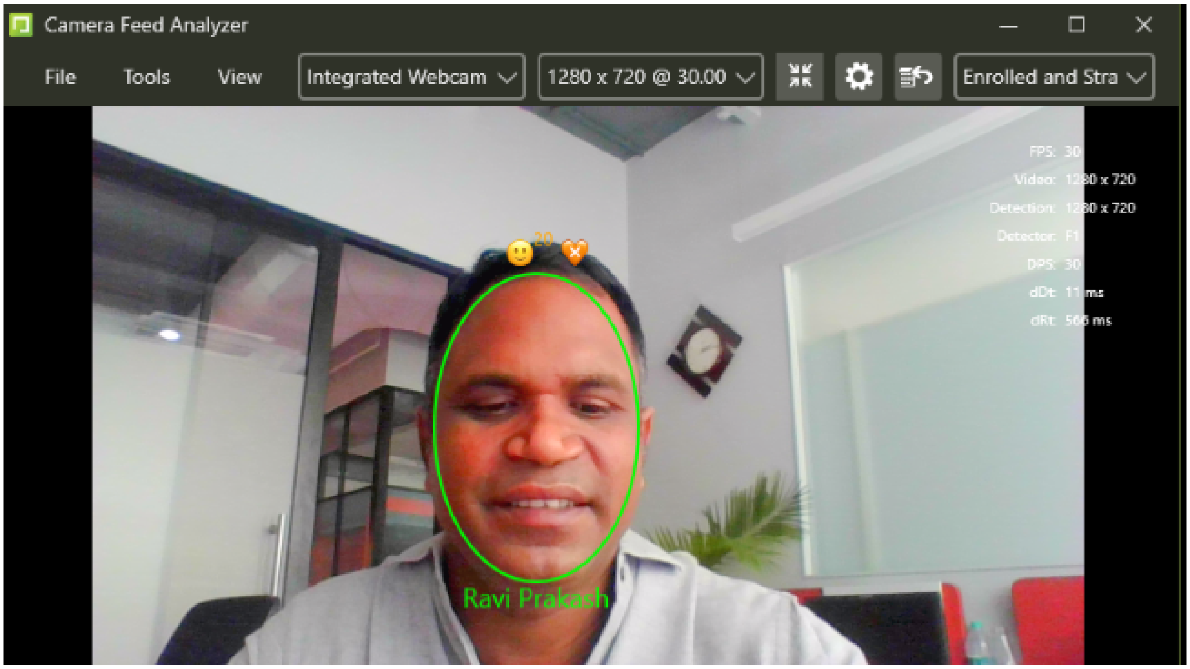
Liveness detection is the process whereby facial recognition software attempts to differentiate between genuine live faces and spoofed fake faces. (e.g. a face appearing in a recorded video file)

RGB liveness detection is only available on Windows machines containing an NVidia card that provides GPU.

102.1 Usage Best Practices

RGB liveness detection was designed with the expectation of a cooperative access control use case. Thus, RGB liveness detection is NOT designed to work using typical surveillance-style camera deployments. (i.e. ceiling mounted cameras, subject faces are seen at angles, faces are in motion, etc.) Instead, RGB liveness detection works best under the following conditions:

- Subjects must look directly at the camera. (i.e. their center pose quality should be very high)
- RGB liveness detection requires high resolution images. Thus, a subject's face should be at least 150 pixels.
- A subject's face must remain fairly still for at least 1 second.
- Lighting must be controlled so that there is even lighting across the subject's face.
- Ideally, there shouldn't be any back-lighting.
- The subject's face shouldn't occupy more than 50% of the screen.
- There should be high color saturation. (i.e. colors should be vibrant)
- There should be low contrast. (i.e. no grain)
- Focus into the distance with reasonable focus in the target range. (The closer the subject is to the camera, the more blur there should be.)
- Exposure should yield a properly exposed face. (i.e. there should be color depth in the face rather than a washed out image)
- The background behind the subject should be as featureless as possible. If the background has edges and/or lines, SAFR may incorrectly interpret the background lines as edges of a tablet, thus causing an erroneous NOTLIVE_CONFIRMED result to be returned.
  For example, in the image below both the greaseboard on the right and the window on the left could cause SAFR to return an incorrect liveness value.



FPS: 30
Video: 1280 x 720
Detection: 1280 x 720
Detector: F1
DPS: 30
dDt: 11 ms
dRt: 566 ms

102.2 Algorithm and Settings

RGB liveness detection uses one or both of the following models to test for liveness, depending on the **Detection scheme** that has been selected in the Detection Preferences menu:

- **Texture Model**: Examines the texture and color of the face. This model is particularly good at detecting printed face images.
- **Context Model**: Examines the context around the face, rather than the face itself.

There are five **Detection schemes** available:

- **Fast Unimodal**: Only the Texture model will be used.
- **Normal Unimodal**: Only the Context model will be used.
- **Strict Multimodal**: Both the Texture and Context models will be used. For a subject to pass the RGB liveness test, both of the results of the models must meet or exceed the **Liveness detection threshold** value. This is the default **Detection scheme** option.
- **Normal Multimodal**: Both the Texture and Context models will be used. For a subject to pass the RGB liveness test, the average of the results of the two models must meet or exceed the **Liveness detection threshold** value.
- **Tolerant Multimodal**: Both the Texture and Context models will be used. Subjects pass the RGB liveness test when the result of either model meets or exceeds the **Liveness detection threshold** value.

When a model is used, the following steps are performed:

1. N frames of the subject's face are evaluated for liveness, as specified by the **Evaluate liveness over N frames** setting.
2. If at least **Minimum confirmations required** x **Evaluate liveness over N frames** frames are encountered where the returned liveness value is equal to or greater than the **Liveness detection threshold** setting, then rgbLivenessState.currentState is set to LIVENESS_CONFIRMED for that subject.
3. If rgbLivenessState.currentState has not been set to LIVENESS_CONFIRMED, then N frames of the subject's face are evaluated for fake, as specified by the **Evaluate fake over N frames** setting.
4. If at least **Minimum confirmations required** x **Evaluate fade over N frames** frames are encountered where the returned liveness value is less than the **Fake detection threshold** setting, then

rgbLivenessState.currentState is set to NOTLIVE_CONFIRMED for that subject.

5. If after both evaluations LivenessConfirmed isn't set to LIVENESS_CONFIRMED or NOTLIVE_CONFIRMED, then rgbLivenessState.currentState will be set to LIVENESS_UNKNOWN. Note that rgbLivenessState.currentState will also be set to LIVENESS_UNKNOWN if the quality of the subject's facial image doesn't meet the specified thresholds. (i.e. the size, center pose, sharpness, and/or contrast of the subject's facial image are so poor that RGB liveness detection can't even be attempted.)

If a multimodal **Detection scheme** has been selected, the Texture Model will be used to evaluate liveness first. If the returned liveness value is lower than the **Minimum preliminary liveness threshold** setting, then no further computations are performed and the liveness value from the Texture Model, capped at **Liveness detection threshold** - 0.01, is returned. If the returned liveness value is equal to or greater than the **Minimum preliminary liveness threshold** setting, then the Context Model is also used to evaluate liveness.

102.3 Troubleshooting

If you're getting incorrect liveness results, try doing the following.

1. Ensure that the subject's face is uniformly lit and isn't washed out.
2. Ensure that the subject's face doesn't occupy more than 50% of the screen.
3. Try changing the background to a plain design. (i.e. the background doesn't contain any lines)

If none of the above fix your issue, it may help to isolate which model is causing the problem.

1. Go to the Detection Preferences menu and select *Texture Unimodal* for the **Detection scheme**. Does this fix your issue? If so, then something about your background is causing your issue.
2. Go to the Detection Preferences menu and select *Context Unimodal* for the **Detection scheme**. Does this fix your issue? If so, then something about your subject's face is causing your issue. (e.g. lighting, out of focus, etc.)

If you'd like the SAFR team to investigate your issue, please capture video recorded directly from the camera (i.e. not screen captures) in order to enable SAFR team to reproduce the issue.

103 SAFR Avigilon Integration Guide

Integrated SAFR Avigilon is only available on Windows.

Integrating SAFR's facial recognition and analysis capabilities into Avigilon enables you to use SAFR's video feed information overlays within Avigilon's camera video feeds, thus making it much easier to quickly and accurately separate unknown people from authorized people from known threats. You'll also have immediate access to additional infomation such as age, gender, sentiment, name, company, known associates, and any other configurable information you want to create.

103.1 Integration Overview and Requirements

A typical deployment requires the following:

- A machine running the Avigilon Control Center (ACC) Server
- One or more machines running the ACC Client.
- One or more machines running the SAFR Desktop Client to process videos.
- If you're doing a SAFR local deployment, you'll also need a machine running SAFR Server. SAFR Server can run on the same machine as one of the Desktop Clients, provided the host machine meets the system requirements.

103.1.1 System Requirements

Avigilon has the following requirements:

- Avigilon 7.4.0 or later

SAFR has the following requirements:

- Each machine running the SAFR Desktop Client must meet the following requirements:
  - Windows 10.
  - The Desktop Client must be version 2.0.106 or later.
  - Additional system requirements as described here.
- Local SAFR deployments require at least one machine running SAFR Platform 2.0.106 or later.
- Each machine running SAFR Server must meet the following requirements:
  - Windows 10.
  - Additional system requirements as described here.

103.2 Install the Avigilon Client and Server

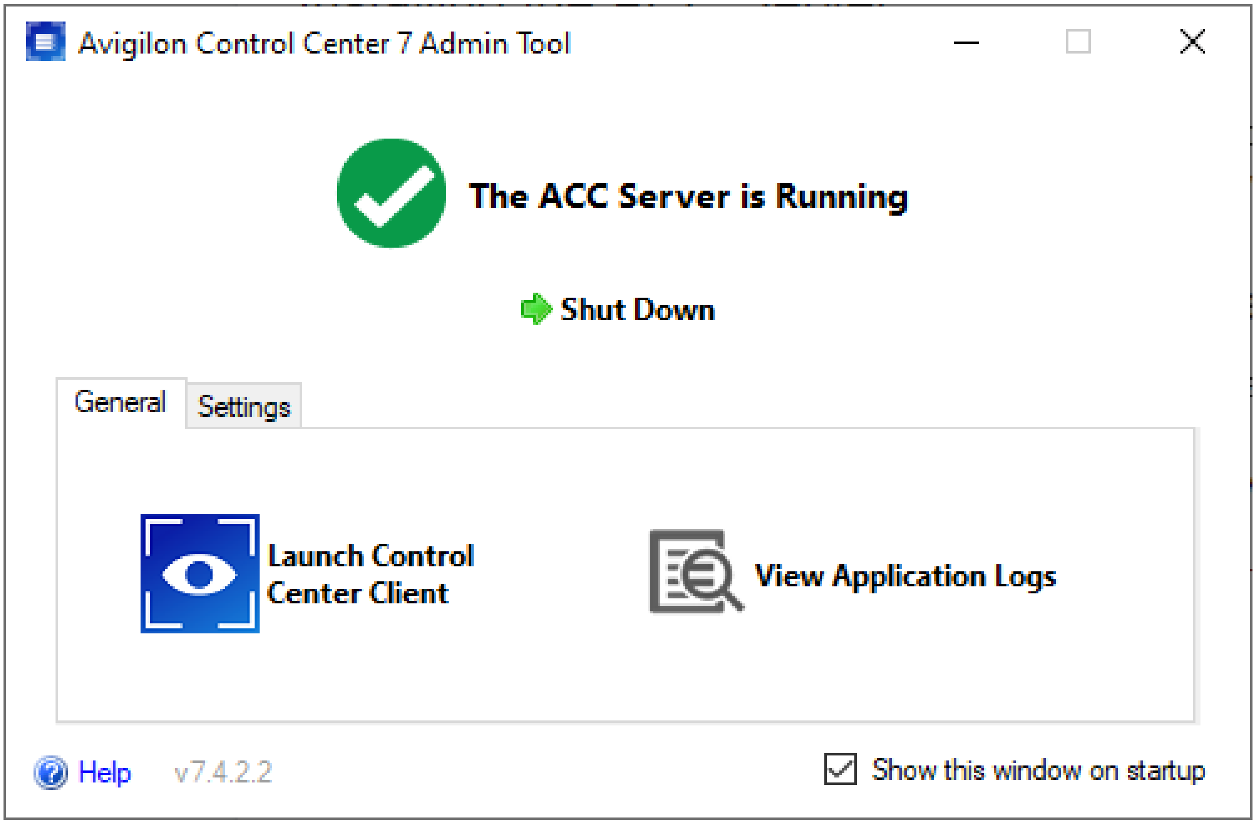
Download and install the ACC Client and the ACC Server from the Avigilon website:

- Full install (server + client): https://partners.avigilon.com/prm/English/s/assets?id=134904

If you've already installed the ACC Server and merely want to install a second ACC Client, there's a client-only install location:

- Client-only install: https://www.avigilon.com/products/acc/7#download (scroll down to the "SOFTWARE DOWNLOADS" section)

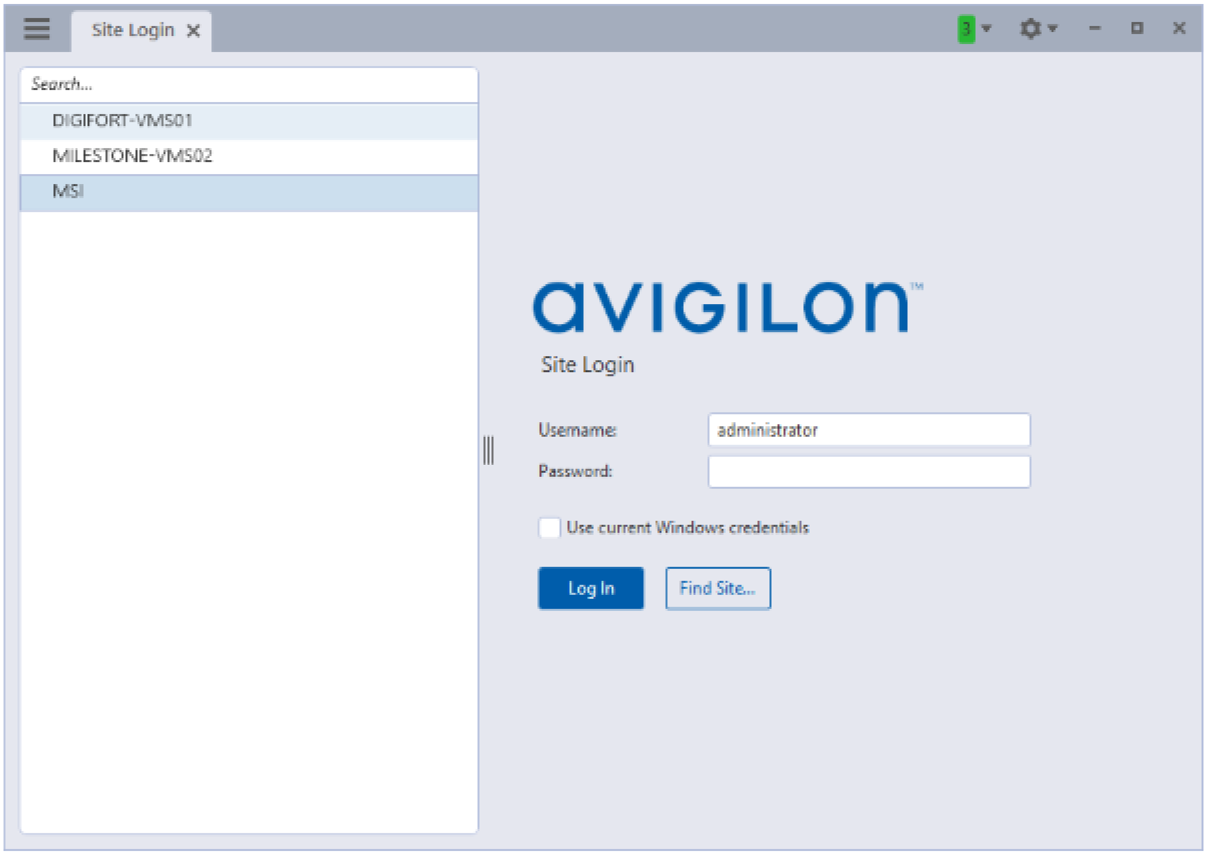
The ACC Admin Tool can be used to manage network and storage configurations.


Avigilon Control Center 7 Admin Tool
Shut Down
General
Settings
Launch Control Center Client
View Application Logs
Help
v7.4.2.2

The ACC Server is Running

☑ Show this window on startup

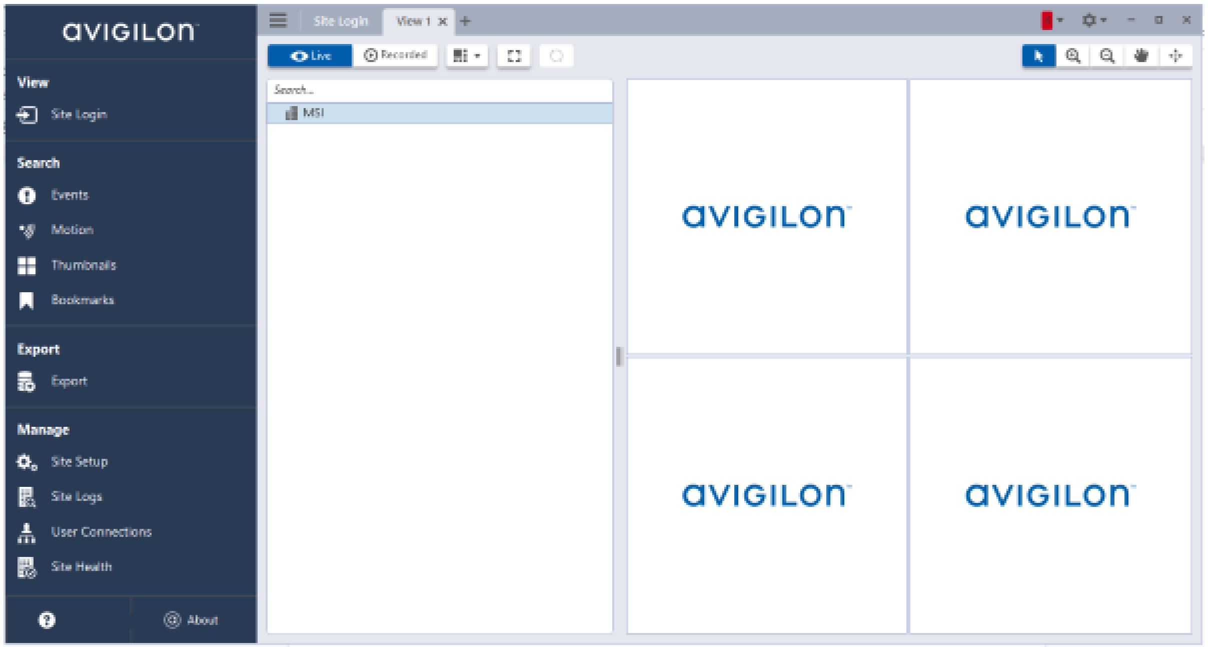
When logging in to the site for the first time, the default credentials use administrator as the username without a password. You'll be asked to immediately enter a new password.



Username: administrator

Password:

□ Use current Windows credentials



103.2.1 Install the ACC Web Endpoint Service

To install the ACC Web Endpoint Service, download and install the ACC 7 Web Endpoint Service from the Avigilon website at https://www.avigilon.com/support/. Note that the ACC Web Endpoint Service must be installed on the same machine as the ACC Server.

Once installed, you can view the health of the ACC 7 Web Endpoint Service at https://localhost:8443/

103.2.2 Change the Default Ports

The default port for the ACC Web Endpoint Service is 8443. You can change the default port by doing the following:

1. In the `%ProgramData%\Avigilon\` folder, open the WebEndpoint.config.yaml file in a text editor.
2. Add the following config parameter to the file, where `123` is the new port number:

   ```
   publicRestInterface: port: 123
   ```

3. Save the config file and restart the ACC Web Endpoint Service.

The default port is updated. All commands should be sent to the new port.

103.2.3 Using Insecure Connections

Although the default connection type used between SAFR and Avigilon is secure, (i.e. HTTPS) insecure connections (i.e. HTTP) are also supported. To change to an insecure connection, do the following:

1. In the %ProgramData% folder, open the WebEndpoint.config.yaml file in a text editor.
2. Add the following config parameter to the file:

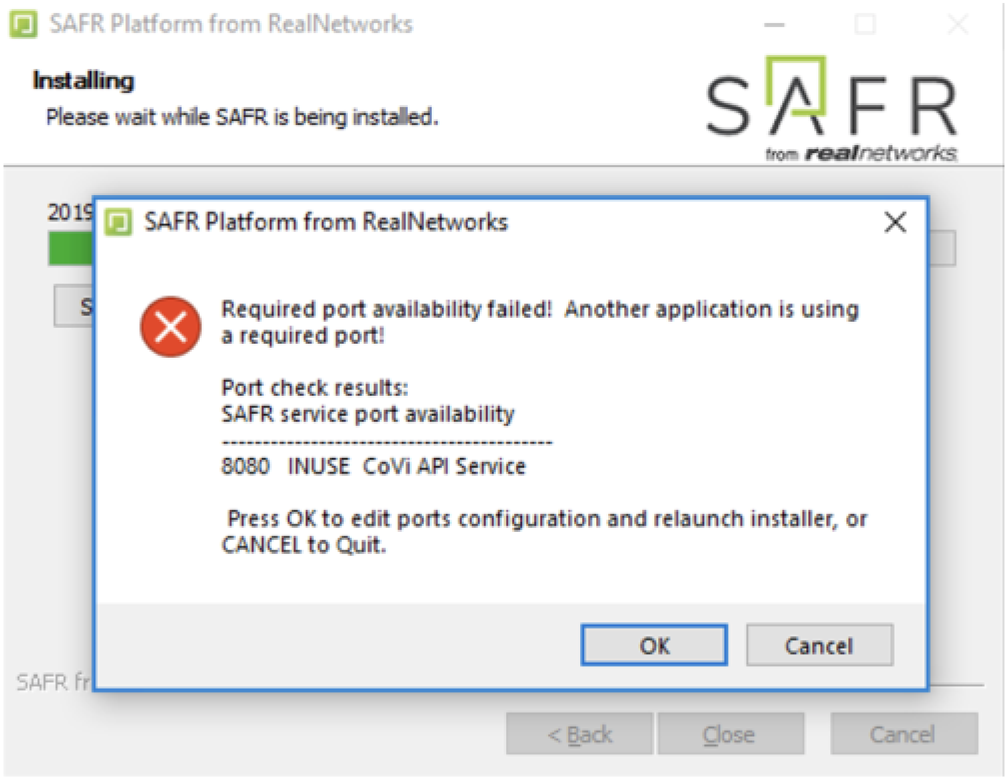
   ```
   publicRestInterface: secure: false
   ```

3. Save the WebEndpoint.config.yaml file and restart the ACC Web Endpoint Service.

All communication with the WebEndpoint will now be done insecurely using HTTP.

103.3 Install and Configure SAFR

1. Go to the SAFR Download Portal.
2. If you're doing a cloud deployment, download and install Windows SAFR Desktop. Make sure to select the Avigilon Control Center install option.
3. If you're doing a local deployment, download and install Windows SAFR Platform. Make sure to select the Avigilon Control Center install option.
   - When installing the SAFR Platform, the default SAFR port assignments sometimes conflict with other software port assignments. If a port conflict occurs, the error message shown below will pop up in the middle of your installation.


SAFR Platform from RealNetworks
Installing
Please wait while SAFR is being installed.
SAFR from realnetworks
< Back
Close
Cancel

SAFR Platform from RealNetworks

Required port availability failed! Another application is using a required port!

Port check results:
SAFR service port availability

```
-----------------------------------------------
8080   INUSE  CoVi API Service
```

Press OK to edit ports configuration and relaunch installer, or CANCEL to Quit.

If this happens, do the following:

1. Click **OK** to edit port configurations.
2. **Notepad** will open, displaying the *safrports.conf* file.
3. Edit any conflicting ports to new values. (e.g. CoviHTTP=18080)
4. Save and exit **Notepad**.

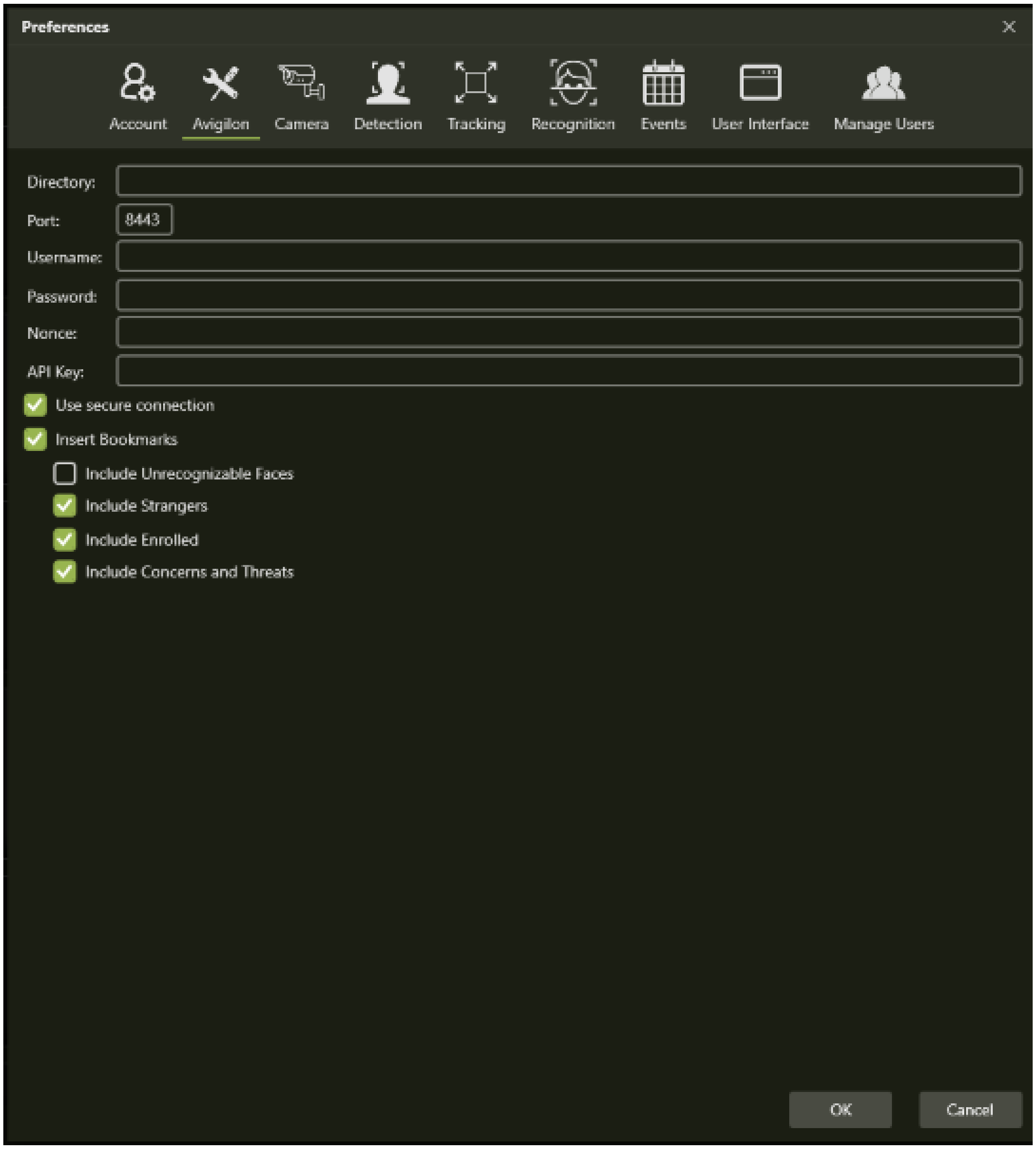
The Platform installer will then restart and the new port values will be used. You can find the modified *safrports.conf* file at `C:\Program Files\RealNetworks\SAFR\`.

After the installation finishes, two icons will appear on your desktop: one labeled *SAFRActions* and another labeled *SAFR*. *SAFRActions* launches SAFR Actions, while *SAFR* launches the Desktop Client. The SAFR Server (when installed as part of a local deployment) automatically runs as a collection of background services.

Immediately following installation, the installer opens the Desktop Client and prompts you to log in with your SAFR Account credentials. Make sure to log in; it's important in acquiring the SAFR license.

103.3.1 Configure SAFR

You can configure several Avigilon-specific preferences by opening the SAFR Desktop Client, going to the Preferences Window and clicking on the Avigilon tab.



Directory:

Port: 8443

Username:

Password:

Nonce:

API Key:

☑ Use secure connection

☑ Insert Bookmarks

- □ Include Unrecognizable Faces
- ☑ Include Strangers
- ☑ Include Enrolled
- ☑ Include Concerns and Threats

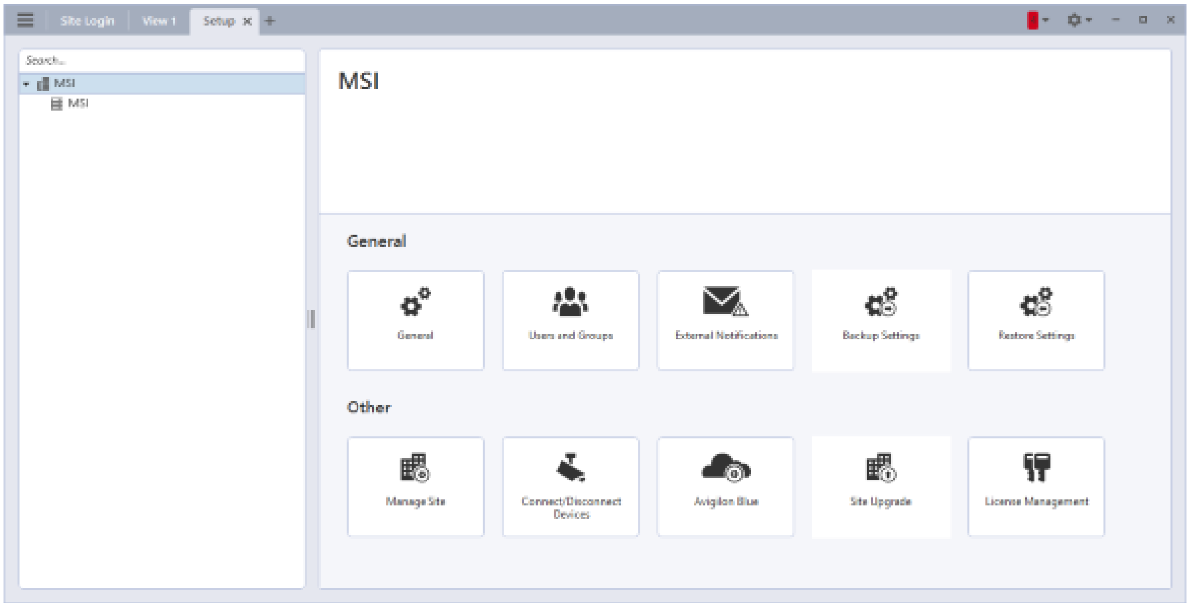
- **Directory**: IP address or hostname of the machine where the ACC server is installed.
- **Port**: The port number that the Avigilon server is configured to use. By default, Avigilon uses port 8443. See Change the Default Port above for information on how to change the port that the Avigilon server uses.
- **Username**: The username of a user that has been added to the Avigilon server via the "Users and Groups" tool.
- **Password**: The password of a user that has been added to the Avigilon server via the "Users and Groups" tool.
- **Nonce**: This value will be provided to you by Avigilon when you obtain a license. It will look something like `FO#26133902`.
- **API Key**: This value will be provided to you by Avigilon when you obtain a license. It will look something like `349f16ea6b3bc5cfd89dfeca3be33a602fcfe7e73b6b7437646a80ae1ed7ce3a`.
- **Use secure connection**: Specifies if SAFR uses a secure connection with Avigilon. By default, Avigilon uses secure connections. Only uncheck this if you have configured Avigilon to use non-secure connections. See Using Insecure Connections above for more information.

104 SAFR Avigilon Operation Guide

104.1 Connect Cameras

To connect a camera to Avigilon, do the following:

1. Open the hamburger menu and select “Site Setup”.
2. Select “Connect/Disconnect Devices”.



Note: Unlike in camera views within SAFR Desktop Clients, camera views within Avigilon don’t have SAFR’s overlay information.

104.2 Bookmarks

To view bookmarks, open the hamburger menu and select “Bookmarks”.

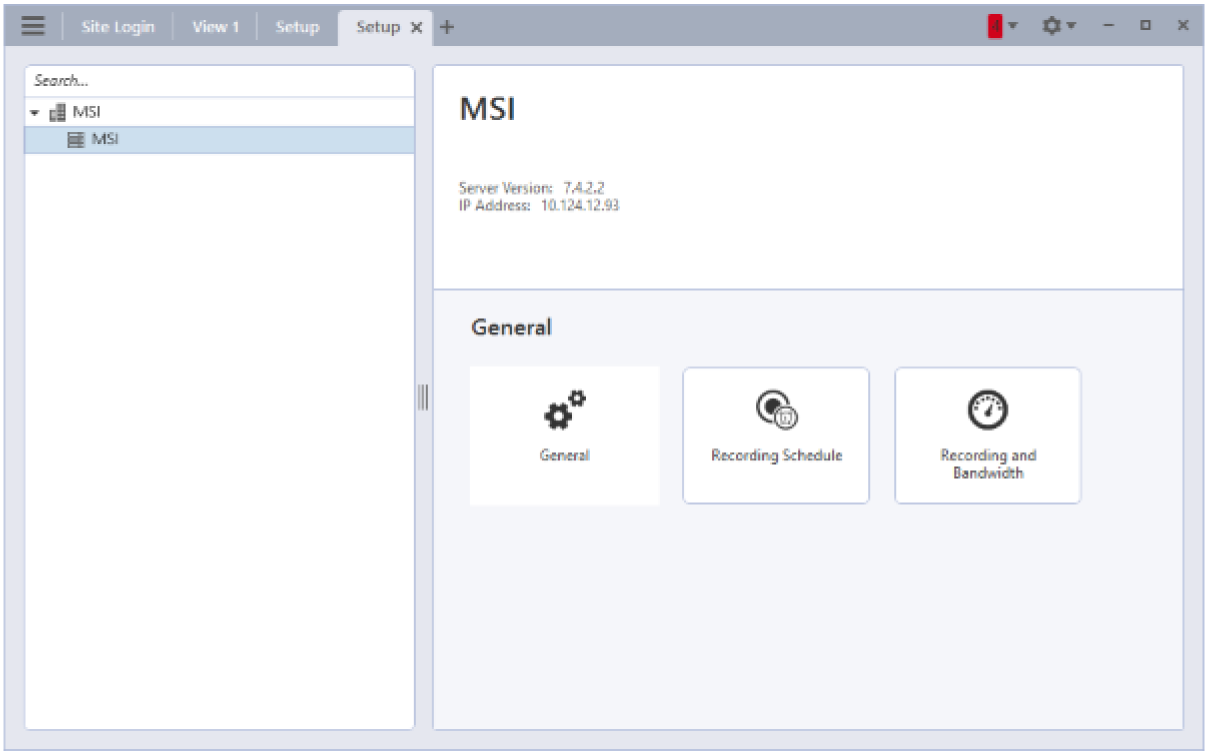
Selecting a bookmark in the left pane will show bookmark details and the associated video clip. Note that if there is no recorded video for the bookmark, then you may see a blank video or a snapshot of the current ‘live’ stream.

Bookmark titles are created to allow easy searching for relevant events

Bookmark Title	Bookmark Criteria	Bookmark Description
SAFR Unrecognizable Face	idClass=“unidentified”	SAFR Unrecognizable face detected.
SAFR Stranger	idClass=“stranger”	SAFR Stranger detected.
SAFR Unnamed Person	idClass=“noconcern” && personType="" && name=""	SAFR Enrolled person detected without name. 98.2% match.
SAFR Person <name>	idClass=“noconcern” && personType="" && name=<name>	SAFR Enrolled person detected with name <name>. 100.0% match.
SAFR Person <personType>	idClass=“noconcern” && personType=<personType> && name=""	SAFR Enrolled person detected of type <personType>. 100.0% match.

Bookmark Title	Bookmark Criteria	Bookmark Description
SAFR Person <personType> <name>	idClass="noconcern" && personType=<personType> && name=<name>	SAFR Enrolled person detected of type <personType> with name <name>. 100.0% match.
SAFR Smile <personType>	personType=<personType> && name=""	SAFR Smile activation by enrolled person of type <personType> without a name. 100.0% match.
SAFR Smile <personType> <name>	personType=<personType> && name=<name>	SAFR Smile activation by enrolled person of type <personType> with name <name>. 100.0% match.
SAFR Concern Person	idClass="concern" && personType="" && name=""	SAFR Concern person detected without a name. 100.0% match.
SAFR Concern Person <name>	idClass="concern" && personType="" && name=<name>	SAFR Concern person detected with name <name>. 100.0% match.
SAFR Concern Person <personType>	idClass="concern" && personType=<personType> && name=""	SAFR Concern person detected of type <personType>. 100.0% match.
SAFR Concern Person <personType> <name>	idClass="concern" && personType=<personType> && name=<name>	SAFR Concern person detected of type <personType> with name <name>. 100.0% match.
SAFR Threat Person	idClass="threat" && personType="" && name=""	SAFR Threat person detected without a name. 100.0% match.
SAFR Threat Person <name>	idClass="threat" && personType="" && name=<name>	SAFR Threat person detected with name <name>. 100.0% match.
SAFR Threat Person <personType>	idClass="threat" && personType=<personType> && name=""	SAFR Threat person detected of type <personType>. 100.0% match.
SAFR Threat Person <personType> <name>	idClass="threat" && personType=<personType> && name=<name>	SAFR Threat person detected of type <personType> with name <name>. 100.0% match.

One way to make sure all bookmarks have recorded clips is to adjust the recording schedule to ensure video is recorded.



Server Version: 7.4.2.2
IP Address: 10.124.12.93

104.3 More Information about ACC Software

For more information about installing, configuring, and using ACC software, see https://www.avigilon.com/support/software/acc7/avigilon-acc7.4-installworkflowchecklist-en-rev2.pdf

105 SAFR Digifort Integration Guide

Integrated SAFR Digifort is only available on Windows.

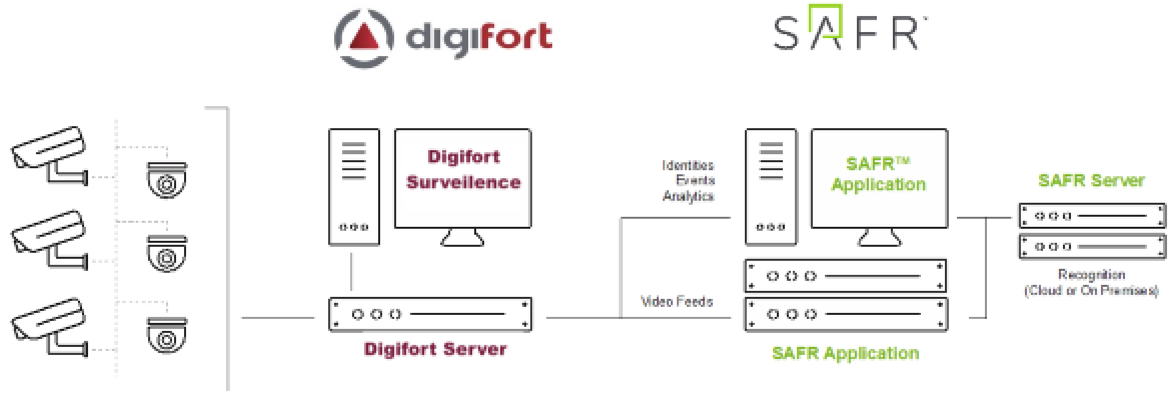
Integrating SAFR's facial recognition and analysis capabilities into Digifort enables you to use SAFR's video feed information overlays within Digifort camera video feeds, thus making it much easier to quickly and accurately separate unknown people from authorized people from known threats. You'll also have immediate access to additional infomation such as age, gender, sentiment, name, company, known associates, or any other configurable information you want to create.

Integrating the two systems also allows SAFR's information about individuals to trigger Digifort alerts and other actions within the Digifort system. Digifort's metadata within bookmarks are enriched with SAFR's additional information, allowing you to more easily find relevant bookmarks.

105.1 Integration Overview and Requirements

A typical deployment requires the following:

- A machine running Digifort Server and Digifort Administration Client.
- A machine running the Digifort Surveillance Client at monitoring locations.
- One or more machines running the SAFR Desktop Client to process videos.
- If you're doing a SAFR local deployment, you'll also need a machine running SAFR Server. SAFR Server can run on the same machine as one of the Desktop Clients, provided the host machine meets the system requirements.



Cameras are connected to Digifort. The SAFR Desktop Client can connect to Digifort to perform analysis of the video. Depending on the number of cameras you need, one or more machines can run SAFR Desktop, each processing multiple video feeds. The Desktop Client processes the video and returns information to Digifort to generate events. The Desktop Client is also used to perform various management activities. This could be run on the same system as Digifort Server.

105.1.1 System Requirements

Digifort has the following requirements:

- Each machine running Digifort must meet the following requirements:
  - The Digifort version must be 7.2.1 or later.
  - The machine must be running Windows 10 or later.
  - .Net Framework 4.6.2 or later must be installed.
- Each camera connected to Digifort requires a Digifort license.

Note: Digifort licenses must be acquired before attempting to discover and add cameras.

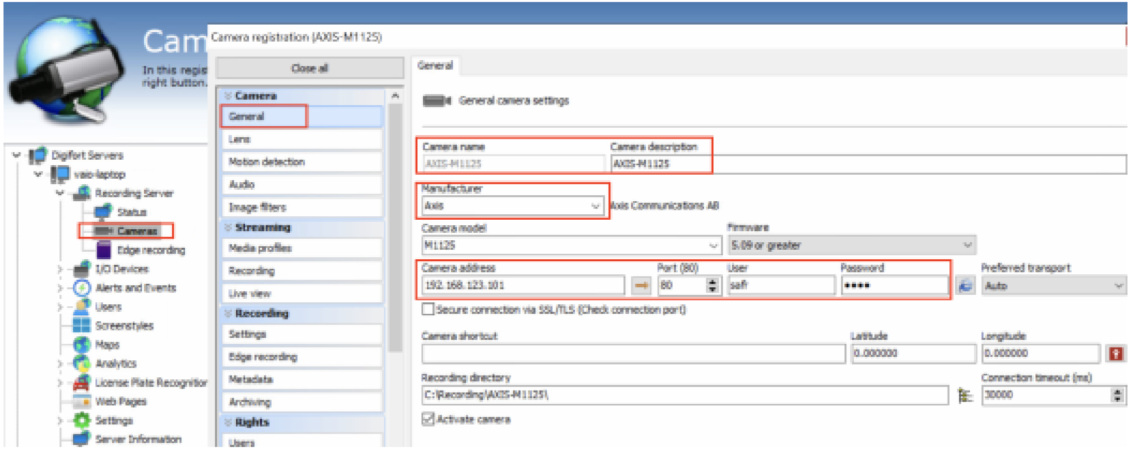
SAFR has the following requirements:

- Each camera running SAFR must have a SAFR license.
- Each machine running the SAFR Desktop Client must meet the following requirements:
  - The Desktop Client must be version 1.4.162 or later.
  - The system requirements described here.
- Local SAFR deployments require at least one machine running SAFR Platform 1.4.157 or later.
- Each machine running SAFR Platform must meet the system requirements described here.

105.2 Install and Configure Digifort

Download and install Digifort by doing the following:

1. Turn off Windows Defender and Firewalls. (Digifort requires that both these features be disabled.)
2. Download the latest Digifort installer package from Digifort and install the full package.
3. Open the *Digifort Administration Client* and add a new server. Log in with the default username **admin** and set a password in the **Users** tab. As an administrator, you can create additional user accounts if needed.
4. To connect cameras to Digifort, do the following:
   1. Go to Recording Servers in the Digifort Administration Client.
   2. Click **Camera** from the Add Button.
   3. Complete the dialog shown in the following graphic to add Camera manufacturer, model, IP Address, and other information as needed.

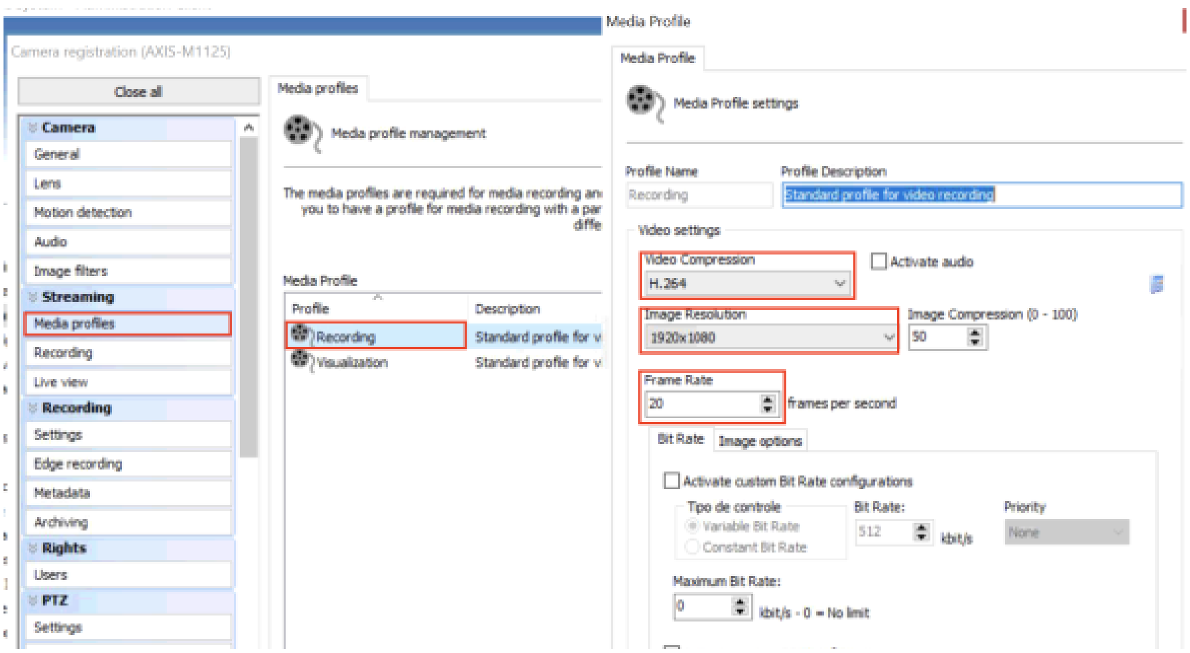


Camera registration (AXIS-M1125)

General camera settings

Camera name: AXIS-M1125
Camera description: AXIS-M1125
Manufacturer: Axis — Axis Communications AB
Camera model: M1125
Firmware: 5.09 or greater
Camera address: 192.168.123.101
Port (80): 80
User: safr
Password: ••••
Preferred transport: Auto
□ Secure connection via SSL/TLS (Check connection port)
Camera shortcut:
Latitude: 0.000000
Longitude: 0.000000
Recording directory: C:\Recording\AXIS-M1125\
Connection timeout (ms): 30000
☑ Activate camera

   Note: Digifort allows you to add mobile cameras as well as using the Digifort Mobile Camera Pro App. In the Add Camera dialog, select **Digifort** as the manufacturer and **Mobile Camera Pro** as the make.
5. You may need to update the *Video Compression*, *Image Resolution*, *Frame Rate*, or *Image Rotation*, as shown below:



Media profile management

The media profiles are required for media recording an
you to have a profile for media recording with a par
diffe

Media Profile

Profile	Description
Recording	Standard profile for v
Visualization	Standard profile for v

Media Profile settings

Profile Name: Recording

Profile Description: Standard profile for video recording

Video settings

Video Compression: H.264

□ Activate audio

Image Resolution: 1920x1080

Image Compression (0 - 100): 50

Frame Rate: 20 frames per second

Bit Rate | Image options

□ Activate custom Bit Rate configurations

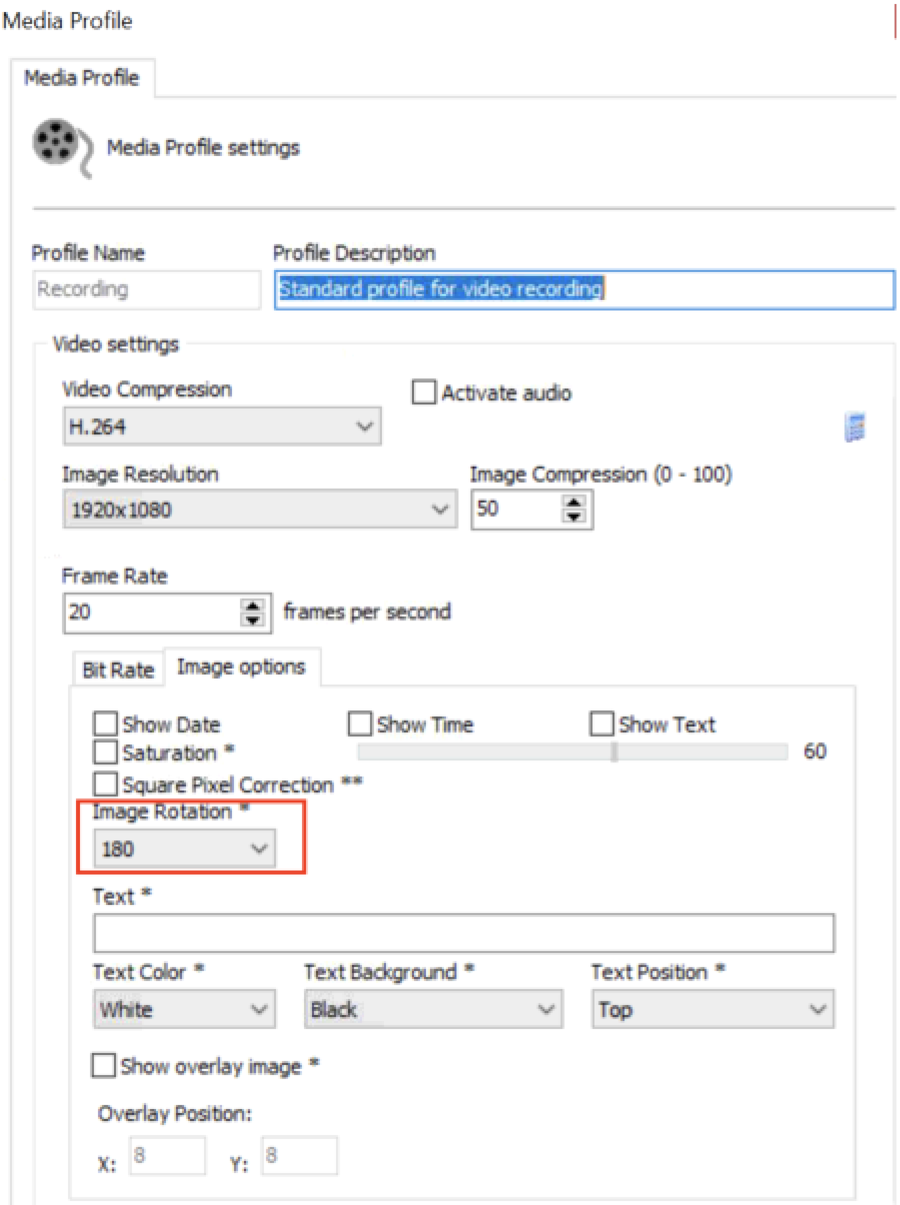
Tipo de controle: ● Variable Bit Rate ○ Constant Bit Rate

Bit Rate: 512 kbit/s

Priority: None

Maximum Bit Rate: 0 kbit/s - 0 = No limit

6. You may need to update the *Image Rotation* or *Profile Description* on the Media Profile settings page, as shown below:



Profile Name: Recording

Profile Description: Standard profile for video recording

Video settings

Video Compression: H.264

□ Activate audio

Image Resolution: 1920x1080

Image Compression (0 - 100): 50

Frame Rate: 20 frames per second

□ Show Date □ Show Time □ Show Text

□ Saturation * 60

□ Square Pixel Correction **

Image Rotation * 180

Text *

Text Color * White

Text Background * Black

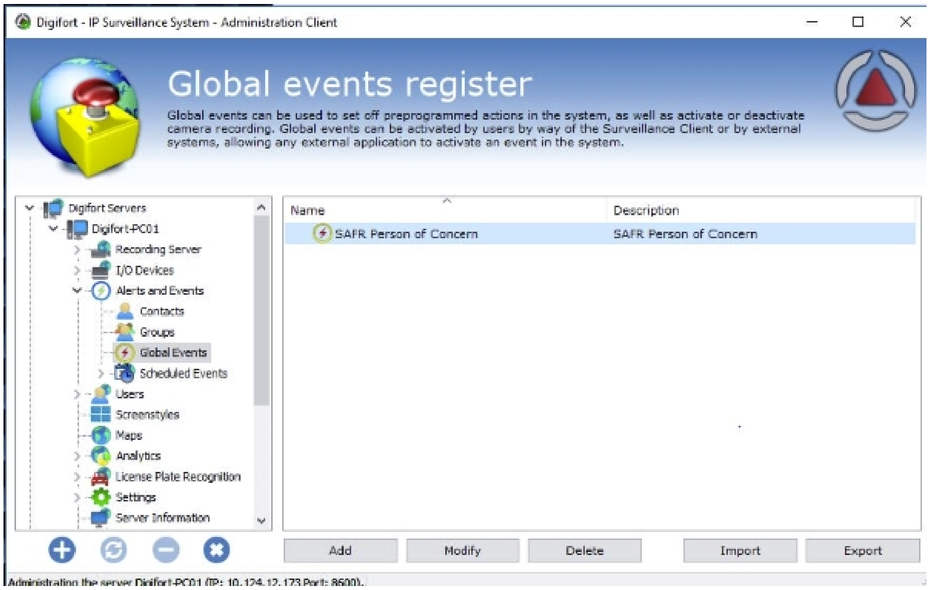
Text Position * Top

□ Show overlay image *

Overlay Position:

X: 8 Y: 8

7. Navigate to **Alerts and Events > Global Events** to add events as needed. You can configure the event actions there too, when needed.



Global events register

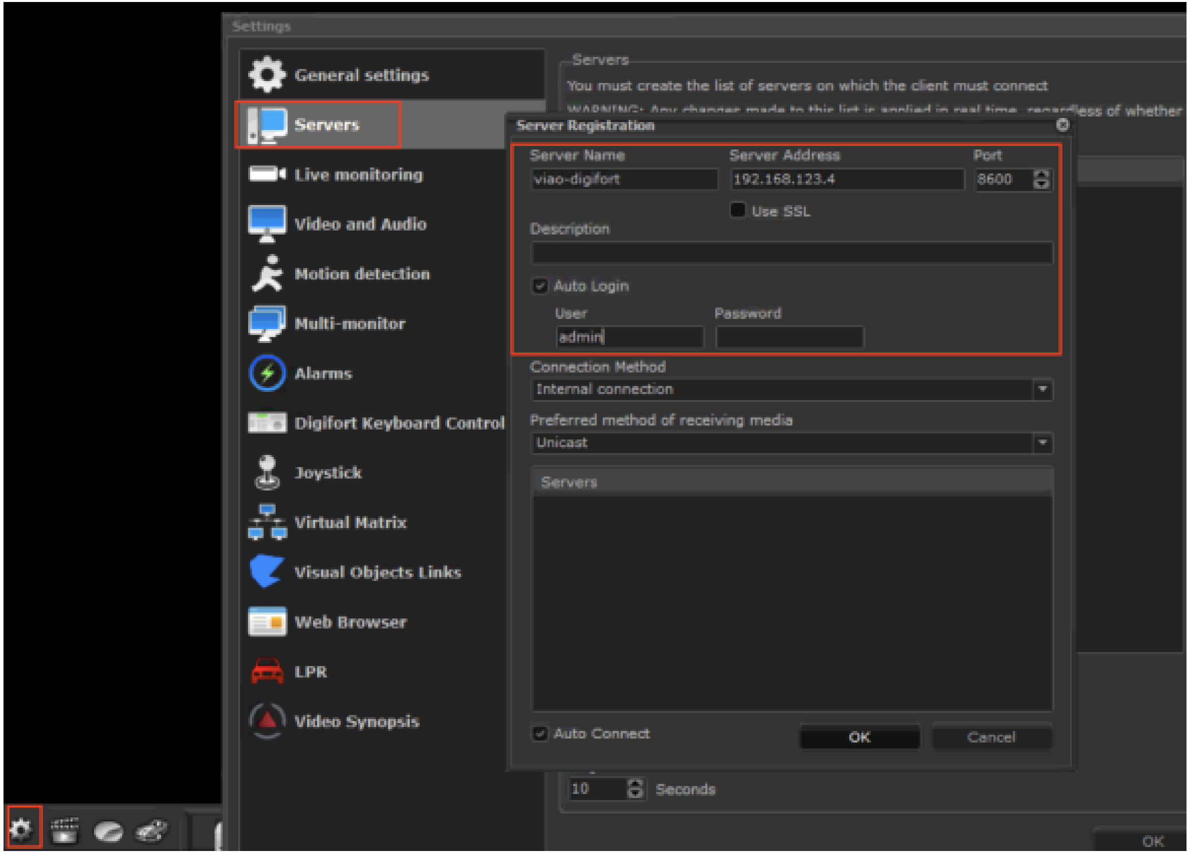
Global events can be used to set off preprogrammed actions in the system, as well as activate or deactivate camera recording. Global events can be activated by users by way of the Surveillance Client or by external systems, allowing any external application to activate an event in the system.

Name	Description
SAFR Person of Concern	SAFR Person of Concern

Administrating the server Digifort-PC01 (IP: 10.124.12.173 Port: 8600).

Configure the Digifort Surveillance Client by doing the following:

1. Start Digifort Serveillance Client.
2. Go to Setting > Servers > Add.
3. Restart Digifort Serveillance Client.
4. Select the IP camera from the list.



Servers

You must create the list of servers on which the client must connect

WARNING: Any changes made to this list is applied in real time, regardless of whether

Server Name: viao-digifort

Server Address: 192.168.123.4

Port: 8600

□ Use SSL

Description:

☑ Auto Login

User: admin

Password:

Connection Method: Internal connection

Preferred method of receiving media: Unicast

Servers

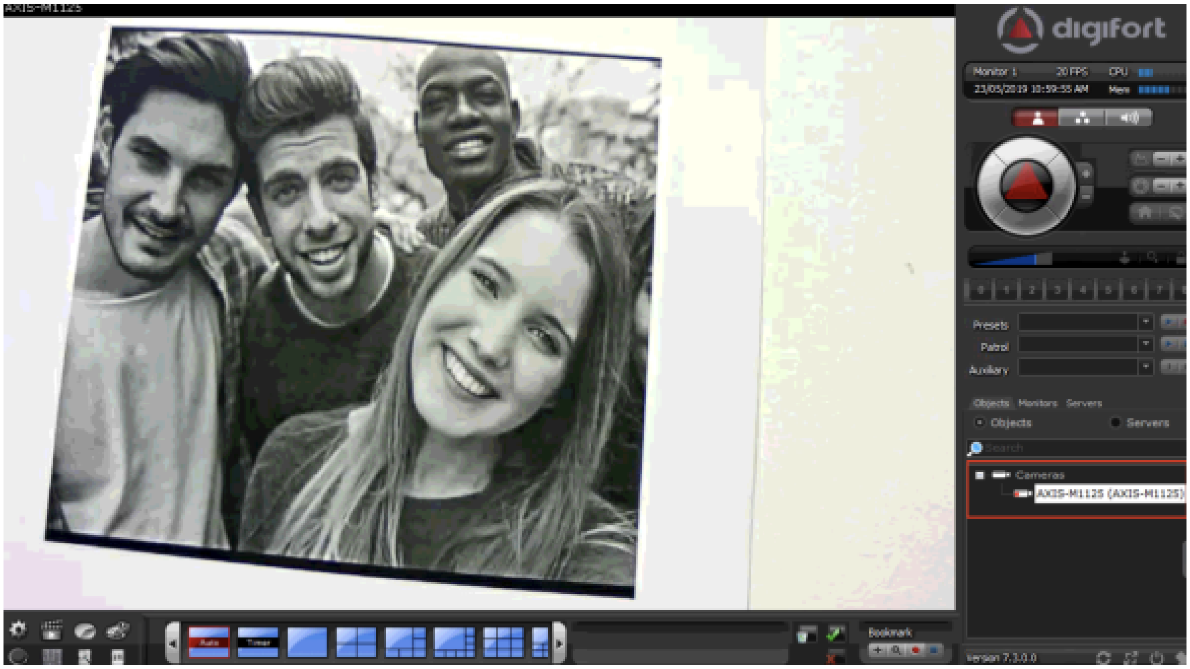
☑ Auto Connect

10 Seconds

105.2.1 Connect an IP Camera

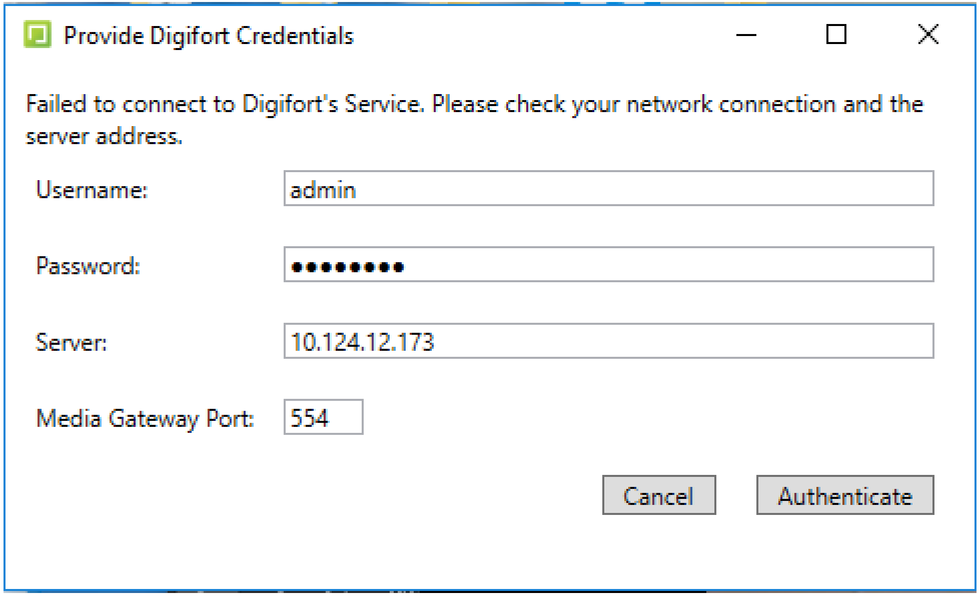
Important: Digifort licenses must be acquired prior to attempting to discover and add cameras.

1. Start the Digifort Serveillance Client.
2. Select the IP camera from the Cameras list.



105.3 Install and Configure SAFR

1. From the SAFR Download Portal, download and install either SAFR Platform or SAFR Desktop, depending on your deployment type. Make sure to select the Digifort VMS extension install option.
2. After installing SAFR, you'll be prompted for the Digifort Credentials as shown in the following dialog:



Provide Digifort Credentials

Failed to connect to Digifort's Service. Please check your network connection and the server address.

Username: admin

Password: ●●●●●●●●●

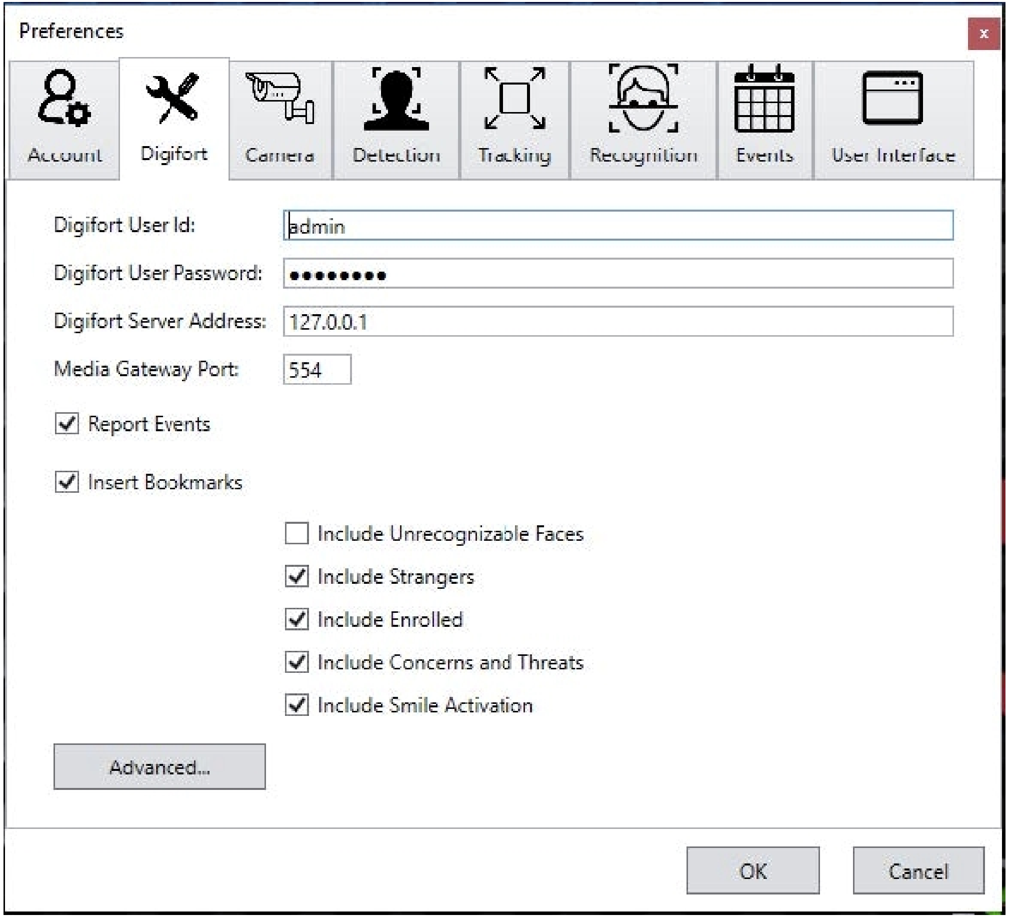
Server: 10.124.12.173

Media Gateway Port: 554

3. Enter the information for the Digifort user created previously to connect to Digifort server, and click **OK**.

Note: During the Digifort login and authentication process, you may be prompted to enter your SAFR account credentials as well as to log into any automatically detected cameras.

4. After SAFR finishes installing, open the SAFR Desktop Client.
5. From the **Tools** menu, select **Preferences**, and click the Digifort tab.



Digifort User Id: admin

Digifort User Password: •••••••••

Digifort Server Address: 127.0.0.1

Media Gateway Port: 554

☑ Report Events

☑ Insert Bookmarks

□ Include Unrecognizable Faces

☑ Include Strangers

☑ Include Enrolled

☑ Include Concerns and Threats

☑ Include Smile Activation

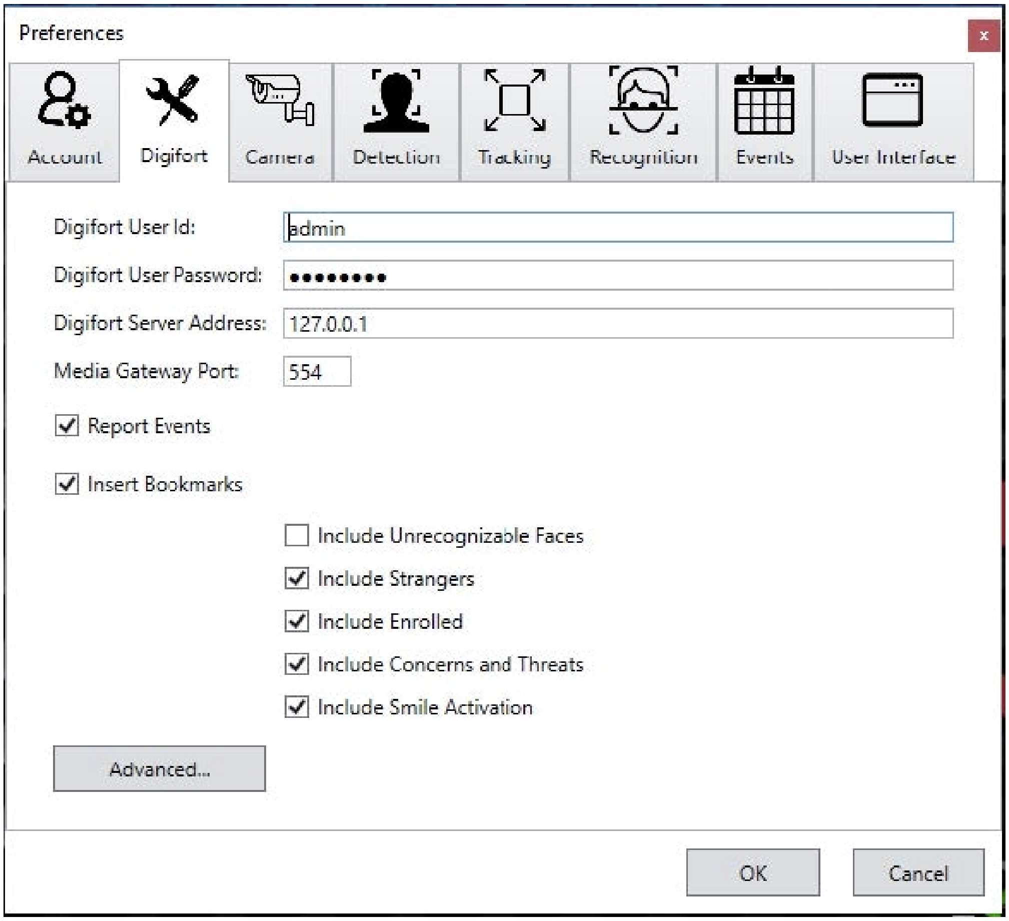
6. Enter the following information.
   - **Digifort User Id**: User created previously in Configure Digifort.
   - **Digifort User Password**: Password created for the SAFR user.
   - **Digifort Server Address**: IP address of server running Digifort.
   - **Media Gateway Port**: Set to 554 unless configured otherwise in Digifort.
7. Click **OK**.

105.4 Verify your Connection

To verify successful connection to the Digifort system, open the **Preferences > Camera** tab. Cameras connected to the Digifort system should be visible. All cameras connected to the Digifort system have a Digifort prefix in their names.

106 SAFR Digifort Operation Guide

106.1 SAFR Digifort Preferences



Digifort User Id: admin

Digifort User Password: ●●●●●●●●

Digifort Server Address: 127.0.0.1

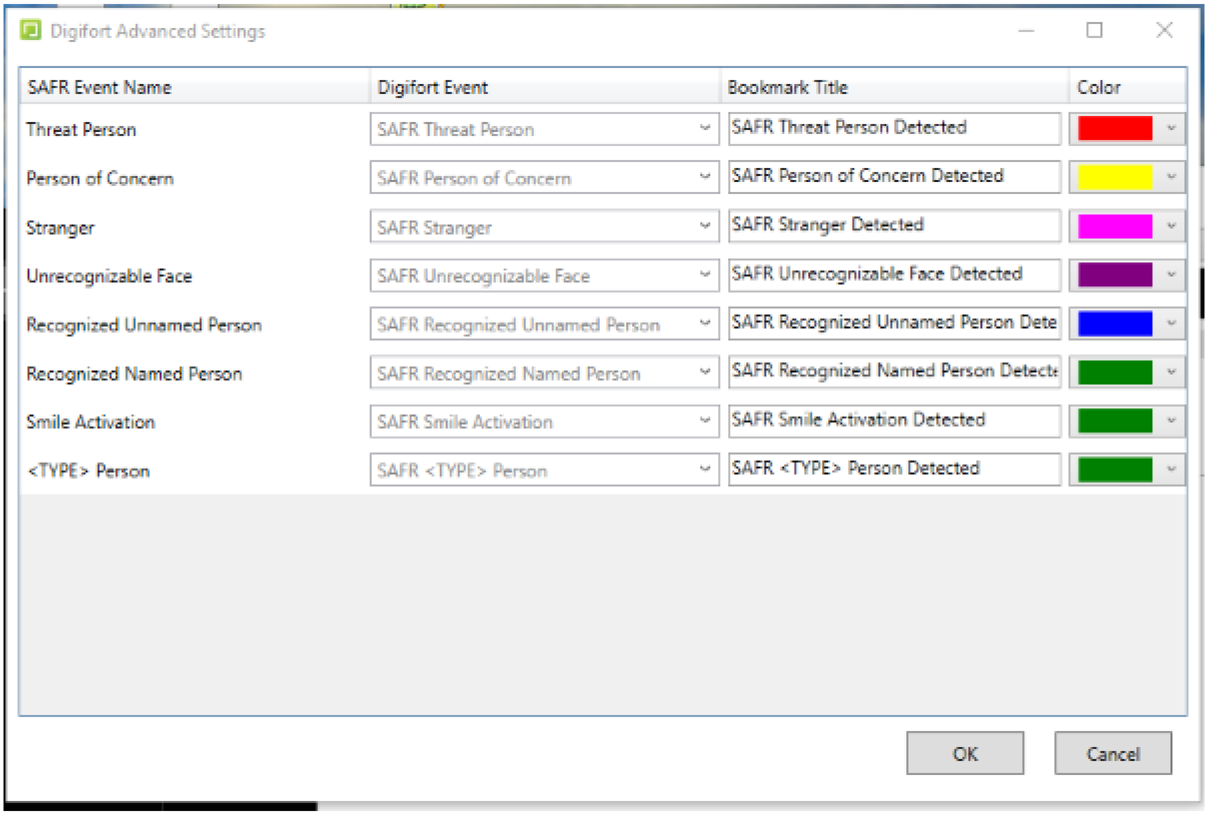
Media Gateway Port: 554

☑ Report Events

☑ Insert Bookmarks

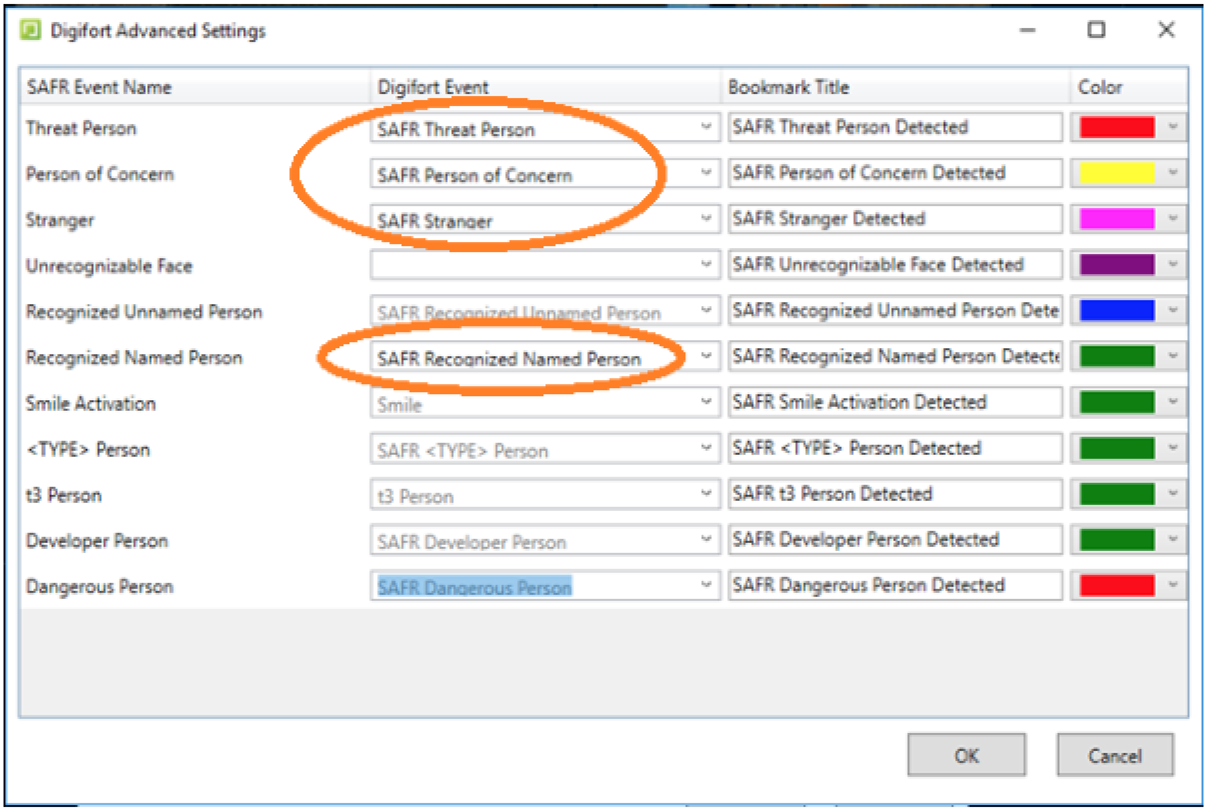
- □ Include Unrecognizable Faces
- ☑ Include Strangers
- ☑ Include Enrolled
- ☑ Include Concerns and Threats
- ☑ Include Smile Activation

- **Digifort User Id**: User created previously in Configure Digifort
- **Digifort User Password**: Password created for the SAFR user
- **Digifort Server Address**: IP address of server running Digifort
- **Media Gateway Port**: Set to 554 unless configured otherwise in Digifort
- **Report Events**: Controls if events are sent to Digifort. Events are used to trigger alarms in Digifort.
- **Insert Bookmarks**: Adds bookmarks to the video stream related to events. Allows operators to search video for events or recognized person names. **Note**: Use caution when deciding what to include since many faces can cause many bookmarks to be created.
  - **Include Unrecognizable Faces**: Adds bookmarks when a face detected by SAFR does not have enough information to determine if it is a stranger or known person. This can become visually noisy and is disabled by default. Generally useful for areas where nobody should enter.
  - **Include Strangers**: Adds bookmarks when a face is determined to be a stranger. Generally useful for secured areas where only known people should be.
  - **Include Enrolled**: Adds bookmarks when a face is determined to be a known person.
  - **Include Concerns and Threats**: Adds bookmarks when a face is determined to be a known concern or threat.
  - **Include Smile Activation**: Requires smile activation to trigger recognition.
- **Advanced**: Clicking on the **Advanced** button opens the following window:



SAFR Event Name	Digifort Event	Bookmark Title	Color
Threat Person	SAFR Threat Person	SAFR Threat Person Detected	🟥
Person of Concern	SAFR Person of Concern	SAFR Person of Concern Detected	🟨
Stranger	SAFR Stranger	SAFR Stranger Detected	🟪
Unrecognizable Face	SAFR Unrecognizable Face	SAFR Unrecognizable Face Detected	🟪
Recognized Unnamed Person	SAFR Recognized Unnamed Person	SAFR Recognized Unnamed Person Dete	🟦
Recognized Named Person	SAFR Recognized Named Person	SAFR Recognized Named Person Detecte	🟩
Smile Activation	SAFR Smile Activation	SAFR Smile Activation Detected	🟩
<TYPE> Person	SAFR <TYPE> Person	SAFR <TYPE> Person Detected	🟩

If the Digifort events are disabled (gray; no matching Digifort event), refer to Digifort documentation to create Digifort events. Once the events are created, open the Digifort Advanced Settings dialog and, from the menu for the associated SAFR event, select the Digifort event. Once the Digifort events are created and matched, the events are enabled (black). For information on creating events, refer to Digifort documentation.

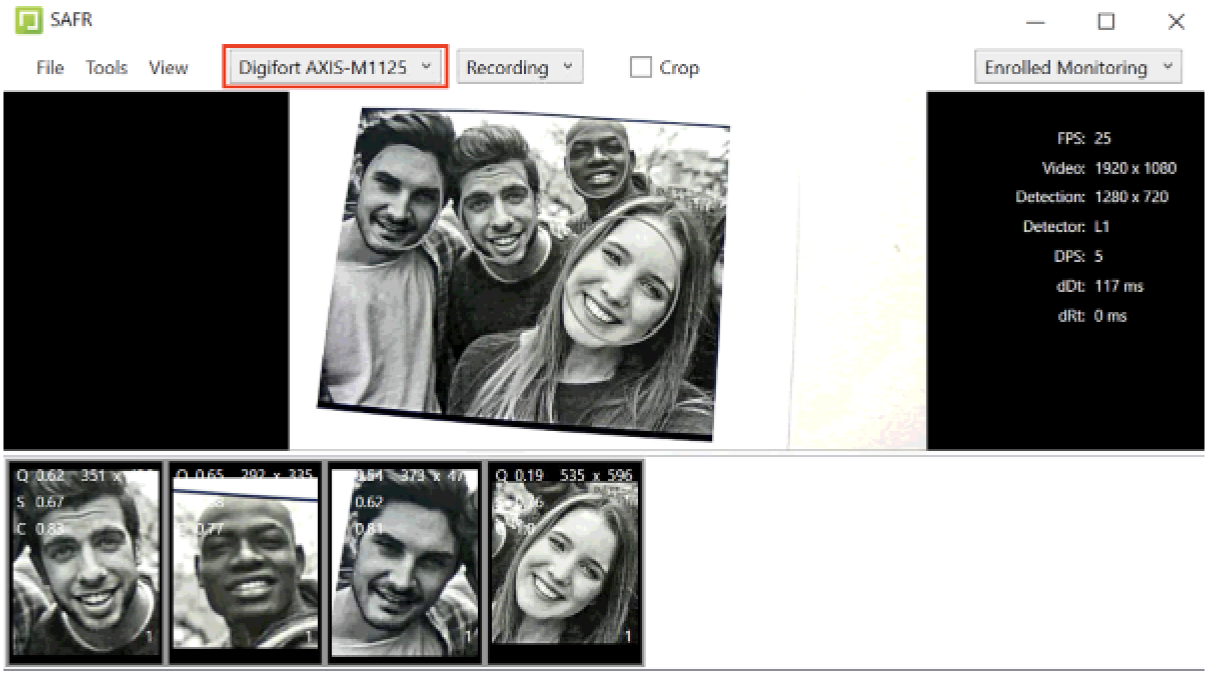


SAFR Event Name	Digifort Event	Bookmark Title	Color
Threat Person	SAFR Threat Person	SAFR Threat Person Detected	🟥
Person of Concern	SAFR Person of Concern	SAFR Person of Concern Detected	🟨
Stranger	SAFR Stranger	SAFR Stranger Detected	🟪
Unrecognizable Face		SAFR Unrecognizable Face Detected	🟪
Recognized Unnamed Person	SAFR Recognized Unnamed Person	SAFR Recognized Unnamed Person Dete	🟦
Recognized Named Person	SAFR Recognized Named Person	SAFR Recognized Named Person Detecte	🟩
Smile Activation	Smile	SAFR Smile Activation Detected	🟩
<TYPE> Person	SAFR <TYPE> Person	SAFR <TYPE> Person Detected	🟩
t3 Person	t3 Person	SAFR t3 Person Detected	🟩
Developer Person	SAFR Developer Person	SAFR Developer Person Detected	🟩
Dangerous Person	SAFR Dangerous Person	SAFR Dangerous Person Detected	🟥

This affects the titles given to bookmarks created from the respective events.

Note: Each event type is only created if enabled in the SAFR Digifort Preferences tab.

106.2 Connect and Use Cameras and Video Feeds



```
FPS: 25
Video: 1920 x 1080
Detection: 1280 x 720
Detector: L1
DPS: 5
dDt: 117 ms
dRt: 0 ms
```

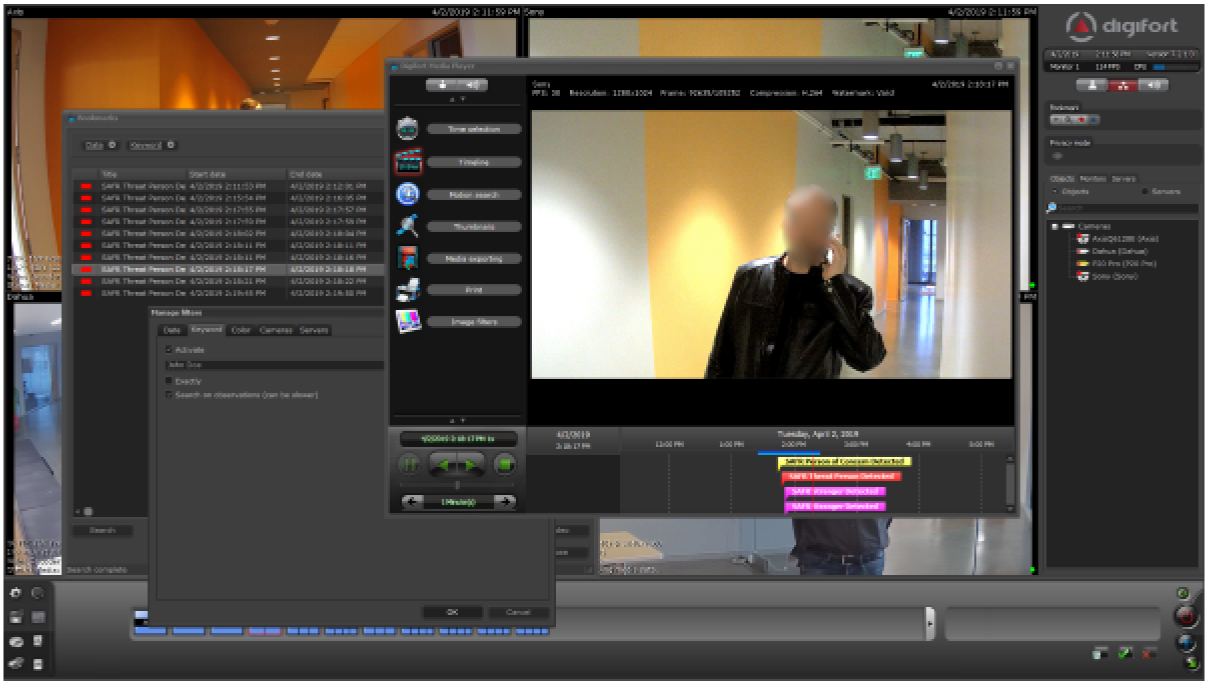
In the SAFR Desktop Client, view the video feed for the camera selected from the camera selection menu.

The menu shows the cameras available from the Digifort servers. To enable the row at the bottom of the screen that isolates individual faces, click **View > Detection List**.

106.3 Digifort Bookmarks

Digifort creates bookmarks to help locate important events. Bookmarks are populated with *Person Type*, *ID Class*, and *Name*. They can also provide more detailed information with even more search attributes, such as *Age* and *Gender*.

The following illustration shows how bookmarks can be used to review important events, such as the detection of a stranger tailgating behind a registered user.



To view Digifort Bookmarks, do the following:

- Click the **Bookmark** icon on right side panel.
- Set a date range or other criteria, and click **Search**.
- Click a bookmark of interest.
- To play the video, click **Video**.

106.4 SAFR Identities

To add people through the SAFR Desktop Client from an image or video file, do the following:

1. Open the Desktop Client.
2. Click **File > Import Faces**.
3. Select the image.
   - For an image, each recognized face is enclosed by a box, and you have the option to type a name.
   - For a video, each recognized person is learned automatically as long as the faces meet the minimum criteria for recognition.
4. If faces are not learned, check the settings in the Detection and Recognition tabs under Preferences to ensure faces meet minimum criteria.
   - Detection > Minimum searched face size
   - Recognition > To allow identification

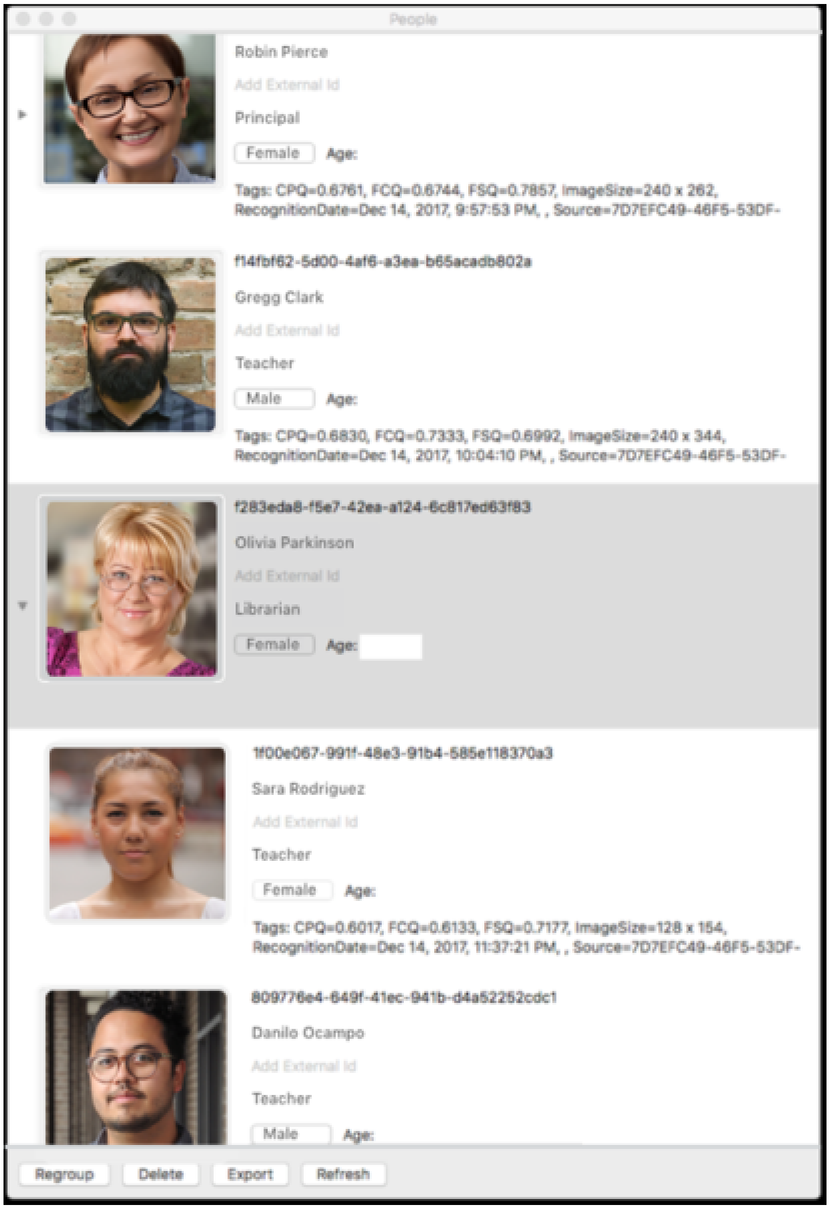
Warning: Reducing detection and recognition settings lowers the quality of the reference face and negatively impacts recognition. It is preferable to increase the quality of your sources than to lower the criteria for learning.

Warning: Users added to SAFR are not synchronized with Digifort; these users exist only in SAFR.

It may be desirable to edit people properties to control which events get triggered when that person is recognized. For example, setting a person's *ID Class* to *Concern* or *Threat* triggers the respective alarms. The most important people attributes are *Name*, *Image*, *Person Type*, and *ID Class*.

Name, *Image*, and *Person Type* should be edited through SAFR. *Person Type* defines a person's role (e.g. staff or visitor), while the *ID Class* defines the risk level (No-Concern, Concern, or Threat). *Person Type* and *Image* can be edited in the Desktop Client by changing the *Person Type* on the People screen.

ID Class and all other attributes of a person are also edited within SAFR People dialog, accessed through the SAFR Desktop Client **Tools** menu. All identities are created by default with an *ID Class* of *No Concern*. To edit a person's *ID Class*, open the People window from the SAFR Desktop Client **Tools** menu as follows:



Robin Pierce
Add External Id
Principal
Female Age:
Tags: CPQ=0.6761, FCQ=0.6744, FSQ=0.7857, ImageSize=240 x 262, RecognitionDate=Dec 14, 2017, 9:57:53 PM, , Source=7D7EFC49-46F5-53DF-

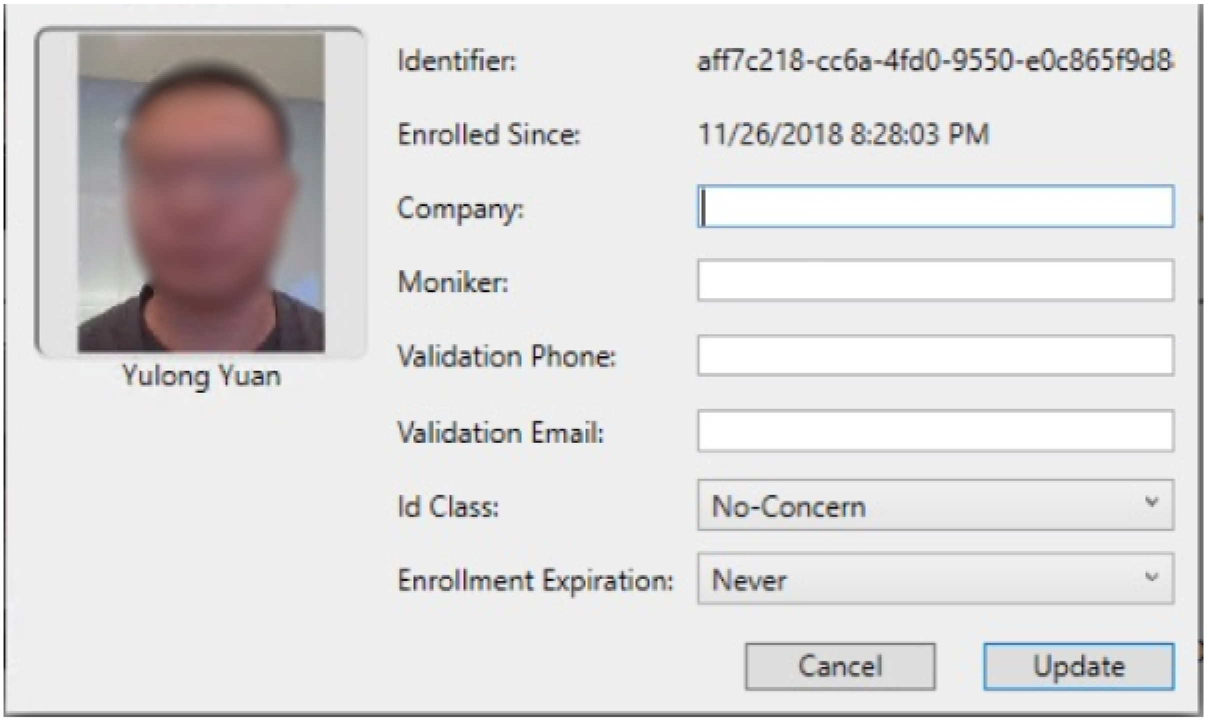
f14fbf62-5d00-4af6-a3ea-b65acadb802a
Gregg Clark
Add External Id
Teacher
Male Age:
Tags: CPQ=0.6830, FCQ=0.7333, FSQ=0.6992, ImageSize=240 x 344, RecognitionDate=Dec 14, 2017, 10:04:10 PM, , Source=7D7EFC49-46F5-53DF-

f283eda8-f5e7-42ea-a124-6c817ed63f83
Olivia Parkinson
Add External Id
Librarian
Female Age:

1f00e067-991f-48e3-91b4-585e118370a3
Sara Rodriguez
Add External Id
Teacher
Female Age:
Tags: CPQ=0.6017, FCQ=0.6133, FSQ=0.7177, ImageSize=128 x 154, RecognitionDate=Dec 14, 2017, 11:37:21 PM, , Source=7D7EFC49-46F5-53DF-

809776e4-649f-41ec-941b-d4a52252cdc1
Danilo Ocampo
Add External Id
Teacher
Male Age:

The *Person Type* and *Name* can be edited by clicking the respective fields on the People screen. To edit *ID Class*, double-click the person, and choose the desired value from the ID Class menu in the People Edit dialog as shown in the following graphic:



Yulong Yuan

Identifier: aff7c218-cc6a-4fd0-9550-e0c865f9d8

Enrolled Since: 11/26/2018 8:28:03 PM

Company:

Moniker:

Validation Phone:

Validation Email:

Id Class: No-Concern

Enrollment Expiration: Never

106.5 SAFR Events

You can use SAFR to generate events. When enabled on the Events Preferences page, events occur when a person appears in the view of a connected camera. When a person appears in the camera's view, they're immediately assigned an *ID Class* attribute, although that *ID Class* may change if the system successfully recognizes them and assigns them a more appropriate *ID Class*. The types of *ID Classes* are listed below.

- Unrecognizable: A face was detected but it wasn't of sufficient quality to attempt recognition.
- Stranger. The person has been recognized, but they're not in Genetec's cardholder database nor in SAFR's Person Directory.
- Registered person without a name.
- Registered person with a name.
- Registered person marked as a *Concern*.
- Registered person marked as a *Threat*.

The following table lists the available events that are SAFR makes available to Digifort.

Event Message	Id Class	Named	Person Type	Condition	People Attributes
Unrecognizable face detected	N/A	N/A	N/A	Face detected but insufficient information for recognition	idClass="unidentified"

Event Message	Id Class	Named	Person Type	Condition	People Attributes
Stranger detected	Stranger	N/A	N/A	Face detected but not found in registered people	idClass=“stranger”
Registered person detected without name	Normal	No	None	Registered person without name or person type assigned	idClass=“noconcern” && personType="" && name=""
Registered person detected with name <name>	Normal	Yes	None	Registered person with name but no person type	idClass=“noconcern” && personType="" && name=<name>
Registered person detected of type <personType>	Normal	No	Defined	Registered person with person type but no name	idClass=“noconcern” && personType=<personType> && name=""
Registered person detected of type <personType> with name <name>	Normal	Yes	Defined	Registered person with person type and name	idClass=“noconcern” && personType=<personType> && name=<name>
Concern person detected without a name	Concern	No	None	Same as above for Concern	idClass=“concern” && personType="" && name=""
Concern person detected with name <name>	Concern	Yes	None	Same as above for Concern	idClass=“concern” && personType="" && name=<name>
Concern person detected of type <personType>	Concern	No	Defined	Same as above for Concern	idClass=“concern” && personType=<personType> && name=""
Concern person detected of type <personType> with name <name>	Concern	Yes	Defined	Same as above for Concern	idClass=“concern” && personType=<personType> && name=<name>
Threat person detected without a name	Threat	No	None	Same as above for Threat	idClass=“threat” && personType="" && name=""

Event Message	Id Class	Named	Person Type	Condition	People Attributes
Threat person detected with name <name>	Threat	Yes	None	Same as above for Threat	idClass="threat" && personType="" && name=<name>
Threat person detected of type <personType>	Threat	No	Defined	Same as above for Threat	idClass="threat" && personType=<personType> && name=""
Threat person detected of type <personType> with name <name>	Threat	Yes	Defined	Same as above for Threat	idClass="threat" && personType=<personType> && name=<name>

106.5.1 SAFR Video Processing Modes

SAFR has different video processing modes that control what events are generated. Below is a short summary of the modes most relevant to Digifort integration. For a complete description, see Connect to a Video Feed in the *SAFR Documentation*.

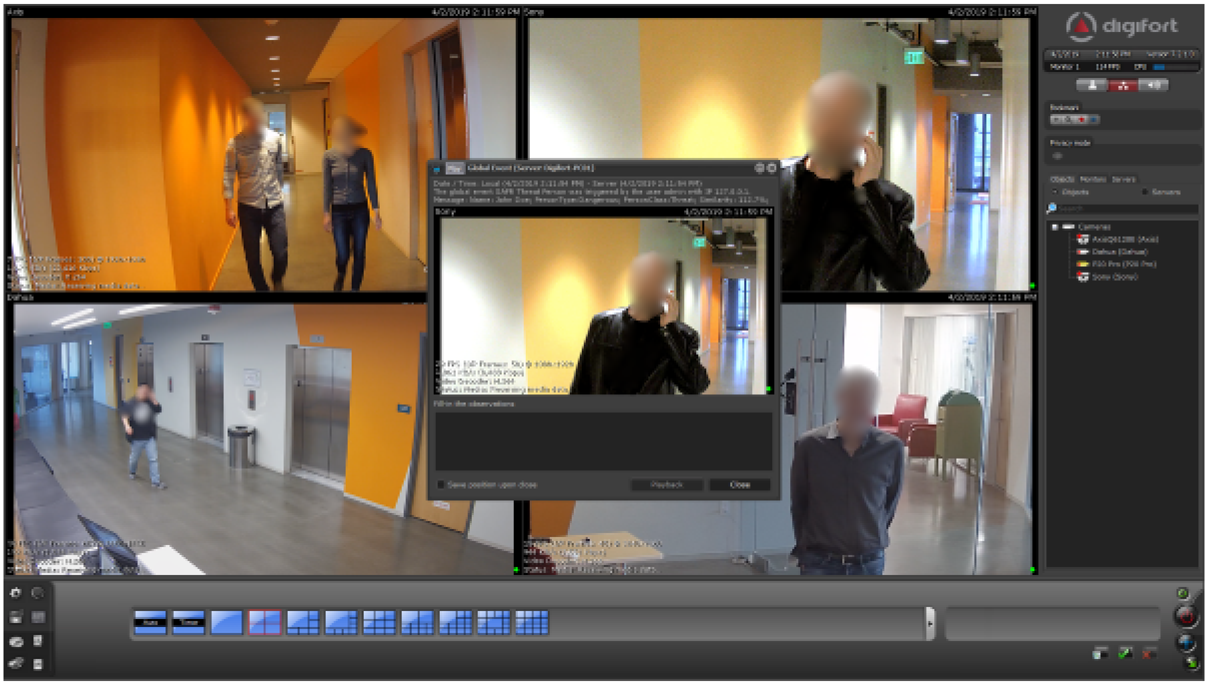
- **Secure Access**: Only triggers events when cardholders and people registered in SAFR's Person Directory are identified with a high degree of certainty. This mode is useful when the system is being used to manage physical access. (i.e. When the system is responsible for unlocking doors for authenticated people.)
- **Secure Access with Smile**: Similar to *Secure Access* mode, except that registered people must smile in order to cause the system to grant them access.
- **Enrolled Monitoring**: Similar to *Secure Access* mode, but events are triggered at a lower recognition confidence level.
- **Enrolled and Stranger Monitoring**: Similar to *Enrolled Monitoring* mode, but events are also triggered for strangers.

106.5.2 Alarms and Notifications

You can also use SAFR to view recognition events. Recognition events occur when a known, unknown, or unrecognized person appears in the view of a camera. The types of recognized persons are:

- Unrecognizable: A face was detected but it wasn't of sufficient quality to attempt recognition.
- Stranger.
- Registered person without a name.
- Registered person with a name.
- Registered person marked as a *Concern*.
- Registered person marked as a *Threat*.

There are several different combinations of these conditions that are triggered. The following graphic shows multiple events populated in the Digifort alerts panel:



106.6 Troubleshooting Tips

Note: When closing SAFR, use the **Quit SAFR** option on the **File** menu. Closing SAFR using the Window Close button will cause you to lose the SAFR state settings and connected cameras for that window.

- If detection or recognition results in few faces found or recognized, check that the Digifort video feeds are of a sufficiently large frame size.
- If Digifort cameras do not appear in the SAFR Desktop Client, make sure you have added cameras to Digifort as described in Connect Your Cameras to Digifort.
- If events are not being triggered, ensure the correct SAFR video processing mode is selected.

107 SAFR Genetec Integration Guide

Integrated SAFR Genetec is only available on Windows.

Integrating SAFR's facial recognition and analysis capabilities into Genetec enables you to use SAFR's video feed information overlays within Genetec camera video feeds, thus making it much easier to quickly and accurately separate unknown people from authorized people from known threats. You'll also have immediate access to additional infomation such as age, gender, sentiment, name, company, known associates, or any other configurable information you want to create.

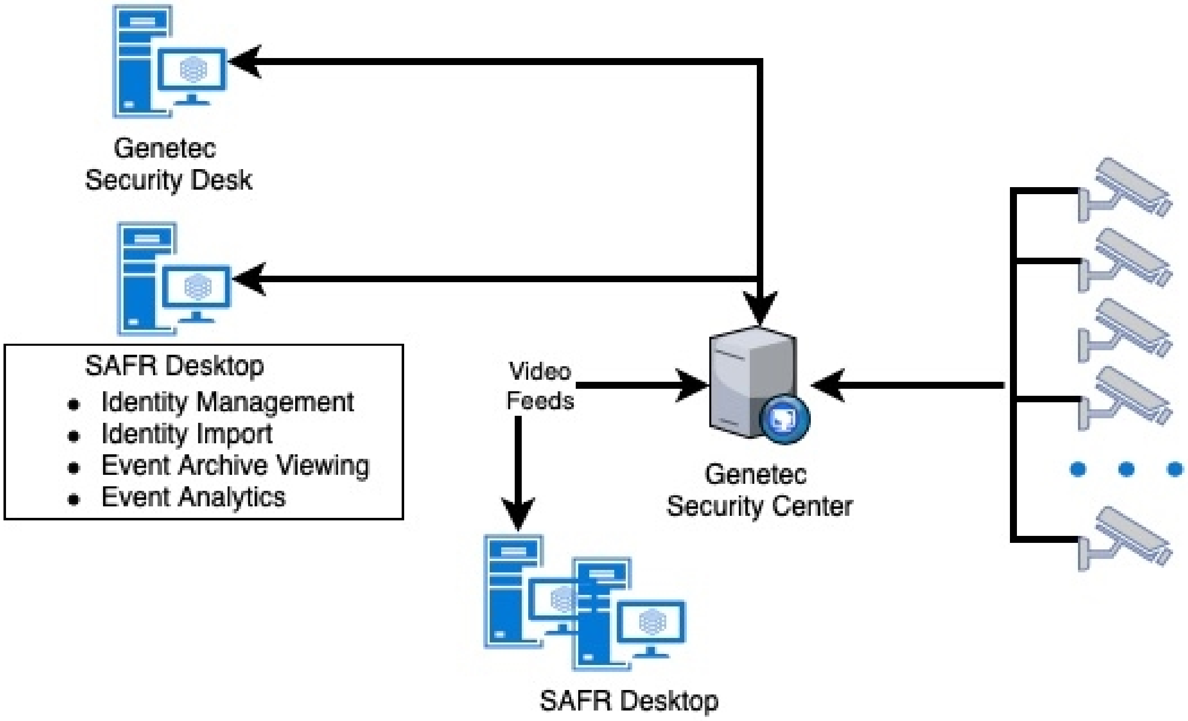
Note: SAFR has the native capability to detect age, gender, and sentiment, while other information needs to be manually entered by an operator.

Integrating the two systems also allows SAFR's information about individuals to trigger Genetec alerts and other actions within the Genetec system. Genetec's metadata within bookmarks are enriched with SAFR's additional information, allowing you to more easily find relevant bookmarks.

107.1 Integration Overview and Requirements

A typical deployment requires the following:

- A machine running Genetec Security Center.
- A machine running Genetec Security Desk and Genetec Config Tool.
- One or more machines running the SAFR Desktop Client to process videos.
- If you're doing a SAFR local deployment, you'll also need a machine running SAFR Server. SAFR Server can run on the same machine as one of the Desktop Clients, provided the host machine meets the system requirements.



Cameras are connected to the Genetec Security Center. The SAFR Desktop Client(s) can then connect to the Genetec Security Center to perform analysis of the video and add overlays. Depending on the number of cameras you need, one or more machines may be required to run the SAFR Desktop Client, each processing multiple video feeds. The Desktop Client processes the video and returns information to Genetic to overlay the video feeds and generate events. The Desktop is also used to perform various management activities.

107.1.1 System Requirements

Genetec has the following system requirements:

- One machine running Genetec Security Center Version 5.7 or later.
- One machine running Genetec Security Desk and Genetec Config Tool.
- Each machine running a Genetec product must meet the following system requirements:
  - Windows 10.
  - Additional system requirements as described in the Genetec documentation.

SAFR has the following system requirements:

- Each machine running the SAFR Desktop Client must meet the following requirements:
  - Windows 10.
  - The Desktop Client must be version 1.3.228 or later.
  - Genetec Security Center SDK for your version of Genetec Security Center must be installed.
  - Additional system requirements as described on the SAFR system requirements page.
- Local SAFR deployments require at least one machine running SAFR Platform 1.3 or later.
- Each machine running SAFR Server must meet the following requirements:
  - Windows 10.
  - Additional system requirements as described on the SAFR system requirements page.

107.1.2 Licensing and the Genetec Part Number

An accompanying Genetec part number must be added to your Genetec connection license. Do the following to discover and add the Genetec part number:

1. Go to the Genetec Portal and sign in using your Genetec credentials.
2. In the applications section, search for *SAFR*. From the results, click *SAFR Facial Recognition*.
3. On the **SAFR Facial Recognition Solution Details** page, in the right column, the *Genetec Part Number* is displayed.
4. Contact Genetec and have them add the part number to your license. You need a quantity of the part number equal to the number of cameras SAFR will be processing plus one additional license for the metadata channel SAFR creates. In other words, if SAFR will be processing cameras, then you need quantity of the part number added to your license.

You'll need the following licenses: each Genetec camera where SAFR face detection and recognition is used, you'll need:

- A Genetec connection license with the accompanying Genetec part number is required for each connected Genetec camera.
- One additional Genetec connection license for the metadata channel SAFR creates.
- A SAFR license for each camera is required

For example, if you have 300 cameras but only need face detection on 30 cameras at a time, then you would obtain a 31 connection license from Genetec and a 30 camera license from RealNetworks. Having a 31 connection Genetec license does not mean you are limited to face detection on a fixed set of 30 cameras. At any time, you can choose to connect the SAFR Desktop Client to a different camera. You may have cameras in your parking garage that you were not previously monitoring with SAFR recognition. You can use a few of your licenses that are connected to other cameras to connect to garage cameras instead.

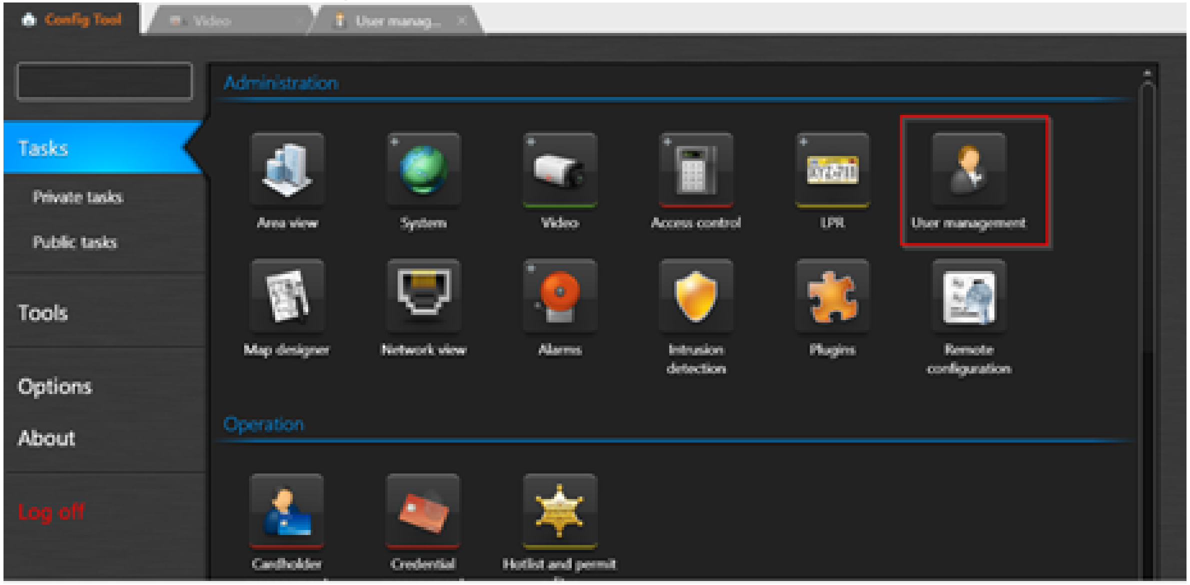
107.2 Install and Configure the Genetec Security Center

1. Download the latest version of Genetec Security Center from the Genetec Portal.
2. Run the installer. For details about which install options to select, see the Security Center Installation and Upgrade Guide.

107.2.1 Create a SAFR User

To create a user with the permissions that SAFR will require, do the following:

1. Open the Genetec Config Tool.
2. Click **Tasks > User Management**.



3. Create a new user (with a username of, for example, *SAFR*) with the following permissions:

All privileges

- Application privileges
  - Log on using the SDK

Administrative privileges

- Physical entities
  - View camera properties

Access control management

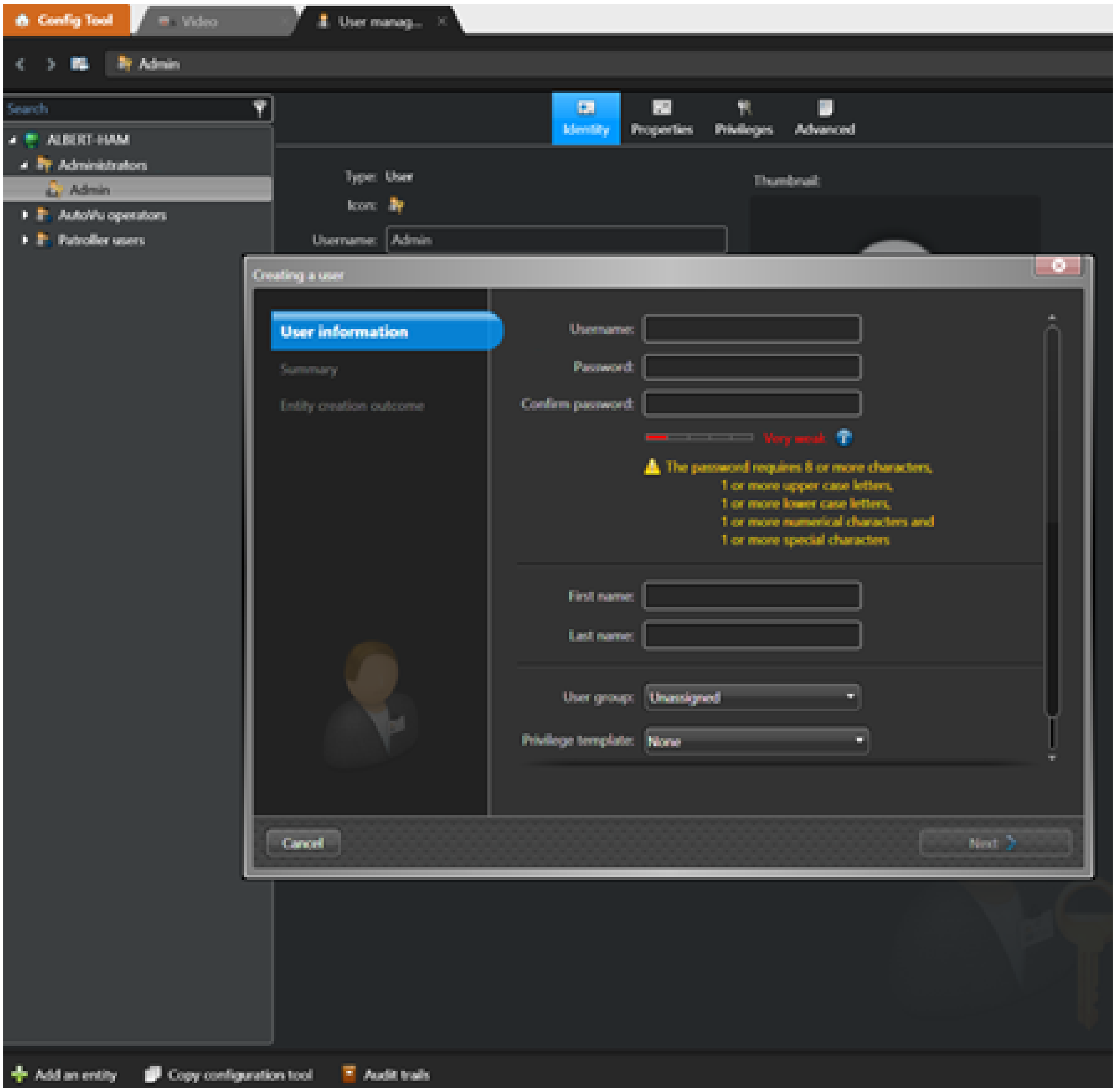
- View cardholder group properties
- View cardholder properties
- View visitor properties

System management

- View general settings
  - Modify custom events

Action Privileges

- Cameras
  - View live video
  - Add bookmarks



Type: User

Icon:

Username: Admin

Username:

Password:

Confirm password:

Very weak

The password requires 8 or more characters,
1 or more upper case letters,
1 or more lower case letters,
1 or more numerical characters and
1 or more special characters

First name:

Last name:

User group: Unassigned

Privilege template: None

107.2.2 Add Permissions for Event-to-Archive Actions

In order to create Event-to-Actions in the Genetec Config Tool, one or more of the following Action permissions must also be added to the SAFR user created in the previous section. Only those actions you want to trigger with SAFR events are needed:

All privileges

- Action privileges
  - Set threat level
  - Cameras
    - Protect video from deletion
    - Save/modify/print snapshots
  - Access control
    - Doors
      - Explicitly unlock doors
      - Override unlock schedules
    - Elevators

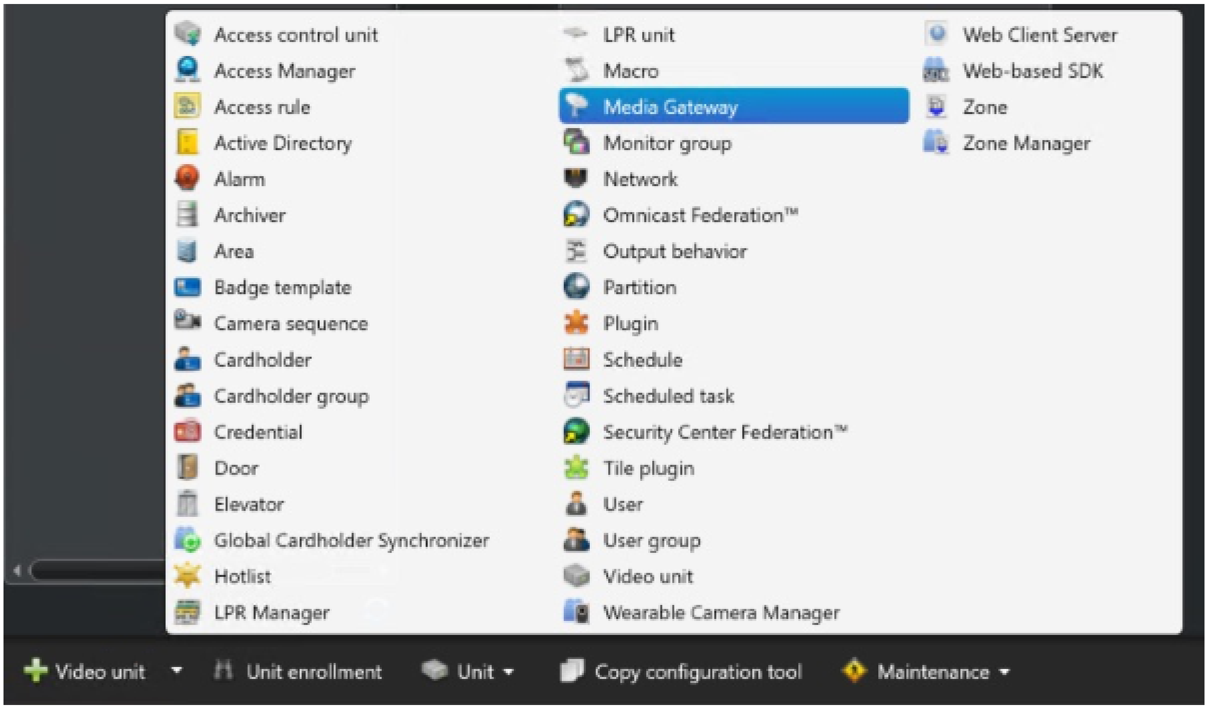
        - Override elevator schedules
    - Alarms
        - Trigger alarms
    - Users
        - Send a message
        - Send an email
        - Send/clear task
    - Macros
        - Execute macros
    - Zones
        - Arm/disarm zones
    - Areas
        - Modify people count

107.2.3 Set Minimum Cardholder Image Size

Do the following:

1. Open the Genetec Config Tool.
2. Open **Tasks > Administration > Access Control > General Settings**.
3. Set *Maximum Picture File Size* to 128k or larger.

107.2.4 Configure the Media Gateway



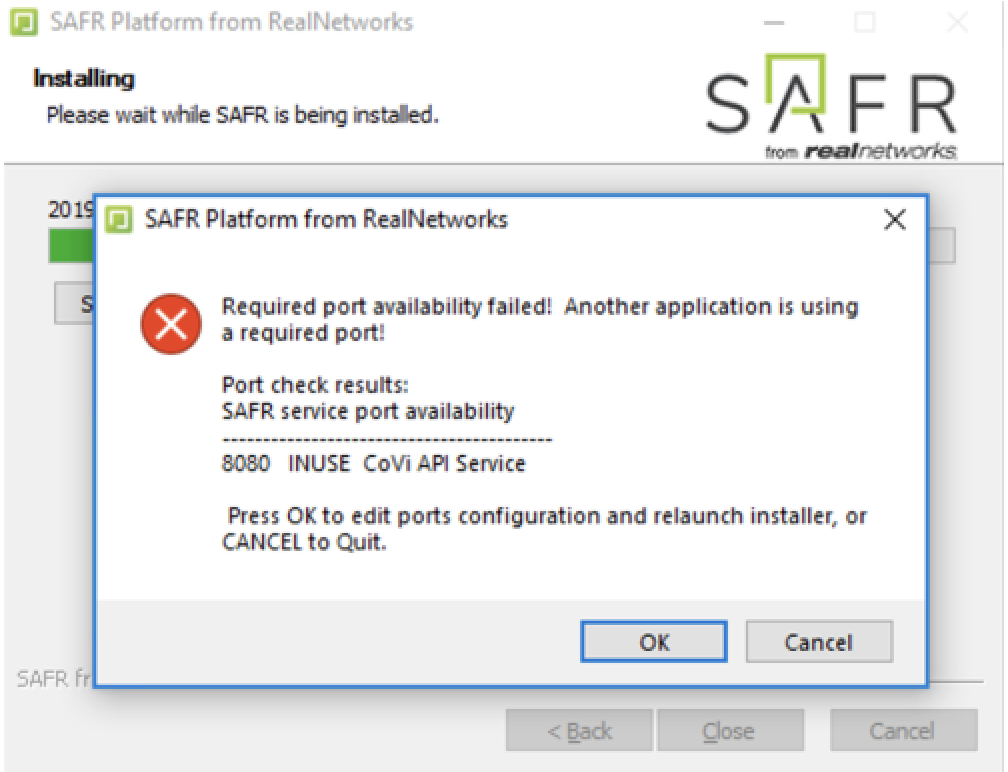
Do the following:

1. Open the Genetec Config Tool.
2. Open **Tasks > Administration > Video**.
3. Click the arrow next to the Video Unit button in the bottom left corner, and select **Media Gateway**.
4. Click **Next**, and in the *Create Media Gateway* wizard, click **Create**. Accept the default values; no changes are needed.
5. Select **Media Gateway**, and click the **Properties** task.
    - This adds a *Media Gateway* entry in the list on left side.

6. Determine the user to be granted access to the media gateway.
    - This can be the SAFR user or a different user; we recommend using the same SAFR user unless you already have one configured to use the Media Gateway.
    - This user does not need to have specific permissions. The permissions for media gateway are granted to this user in the next step.
7. To add this user to the **Accessible To** section, click the + icon. In the bottom right, click **Apply** to save the changes.
8. When prompted, enter a password for the user you are adding.
    - This password can be the same as the user's normal password or it can be different.
9. Save the *username* and *password*.
    - This is the password that must be used in the Media Gateway credentials fields in the SAFR preferences window.

107.3 Install and Configure SAFR

1. On the machine(s) where you plan to install the SAFR Desktop Client, install the Genetec SDK from the Genetec Portal.
2. Go to the SAFR Download Portal.
3. If you're doing a cloud deployment, download and install Windows SAFR Desktop. Make sure to select the Genetec Security Center install option.
4. If you're doing an on-premise deployment, download and install Windows SAFR Platform. Make sure to select the Genetec Security Center install option.
    - When installing the SAFR Platform, the default SAFR port assignments sometimes conflict with other software port assignments. If a port conflict occurs, the error message shown below will pop up in the middle of your installation.


SAFR Platform from RealNetworks
SAFR
from realnetworks
2019
S
SAFR fr
< Back
Close
Cancel
SAFR Platform from RealNetworks
OK
Cancel

Installing

Please wait while SAFR is being installed.

Required port availability failed! Another application is using a required port!

```
Port check results:
SAFR service port availability
------------------------------------------
8080  INUSE  CoVi API Service
```

Press OK to edit ports configuration and relaunch installer, or CANCEL to Quit.

If this happens, do the following:

1. Click **OK** to edit port configurations.
2. **Notepad** will open, displaying the *safrports.conf* file.
3. Edit any conflicting ports to new values. (e.g. CoviHTTP=18080)

4. Save and exit **Notepad**.

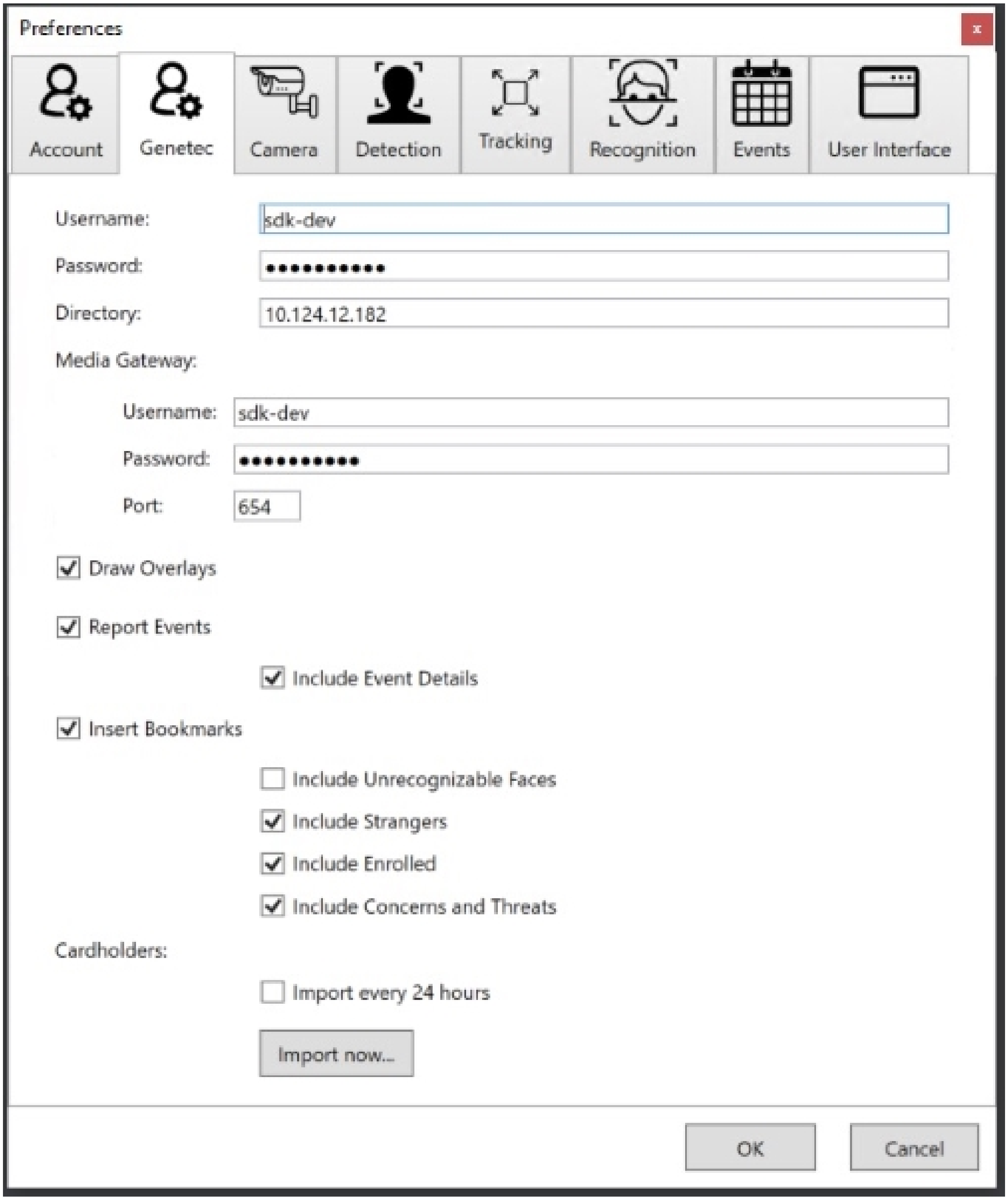
The Platform installer will then restart and the new port values will be used. You can find the modified *safrports.conf* file at `C:\Program Files\RealNetworks\SAFR\`.

After the installation finishes, two icons will appear on your desktop: one labeled *SAFRActions* and another labeled *SAFR*. *SAFRActions* launches SAFR Actions, while *SAFR* launches the Desktop Client. The SAFR Server (when installed as part of a local deployment) automatically runs as a collection of background services.

Immediately following installation, the installer opens the Desktop Client and prompts you to log in with your SAFR Account credentials. Make sure to log in; it's important in acquiring the SAFR license.

107.3.1 Connect SAFR to Genetec

1. Within your SAFR Desktop Client, select **Tools->Preferences->Genetec**.
   Note: If the Genetec preference tab is not showing, it means that the Genetec SDK was not properly installed on your machine.



Username: sdk-dev

Password: ●●●●●●●●●●●

Directory: 10.124.12.182

Media Gateway:

Username: sdk-dev

Password: ●●●●●●●●●●●

Port: 654

☑ Draw Overlays

☑ Report Events

☑ Include Event Details

☑ Insert Bookmarks

☐ Include Unrecognizable Faces

☑ Include Strangers

☑ Include Enrolled

☑ Include Concerns and Threats

Cardholders:

☐ Import every 24 hours

2. Enter the following information in the Genetec preferences tab.
   - **Username**: Enter the SAFR user you created earlier.
   - **Password**: Enter the *Password* you created for the SAFR user.
   - **Directory**: IP address of the server running the Genetec Security Center server.
   - **Media Gateway**: Used for acquiring video streams.
     - **Username**: Enter the SAFR user you created earlier.
     - **Password**: Enter the *Password* you created for the SAFR user.
     - **Port**: Enter the port on which to connect to the Media Gateway. You can use the default value of `654` unless that would create a port conflict.

This should cause your SAFR system to establish a connection with the Genetec system.

To verify that your SAFR system successfully connected to the Genetec system, do the following:

1. On the SAFR Desktop Client, open **Tools -> Preferences -> Camera**.
2. Cameras connected to Genetec system should be visible.
3. All cameras connected to Genetec have the *Genetec* prefix in their names.

107.4 Troubleshooting

107.4.1 How do I Resolve a Certificate Registration Error when Logging in from SAFR to Genetec?

This error is caused by a mismatch between the SAFR Genetec certificate and the Genetec Security Center license. SAFR builds have either a Genetec production certificate or a development certificate. The production certificate can be used only with Security Center installations that use a production or demonstration license. The development certificate can be used only with Security Center installations that use a development license.

Here are some steps you can take to try to diagnose the issue:

1. Use the Genetec Config Tool to connect to the Genetec Security Center server.
2. Click **About** on the left side.
3. Click the **Certificates** tab.
4. If you see a line that says, "Generic certificate for developers" then the Security Center server is using a developer license. You must use a SAFR build that uses a developer certificate. Builds with developer certificate are available only from SAFR build farm and should be used only by developers.
5. If that line is not present, then Security Center is using a production or demonstration license. You must use a SAFR build that uses a production certificate. Download SAFR build with production certificate from the SAFR Download Portal.
   - Click on the **Purchase Order** tab. Production or demonstration licenses must also have a license for SAFR attached to it. There should be a line with `Part #GSC-1SDK-RealN- FaceRec`. The quantity must be equal to or greater than the number of cameras that SAFR will be processing.

107.4.2 How do I Resolve a Connection Error when Logging in from SAFR to Genetec?

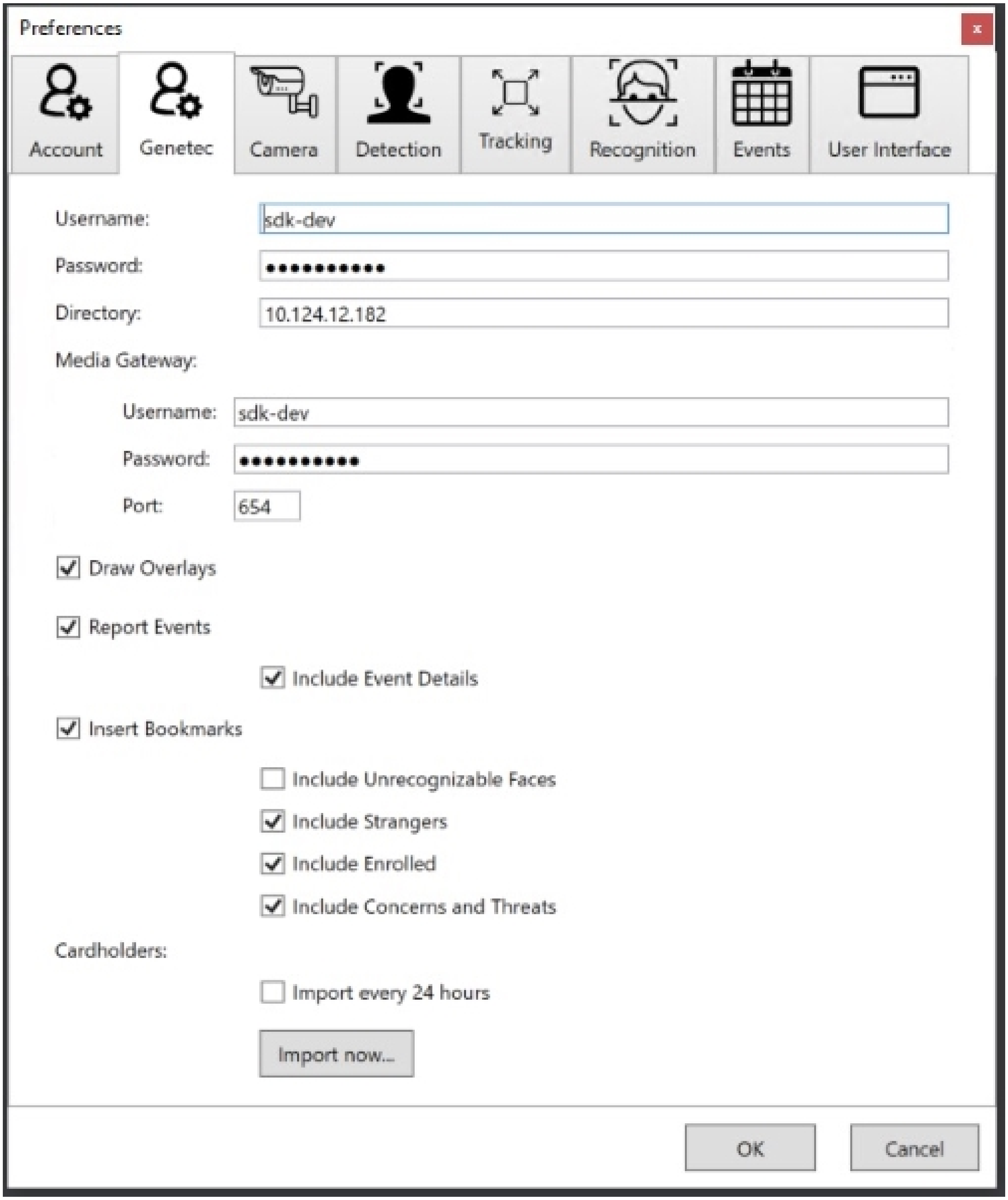
There can be many different causes for a Connection Timeout error from SAFR. However, if you are in a situation where this consistently happens and no cameras are connecting, then doing the following will most likely resolve the error:

1. Connect to the Security Center server using the Genetec Config Tool.
2. Go to the **Video** task.
3. In the left pane, right-click on the **Media Gateway** role.
4. Select the **Maintenance->Deactivate** role.
5. After the role turns gray, right-click on it again.
6. Select the **Maintenance->Activate** role.
7. The Media Gateway will go through a startup routine. It will turn red, yellow, and eventually white.
8. After it turns white, try connecting again.

108 SAFR Genetec Operation Guide

108.1 SAFR Genetec Preferences

You can set several Genetec-specific preferences by opening the SAFR Desktop Client and clicking on **Tools -> Preferences -> Genetec**.



Username: sdk-dev

Password: ●●●●●●●●●●●

Directory: 10.124.12.182

Media Gateway:

Username: sdk-dev

Password: ●●●●●●●●●●●

Port: 654

☑ Draw Overlays

☑ Report Events

☑ Include Event Details

☑ Insert Bookmarks

□ Include Unrecognizable Faces

☑ Include Strangers

☑ Include Enrolled

☑ Include Concerns and Threats

Cardholders:

□ Import every 24 hours

- **Username**: Person with the credentials to connect the SAFR system to the Genetec Security Center

Server.

- **Password**: Password of a person with credentials to connect the SAFR system to a Genetec Security Center Server.
- **Directory**: IP address or hostname of the Genetec server.
- **Media Gateway**: Used for acquiring video streams.
  - **Username**: Person with the credentials to connect the SAFR system to a Genetec Security Center Server.
  - **Password**: Password of a person with credentials to connect the SAFR system to a Genetec Security Center Server.
  - **Port**: The port at which SAFR will connect to the Genetec Security Center Server. The default is 654.
- **Draw Overlays**: Enables the drawing of ovals, names, and other details within Genetec camera video stream. The overlays match what would be shown in the SAFR Desktop Client, so SAFR settings affecting SAFR overlays also affect what is drawn in Genetec.
- **Report Events**: Enables reporting SAFR events to Genetec. If this setting isn't checked, *Include Event Details* is automatically greyed out.
  - **Include Event Details**: When enabled, all of the technical details of the event are attached to events. This option is especially useful if an operator uses macros to handle events for decision making.
- **Insert Bookmarks**: When enabled, bookmarks are added to camera video streams events. This allows operators to search videos for events or recognized people names. Care should be taken as to what to include since encountering many faces can cause numerous bookmarks to be created. When this box isn't checked, the 4 children settings below are all greyed out.
  - **Include Unrecognizable Faces**: When enabled, adds bookmarks when a face is detected but SAFR does not have enough information to determine if they are a stranger or a known person. This can result in an overwhelming number of bookmarks, so it's disabled by default. However, this setting can be useful when monitoring areas with very few people.
  - **Include Strangers**: When enabled, adds bookmarks when a face is recognized and determined to be a stranger. This option is generally useful for secured areas where only known people should be.
  - **Include Enrolled**: When enabled, adds bookmarks when a face is recognized and determined to be a known person.
  - **Include Concerns and Threats**: When enabled, adds bookmarks when a face is recognized and determined to be a known concern or threat.
- **Cardholders**
  - **Import Every 24 Hours**: When enabled, all the Genetec cardholders not already in SAFR's Person Directory are imported and registered to SAFR every 24 hours.
  - **Import now. . .** : Clicking this causes all the Genetec cardholders not already in SAFR's Person Directory to be imported and registered to SAFR.

108.2 Connect and Use Cameras and Video Feeds

1. To connect cameras to Genetec, you need to add the cameras to the Genetec Video Archiver using the Genetec Config Tool. For details, please see the Genetec Security Center Administrator Guide.
2. After a camera has been added to the Video Archiver, it should be displayed as a Genetec camera in SAFR. If it's not, try closing and re-opening the SAFR Desktop Client.
   To get SAFR video feed overlays to be displayed on Genetec camera feeds, do the following:
   1. Open the SAFR Desktop Client.
   2. Select the Genetec version of the camera from the menu in the main windows (upper left). The word "Genetec" will be the first part of the camera name.
   3. After the client has successfully connected to the Genetec camera, video from the Genetec camera is displayed in the SAFR Desktop Client video feed window.
   4. Open the Genetec Security Desk.
   5. Go to the **Monitoring** Task.

6. Drag and drop a camera from the left side into one of the tiles in the middle.
7. The camera feed should appear and show the same video feed overlays that are in SAFR.

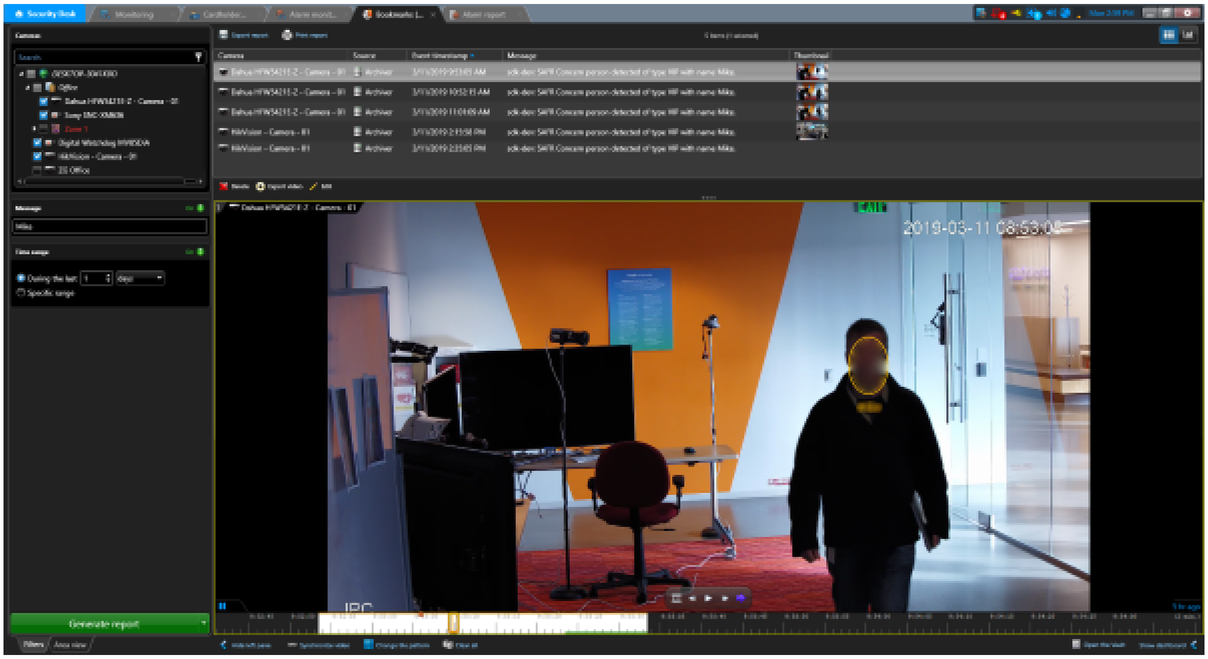
To connect additional cameras:

1. Open another instance of the Desktop Client by selecting **File > New** on the client.
2. Repeat steps 2-6 above.
3. You can keep repeat this procedure to add overlays to as many video feeds as desired.
   Note: Most machines can only support up to 16 video feeds. If you want to connect more feeds than that, you'll need to install the SAFR Desktop Client on additional machines.

By default, the SAFR Desktop Client operates in the *Enrolled Monitoring* video processing mode and generates events and bookmarks into the Genetec system for every registered person. If you want a different mode for a given camera, choose a different mode from that camera's *Camera* window **Mode Selector** menu.

108.3 Bookmarks

SAFR enhances Genetec's bookmarks so that they provide readily accessible additional information, making the bookmarks much more useful. Bookmarked video contains the video feed overlays and enhanced people-related data described above. In addition, the bookmarks themselves are populated with the attributes for each person in the camera view so that searching bookmarks is more fruitful. All bookmarks automatically contain the *Person Type*, *ID Class*, and *Name* of each recognized person, and additional attributes such as *Age* and *Gender* will be included within the bookmarks, if such additional attributes are known. The image below shows how bookmarks can be used to review important events. The yellow overlay indicates that the person is a concern.



108.4 Genetec Cardholders and SAFR Identities

Genetec cardholders can be registered to SAFR by doing the following:

1. Increase the Genetec Security Center setting for thumbnail size to make sure SAFR has access to high quality images to use for face recognition.
2. On the SAFR Desktop Client, click **Tools > Preferences > Genetec**.
3. In the Cardholders section click **Import Now...**. Pressing this button causes the following to occur:
   - Each imported cardholder is given a *Person Type* based on their assigned group.

- If a cardholder has multiple group memberships, the cardholder group with the highest access privilege is used to define the group.
- After import, SAFR updates the events in Genetec to make sure Genetec has one event for each *Person Type*.

4. You can configure SAFR to import new cardholders every 24 hours by selecting the **Import Every 24 Hours** check box.

You can also register people to SAFR by using SAFR's native functionality. For more information, see Importing and Registering People. Although people registered with SAFR are never synchronized to Genetec, you may want to register people to SAFR anyways when you want to add threats, concerns, or other registered people who may not be suitable as Genetec cardholders.

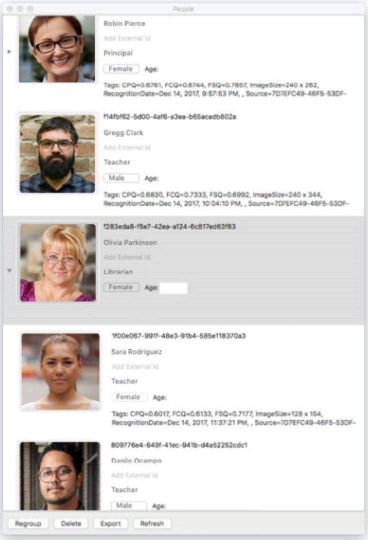
108.4.1 Edit Cardholder Data

You may want to edit people's properties to better manage which events get triggered when that person is recognized, For example, setting a person's *ID Class* to *Concern* or *Threat* triggers the corresponding alarms, while changing a cardholder group can allow you to trigger a VIP alert for specific cardholder groups. The most important people attributes are the *Name*, *Image*, *Person Type*, and *ID Class*.

Attributes should be edited through Genetec Security Center whenever possible. *Person Type* defines a person's role (for example, staff or visitor) while the *ID Class* defines the risk level (No-Concern, Stranger, Concern, or Threat). *Person Type* and *Image* can be edited in Security Center by changing the cardholder group a person belongs to.

To edit these attributes, open Cardholder Management in Genetec Config Tool and update the desired users. After making changes, make sure to either manually synchronize users or set automatic synchronization as described previously in the "Register Cardholders".

ID Class and any other attributes of a person must be edited in SAFR's People dialog accessed through the Desktop Client > Tools menu. All cardholders imported from Genetec Security Center are assigned an *ID Class* of *Normal*. To edit the *ID Class* of a person, click **Tools > People** in the Desktop Client. The following window is displayed:



Robin Pierce
Principal
Female
Tags: CPQ=0.6761, FCQ=0.6744, FSQ=0.7857, ImageSize=240 x 262, RecognitionDate=Dec 14, 2017, 9:57:53 PM, , Source=7D7EFC49-46F5-53DF-

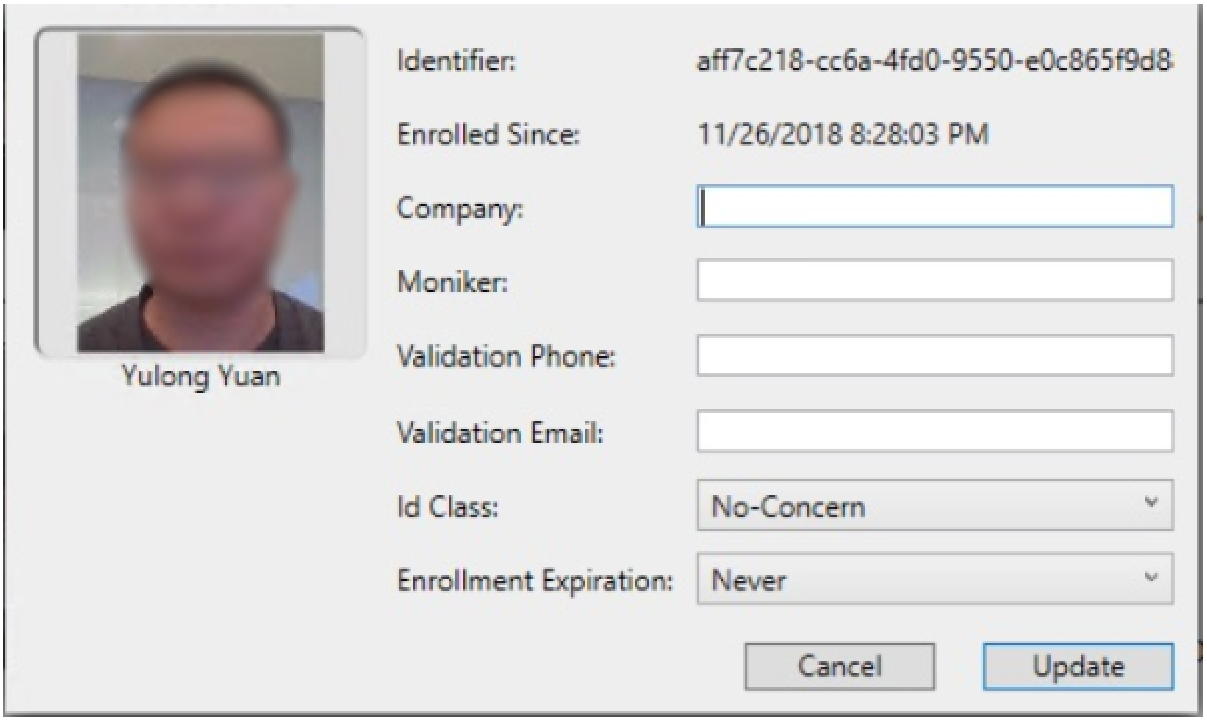
f14fbf62-5d00-4af6-a3ea-b65acadb802a
Gregg Clark
Teacher
Male
Tags: CPQ=0.6830, FCQ=0.7333, FSQ=0.6992, ImageSize=240 x 344, RecognitionDate=Dec 14, 2017, 10:04:10 PM, , Source=7D7EFC49-46F5-53DF-

f283eda8-f5e7-42ea-a124-6c817ed63f83
Olivia Parkinson
Librarian
Female

1f00e067-991f-48e3-91b4-585e118370a3
Sara Rodriguez
Teacher
Female
Tags: CPQ=0.6017, FCQ=0.6133, FSQ=0.7177, ImageSize=128 x 154, RecognitionDate=Dec 14, 2017, 11:37:21 PM, , Source=7D7EFC49-46F5-53DF-

809776e4-649f-41ec-941b-d4a52252cdc1
Danilo Ocampo
Teacher
Male

The *Person Type* and *Name* attributes can be edited by clicking their respective fields in the People window. To edit *ID Class*, in the **People Edit** dialog, double-click the user and choose the desired value from the *ID Class* menu as shown in the following image:



Yulong Yuan

Identifier: aff7c218-cc6a-4fd0-9550-e0c865f9d8

Enrolled Since: 11/26/2018 8:28:03 PM

Company:

Moniker:

Validation Phone:

Validation Email:

Id Class: No-Concern

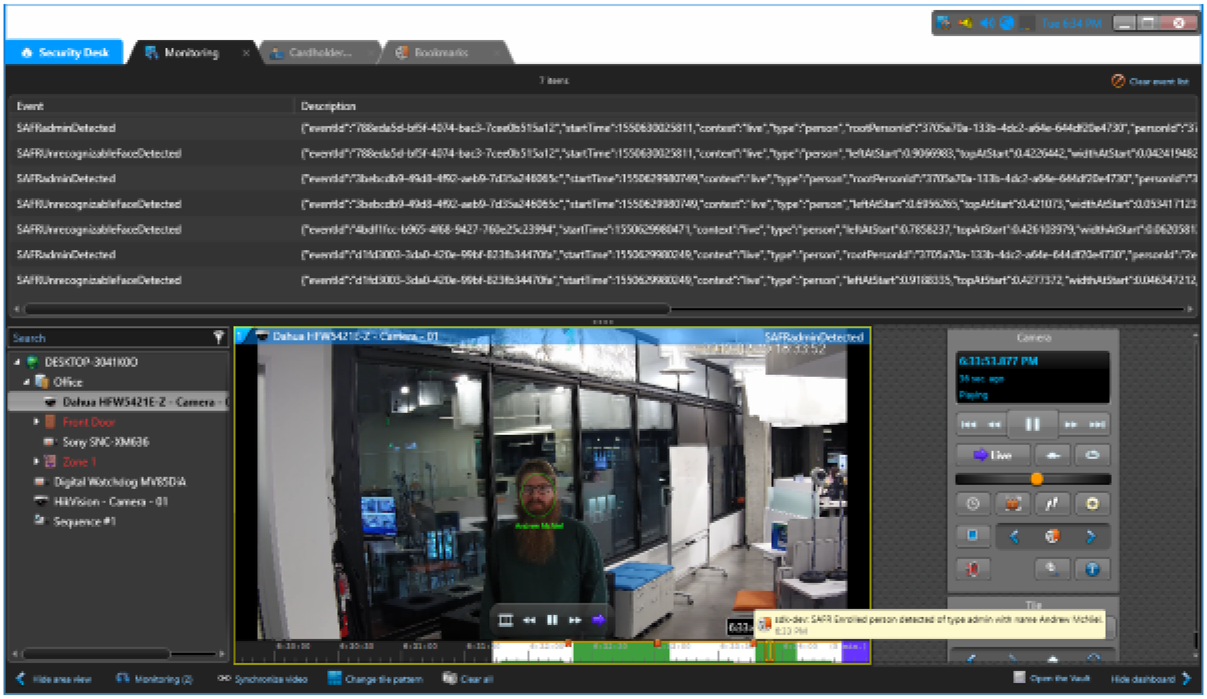
Enrollment Expiration: Never

108.5 SAFR Events

You can use SAFR to generate events. When enabled on the Events Preferences page, events occur when a person appears in the view of a connected camera. When a person appears in the camera's view, they're immediately assigned an *ID Class* attribute, although that *ID Class* may change if the system successfully recognizes them and assigns them a more appropriate *ID Class*. The types of *ID Classes* are listed below.

- Unrecognizable: A face was detected but it wasn't of sufficient quality to attempt recognition.
- Stranger. The person has been recognized, but they're not in Genetec's cardholder database nor in SAFR's Person Directory.
- Registered person without a name.
- Registered person with a name.
- Registered person marked as a *Concern*.
- Registered person marked as a *Threat*.

There are several different combinations of the conditions that are triggered. The following image shows multiple events populated in the Genetec alerts panel. Clicking any of the events allows the video from that event to be replayed:



Event	Description
SAFRadminDetected	{"eventId":"788eda5d-bf5f-4074-bac3-7cee0b515a12","startTime":1550630025811,"context":"live","type":"person","rootPersonId":"3705a70a-133b-4dc2-a64e-644df20e4730","personId":"3
SAFRUnrecognizableFaceDetected	{"eventId":"788eda5d-bf5f-4074-bac3-7cee0b515a12","startTime":1550630025811,"context":"live","type":"person","leftAtStart":0.9066983,"topAtStart":0.4226442,"widthAtStart":0.04241948
SAFRadminDetected	{"eventId":"3bebcdb9-49d8-4f92-aeb9-7d25a248065c","startTime":1550629980749,"context":"live","type":"person","rootPersonId":"3705a70a-133b-4dc2-a64e-644df20e4730","personId":"3
SAFRUnrecognizableFaceDetected	{"eventId":"3bebcdb9-49d8-4f92-aeb9-7d25a248065c","startTime":1550629980749,"context":"live","type":"person","leftAtStart":0.6956265,"topAtStart":0.421073,"widthAtStart":0.05341712
SAFRUnrecognizableFaceDetected	{"eventId":"4bdf1fcc-b965-4f68-9427-760e25c23994","startTime":1550629980471,"context":"live","type":"person","leftAtStart":0.7858237,"topAtStart":0.426103979,"widthAtStart":0.0620581
SAFRadminDetected	{"eventId":"d1fd3003-3da0-420e-99bf-823fb34470fa","startTime":1550629980249,"context":"live","type":"person","rootPersonId":"3705a70a-133b-4dc2-a64e-644df20e4730","personId":"2e
SAFRUnrecognizableFaceDetected	{"eventId":"d1fd3003-3da0-420e-99bf-823fb34470fa","startTime":1550629980249,"context":"live","type":"person","leftAtStart":0.9188335,"topAtStart":0.4277372,"widthAtStart":0.04634721 2

sdk-dev: SAFR Enrolled person detected of type admin with name Andrew McNiel.
6:33 PM

The following table lists the available events that are SAFR makes available to Genetec.

Event Message	Id Class	Named	Person Type	Condition	People Attributes
Unrecognizable face detected	N/A	N/A	N/A	Face detected but insufficient information for recognition	idClass="unidentified"
Stranger detected	Stranger	N/A	N/A	Face detected but not found in registered people	idClass="stranger"
Registered person detected without name	Normal	No	None	Registered person without name or person type assigned	idClass="noconcern" && personType="" && name=""
Registered person detected with name <name>	Normal	Yes	None	Registered person with name but no person type	idClass="noconcern" && personType="" && name=<name>
Registered person detected of type <personType>	Normal	No	Defined	Registered person with person type but no name	idClass="noconcern" && personType=<personType> && name=""

Event Message	Id Class	Named	Person Type	Condition	People Attributes
Registered person detected of type <personType> with name <name>	Normal	Yes	Defined	Registered person with person type and name	idClass=“noconcern” && personType=<personType> && name=<name>
Concern person detected without a name	Concern	No	None	Same as above for Concern	idClass=“concern” && personType="" && name=""
Concern person detected with name <name>	Concern	Yes	None	Same as above for Concern	idClass=“concern” && personType="" && name=<name>
Concern person detected of type <personType>	Concern	No	Defined	Same as above for Concern	idClass=“concern” && personType=<personType> && name=""
Concern person detected of type <personType> with name <name>	Concern	Yes	Defined	Same as above for Concern	idClass=“concern” && personType=<personType> && name=<name>
Threat person detected without a name	Threat	No	None	Same as above for Threat	idClass=“threat” && personType="" && name=""
Threat person detected with name <name>	Threat	Yes	None	Same as above for Threat	idClass=“threat” && personType="" && name=<name>
Threat person detected of type <personType>	Threat	No	Defined	Same as above for Threat	idClass=“threat” && personType=<personType> && name=""
Threat person detected of type <personType> with name <name>	Threat	Yes	Defined	Same as above for Threat	idClass=“threat” && personType=<personType> && name=<name>

108.5.1 SAFR Video Processing Modes

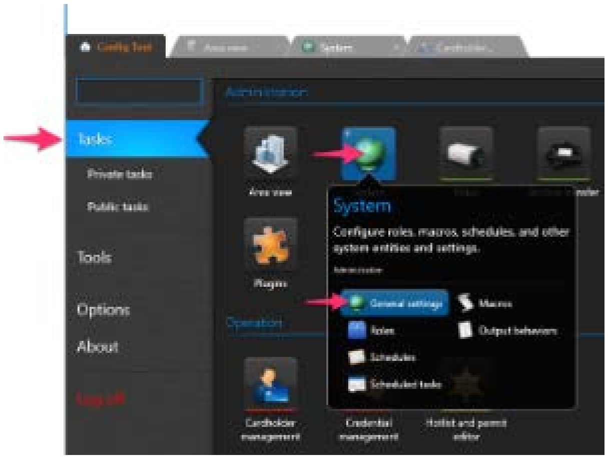
SAFR has different video processing modes that control what events are generated. Below is a short summary of the modes most relevant to Genetec Security Center integration. For a complete description, see Connect to a Video Feed in the *SAFR Documentation*.

- **Secure Access**: Only triggers events when cardholders and people registered in SAFR's Person Directory are identified with a high degree of certainty. This mode is useful when the system is being used to manage physical access. (i.e. when the system is responsible for unlocking doors for authenticated people)
- **Secure Access with Smile**: Similar to *Secure Access* mode, except that registered people must smile in order to cause the system to grant them access.
- **Enrolled Monitoring**: Similar to *Secure Access* mode, but events are triggered at a lower recognition confidence level.
- **Enrolled and Stranger Monitoring**: Similar to *Enrolled Monitoring* mode, but events are also triggered for strangers.

108.5.2 Add and Configure Alerts

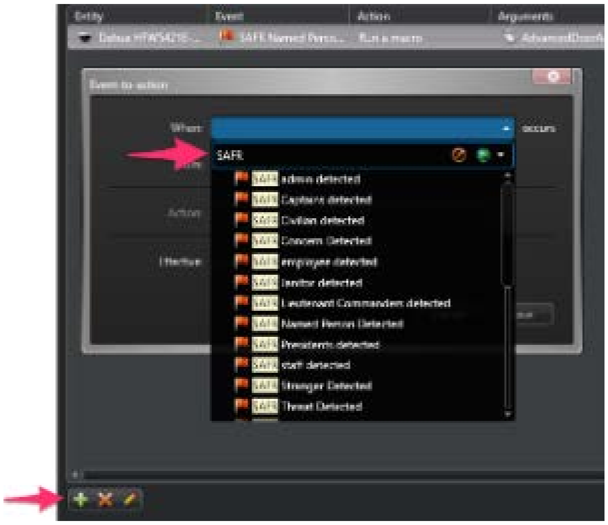
To trigger the alert as a result of a SAFR-generated event, do the following:

1. Open the Genetec Config Tool, and go to the System Panel.

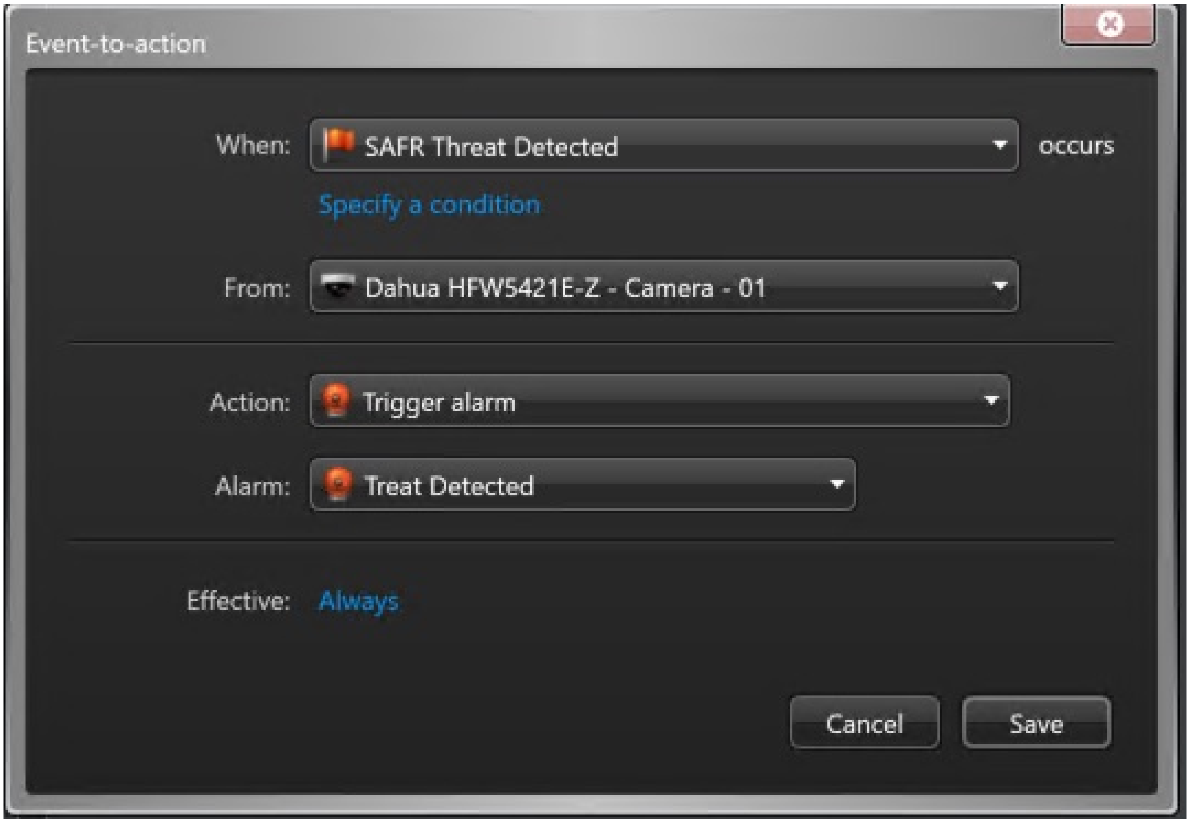


Configure roles, macros, schedules, and other system entities and settings.

2. Click the + icon to add a new alarm, and click the When menu. Type *SAFR* and press **Enter** to see the list of SAFR-enabled alarms.



3. Choose the desired entry from the list.
4. Under **From**, choose the camera you want to use to trigger the event. Under **Action**, choose a desired action. (e.g. Trigger Alarm)



Event-to-action

When: SAFR Threat Detected occurs

Specify a condition

From: Dahua HFW5421E-Z - Camera - 01

Action: Trigger alarm

Alarm: Treat Detected

Effective: Always

5. Click **Save** when done.

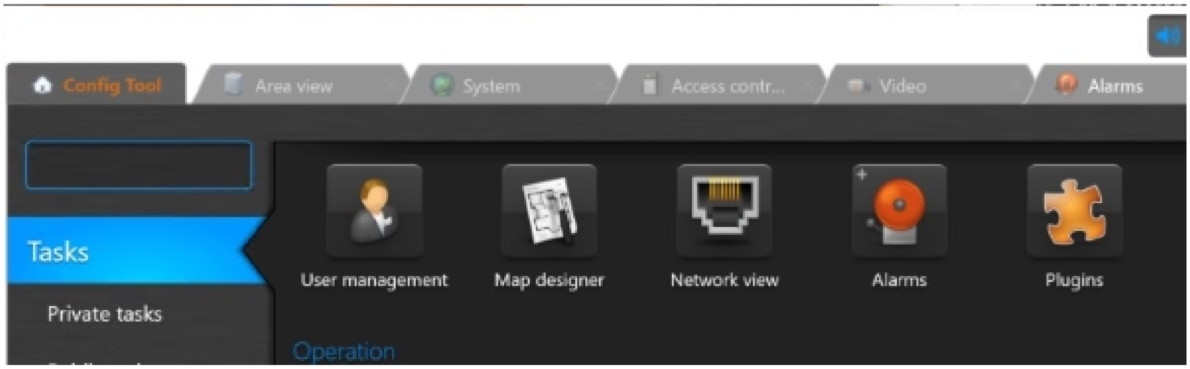
SAFR Events can be tied to Actions which can then trigger an Alarm. Initially create an alarm you want to trigger, and then use Genetec Event-to-Action dialog to tie SAFR Events to any action that can be defined in the Genetec system (for example, Trigger Alarm).

108.5.3 Add an Alarm

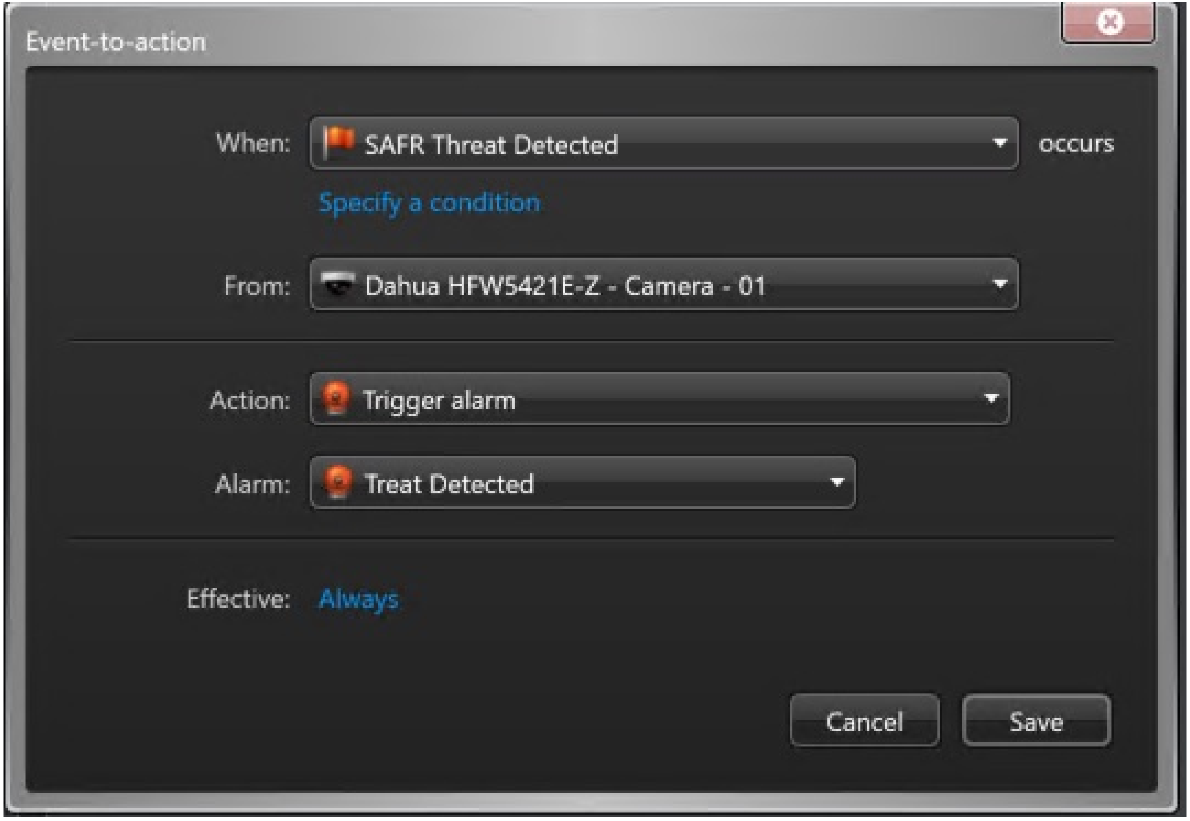
An alarm can be used to make sure an important event is noticed. In this example, we show how to create an alarm that is triggered when someone who has been marked as a threat is recognized on one of the cameras. For more information on triggering events, refer to Genetec support documentation.

To create an alarm, do the following:

1. Open the Genetec Config Tool, and open the Alarms screen.



2. Click **Add Alarm**, and type a *Name* as shown in the following image:


Event-to-action

When: SAFR Threat Detected occurs

Specify a condition

From: Dahua HFW5421E-Z - Camera - 01

Action: Trigger alarm

Alarm: Treat Detected

Effective: Always

3. Click **Save** to save the alarm.

108.5.4 Recommended Settings for Alarms

Properties task	Choose priorities based on circumstances and your organization guidelines (1=high, 255=low) • Stranger: 100 (If infrequent, set high) • Concern: 50 • Threat: 10 Video display option Set to Live to see the live view when alarm loads video Playback may be useful for short events where the subject may have walked off the screen by the time the video loads • If playback mode, set to at least 4 seconds to avoid buffering
Advanced task	Auto-Acknowledge: Good for stranger events; enter the number of seconds to stay in the view before returning to the view you were on prior to the event Choose color to match the SAFR colors (add ref to section in manual that describes colors) Reactivate threshold: Suppresses additional alarm if another similar alarm triggered within this time • Adjust as needed for use case.

108.5.5 Trigger Macros

When SAFR is configured to *Include Event Details* in reported events, highly customized actions can be programmed using macros in the Genetec system. Event details include all information associated with the detected face (e.g. *Name*, *Person Type*, *Age*, *Gender*, *Sentiment*, etc.). For more information on macros, refer to the Genetec support documentation.

108.6 Troubleshooting Tips

Note: When closing SAFR, use the **Quit SAFR** option on the **File** menu. Closing SAFR using the Window Close button will cause you to lose the SAFR state settings and connected cameras for that window.

- If detection or recognition is poor (not many faces found or recognized), make sure the Genetec video feeds are set for a sufficiently large frame size.
- If events are not being triggered, check the following:
  - Permissions are set correctly on Event-to-Actions.
  - Make sure the applicable SAFR Video Processing Mode is selected.

109 SAFR Genetec FaceRec Integration Guide

Integrated SAFR Genetec FaceRec is only available on Windows.

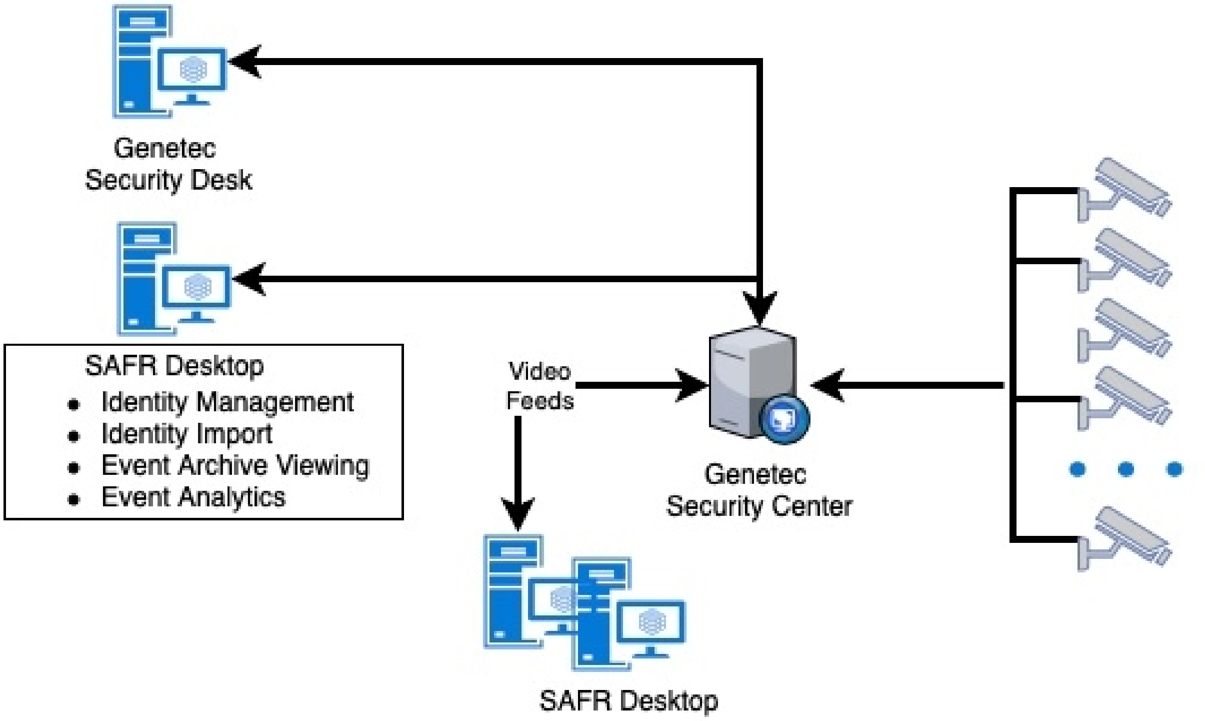
Integrating SAFR's facial recognition and analysis capabilities into Genetec enables you to use SAFR's video feed information overlays within Genetec camera video feeds, thus making it much easier to quickly and accurately separate unknown people from authorized people from known threats. You'll also have immediate access to additional infomation such as age, gender, sentiment, name, company, known associates, or any other configurable information you want to create.

Integrating the two systems also allows SAFR's information about individuals to trigger Genetec alerts and other actions within the Genetec system.

109.1 Integration Overview and Requirements

A typical deployment requires the following:

- A machine running Genetec Security Center.
- A machine running Genetec Security Desk and Genetec Config Tool.
- One or more machines running the SAFR Desktop Client to process videos.
- If you're doing a SAFR local deployment, you'll also need a machine running SAFR Server. SAFR Server can run on the same machine as one of the Desktop Clients, provided the host machine meets the system requirements.



Cameras are connected to the Genetec Security Center. The SAFR Desktop Client(s) can then connect to the Genetec Security Center to perform analysis of the video and add overlays. Depending on the number of cameras you need, one or more machines may be required to run the SAFR Desktop Client, each processing multiple video feeds. The Desktop Client processes the video and returns information to Genetic to overlay the video feeds and generate events. The Desktop is also used to perform various management activities.

109.1.1 System Requirements

Genetec has the following system requirements:

- One machine running Genetec Security Center Version 5.7 or later.
- One machine running Genetec Security Desk and Genetec Config Tool.
- Each machine running a Genetec product must meet the following system requirements:
  - Windows 10.
  - Additional system requirements as described here.
  - Genetec FaceReq plugin. **Note**: The installation of this plugin is performed when you install SAFR. See the Install and Configure SAFR section below for details.

SAFR has the following system requirements:

- One or more machines running Windows SAFR Desktop Client 1.3.228 or later.
- Each machine running the SAFR Desktop Client must meet the following requirements:
  - Windows 10.
  - Additional system requirements as described here.
- Local SAFR deployments require at least one machine running SAFR Platform 1.3 or later.
- Each machine running SAFR Server must meet the following requirements:
  - Windows 10.
  - Additional system requirements as described here.

109.1.2 Licensing and the Genetec Part Number

An accompanying Genetec part number must be added to your Genetec connection license. Do the following to discover and add the Genetec part number:

1. Go to the Genetec Portal and sign in using your Genetec credentials.
2. In the applications section, search for *SAFR*. From the results, click *SAFR Facial Recognition*.
3. On the **SAFR Facial Recognition Solution Details** page, in the right column, the *Genetec Part Number* is displayed.
4. Contact Genetec and have them add the part number to your license. You need a quantity of the part number equal to the number of cameras SAFR will be processing plus one additional license for the metadata channel SAFR creates. In other words, if SAFR will be processing cameras, then you need quantity of the part number added to your license.

You'll need the following licenses: each Genetec camera where SAFR face detection and recognition is used, you'll need:

- A Genetec connection license with the accompanying Genetec part number is required for each connected Genetec camera.
- One additional Genetec connection license for the metadata channel SAFR creates.
- A SAFR license for each camera is required.

For example, if you have 300 cameras but only need face detection on 30 cameras at a time, then you would obtain a 31 connection license from Genetec and a 30 camera license from RealNetworks. Having a 31 connection Genetec license does not mean you are limited to face detection on a fixed set of 30 cameras. At any time, you can choose to connect the SAFR Desktop Client to a different camera. You may have cameras in your parking garage that you were not previously monitoring with SAFR recognition. You can use a few of your licenses that are connected to other cameras to connect to garage cameras instead.

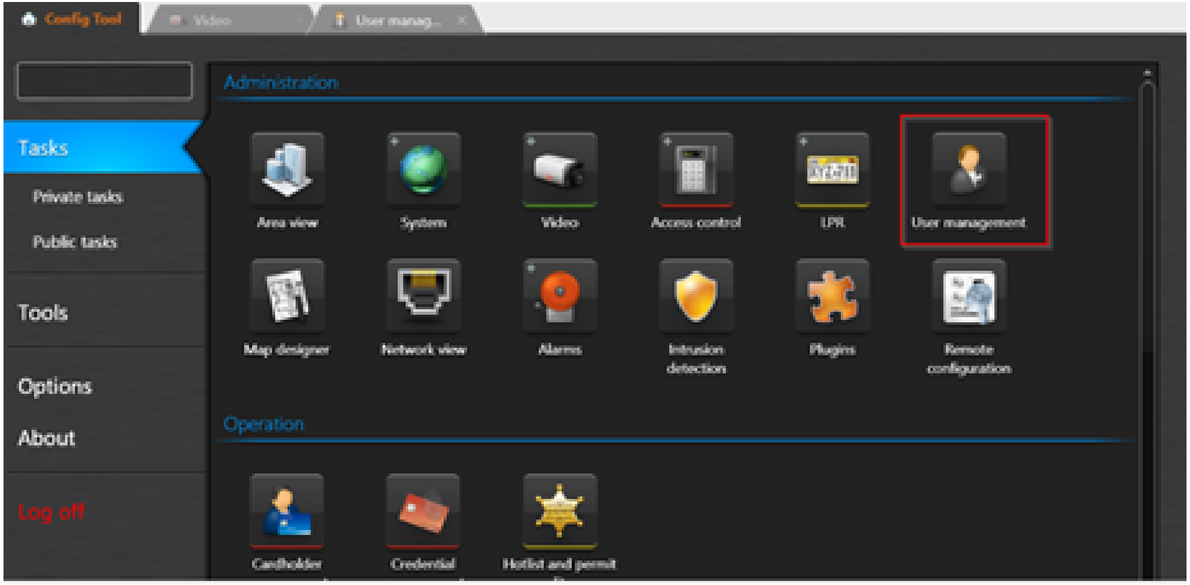
109.2 Install and Configure Genetec Products Security Center

1. Download the latest version of Genetec Security Center from the Genetec Portal.
2. Run the installer. For details about which install options to select, see the Security Center Installation and Upgrade Guide.
3. Download and install the latest Genetec SDK package.

109.2.1 Create a SAFR User

To create a user with the permissions that SAFR will require:

1. Open the Genetec Config Tool.
2. Click **Tasks** > **User Management**.



3. Create a new user (with a username of, for example, *SAFR*) with the following permissions:

All privileges

- Application privileges
  - Log on using the SDK

Administrative privileges

- Physical entities
  - View camera properties

Access control management

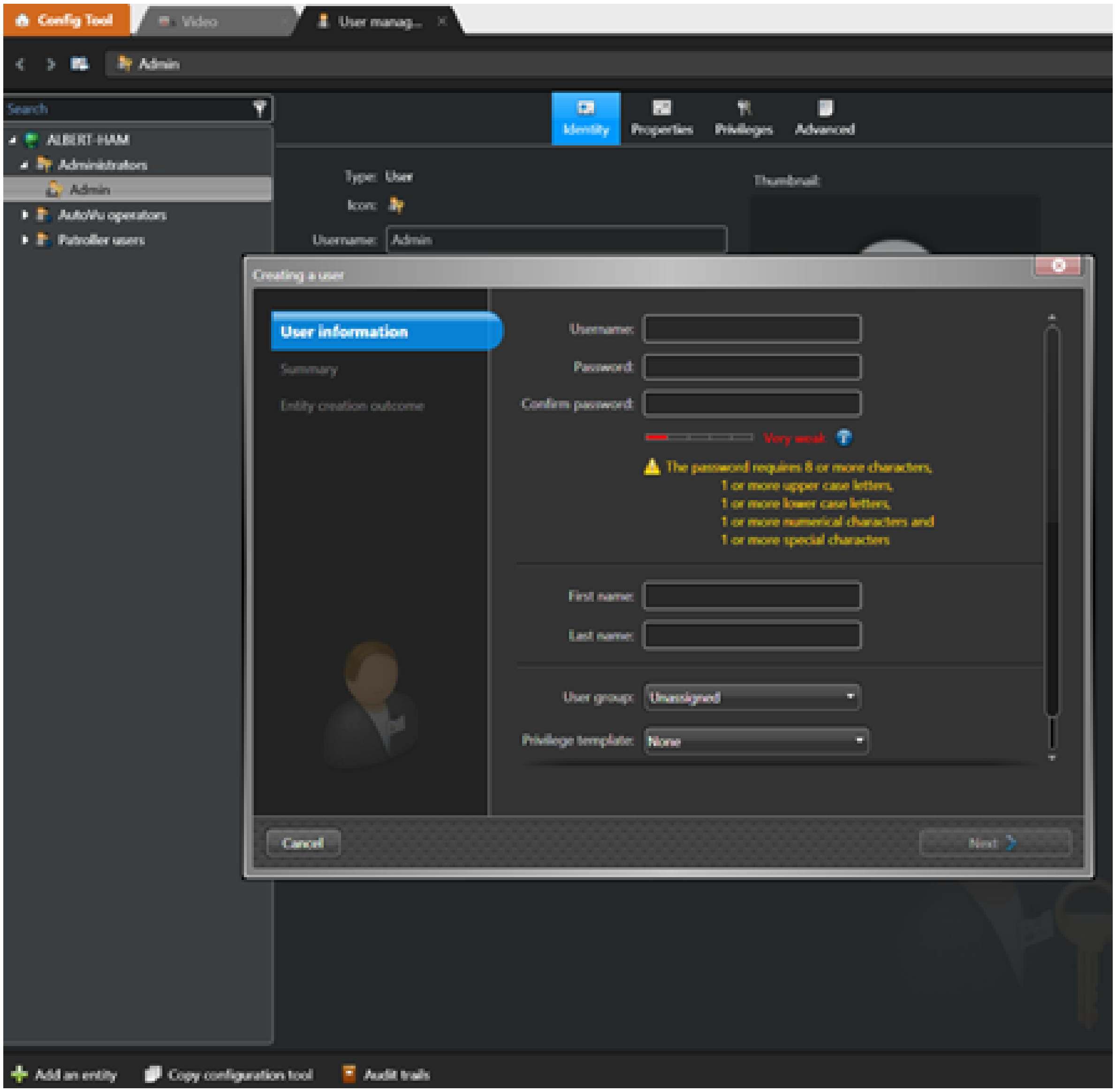
- View cardholder group properties
- View cardholder properties
- View visitor properties

System management

- View general settings
  - Modify custom events

Action Privileges

- Cameras
  - View live video



Type: User

Username: Admin

Username:

Password:

Confirm password:

Very weak

⚠ The password requires 8 or more characters,
1 or more upper case letters,
1 or more lower case letters,
1 or more numerical characters and
1 or more special characters

First name:

Last name:

User group: Unassigned

Privilege template: None

109.2.2 Add Permissions for Event-to-Archive Actions

In order to create Event-to-Actions in the Genetec Config Tool, one or more of the following Action permissions must also be added to the SAFR user created in the previous section. Only those actions you want to trigger with SAFR events are needed:

All privileges

- Action privileges
  - Set threat level
  - Cameras
    - Protect video from deletion
    - Save/modify/print snapshots
  - Access control
    - Doors
      - Explicitly unlock doors
      - Override unlock schedules
    - Elevators

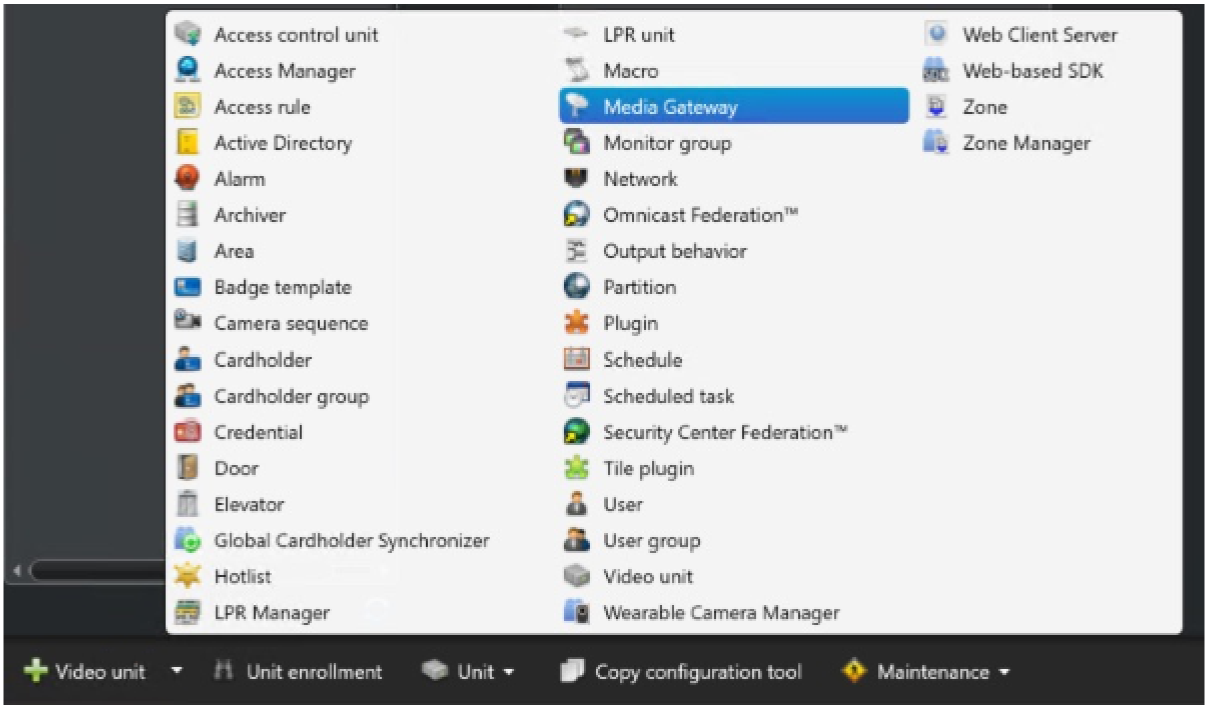
        - Override elevator schedules
- Alarms
    - Trigger alarms
- Users
    - Send a message
    - Send an email
    - Send/clear task
- Macros
    - Execute macros
- Zones
    - Arm/disarm zones
- Areas
    - Modify people count

109.2.3 Set Minimum Cardholder Image Size

Do the following:

1. Open the Genetec Config Tool.
2. Open **Tasks > Administration > Access Control > General Settings**.
3. Set *Maximum Picture File Size* to 128k or larger.

109.2.4 Configure the Media Gateway



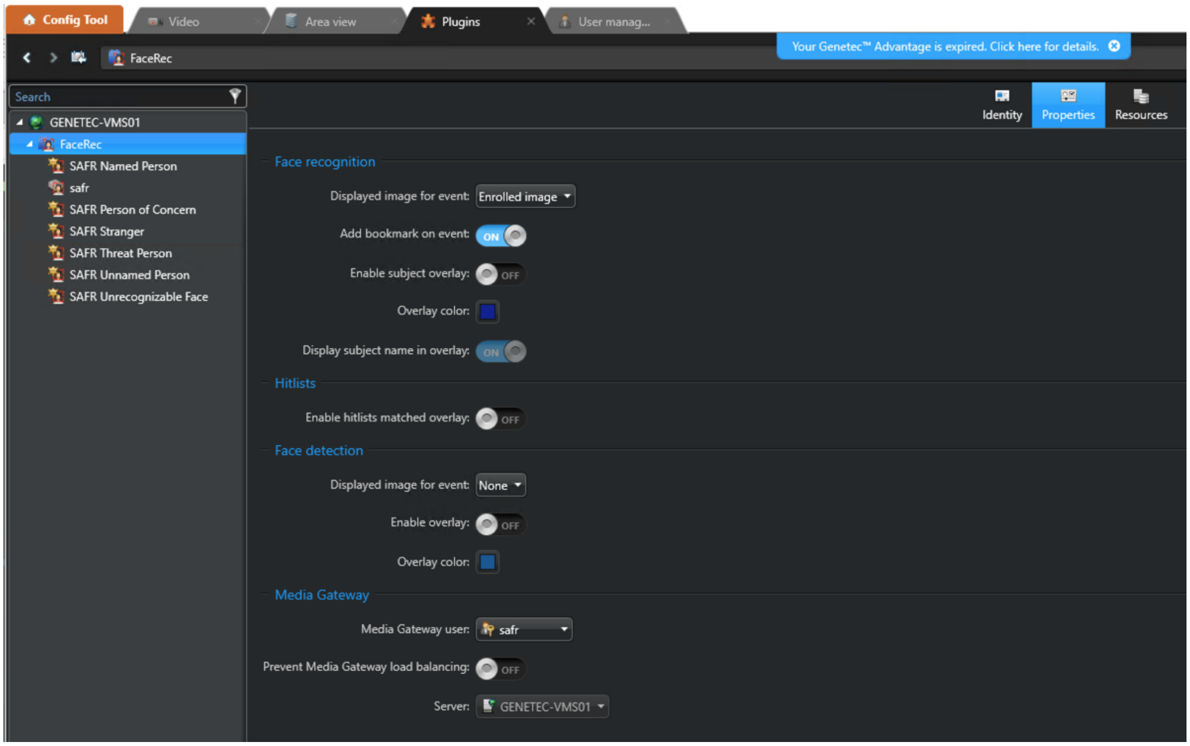
Do the following:

1. Open the Genetec Config Tool.
2. Open **Tasks > Administration > Video**.
3. Click the arrow next to the Video Unit button in the bottom left corner, and select **Media Gateway**.
4. Click **Next**, and in the *Create Media Gateway* wizard, click **Create**. Accept the default values; no changes are needed.
5. Select **Media Gateway**, and click the **Properties** task.
    - This adds a *Media Gateway* entry in the list on left side.

6. Determine the user to be granted access to the media gateway.
    - This can be the SAFR user or a different user; we recommend using the same SAFR user unless you already have one configured to use the Media Gateway.
    - This user does not need to have specific permissions. The permissions for media gateway are granted to this user in the next step.
7. To add this user to the **Accessible To** section, click the + icon. In the bottom right, click **Apply** to save the changes.
8. When prompted, enter a password for the user you are adding.
    - This password can be the same as the user's normal password or it can be different.
9. Save the *username* and *password*.
    - This is the password that must be used in the Media Gateway credentials fields in the SAFR preferences window.

109.3 Configure the Genetec FaceReq Plugin

- Click **Add an entry**, choose **Plugin**, then choose **FaceRec**.
- Click **Next**, **Next**, then **Create** to create the new unit.
- Select **FaceRec** entity, click **Add an entry**, then choose **FaceRec Unit**.
    - Enter "safr" as the same and click **Save**.
    - Select **Properties**.
        - Set the following features to OFF:
            - **Face recognition**: Enable subject overlay
            - **Hitlists**: Enable hitlists matched overlay
            - **Face detection**: Enable overlay



Your Genetec™ Advantage is expired. Click here for details.

Face recognition

Displayed image for event: Enrolled image

Add bookmark on event: ON

Enable subject overlay: OFF

Overlay color:

Display subject name in overlay: ON

Hitlists

Enable hitlists matched overlay: OFF

Face detection

Displayed image for event: None

Enable overlay: OFF

Overlay color:

Media Gateway

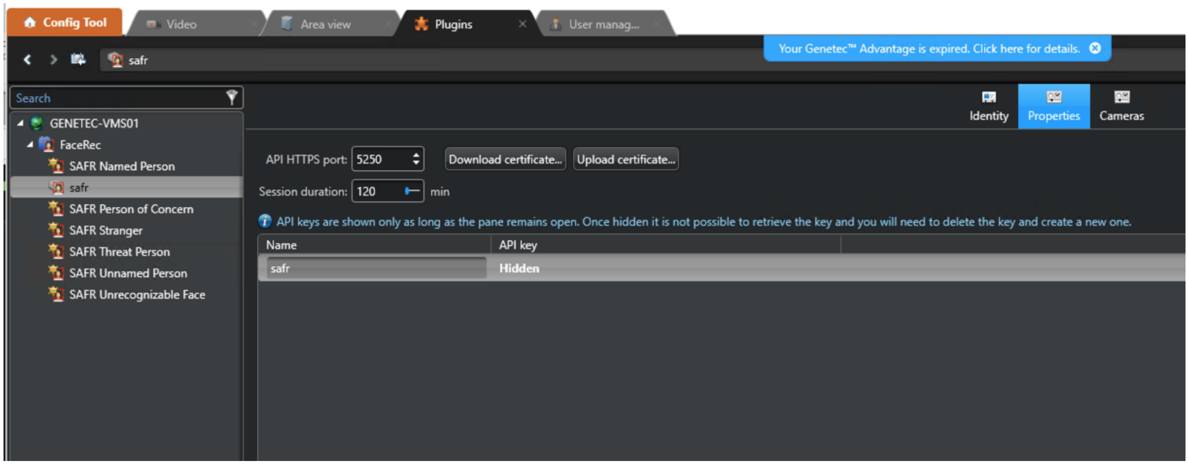
Media Gateway user: safr

Prevent Media Gateway load balancing: OFF

Server: GENETEC-VMS01

- Select **safr** unit.
    - Select **Properties**.
        - In the API dialog area click the + button to add a new API Key entry.

- Name it “safr” and make a note of the API Key value. (The value is only available once.) **Note**: API Key is used later when configuring the SAFR Desktop Client.
- Click **Apply**.



Your Genetec™ Advantage is expired. Click here for details.

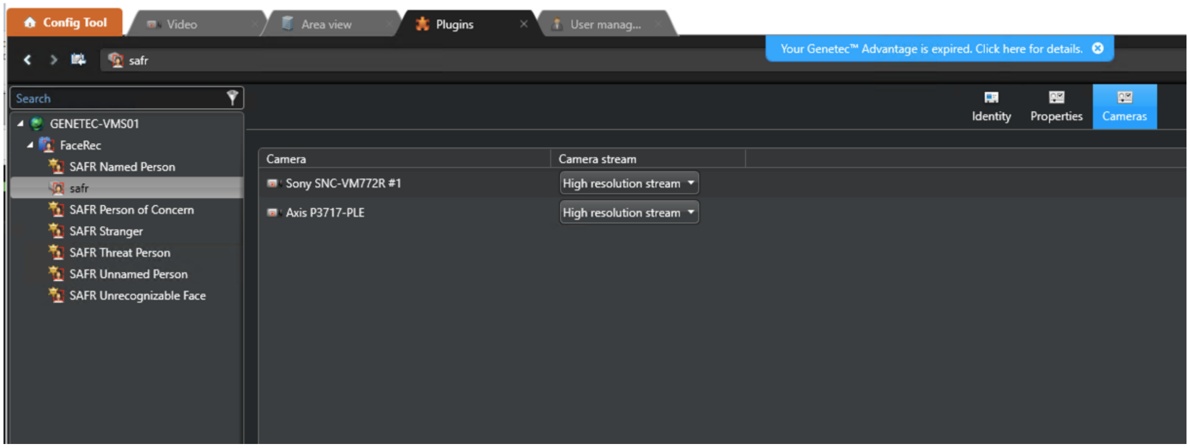
API HTTPS port: 5250

Session duration: 120 min

API keys are shown only as long as the pane remains open. Once hidden it is not possible to retrieve the key and you will need to delete the key and create a new one.

Name	API key
safr	Hidden

- Select **Cameras**.
  - Click the + button and add all relevant cameras which require face recognition.

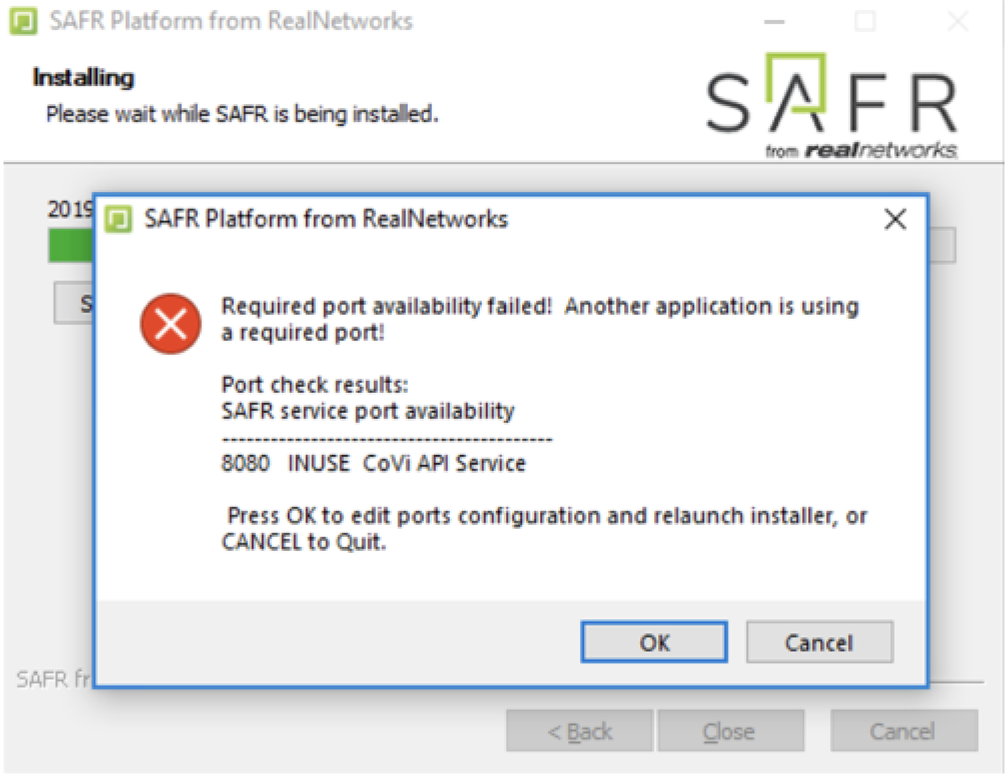


Your Genetec™ Advantage is expired. Click here for details.

Camera	Camera stream
Sony SNC-VM772R #1	High resolution stream
Axis P3717-PLE	High resolution stream

109.4 Install and Configure SAFR

1. On the machine(s) where you plan to install the SAFR Desktop Client, install the Genetec SDK from the Genetec Portal.
2. Go to the SAFR Download Portal.
3. If you’re doing a cloud deployment, download and install Windows SAFR Desktop. Make sure to select the Genetec Security Center with FaceRec install option.
4. If you’re doing a local deployment, download and install Windows SAFR Platform. Make sure to select the Genetec Security Center with FaceRec install option.
   - When installing the SAFR Platform, the default SAFR port assignments sometimes conflict with other software port assignments. If a port conflict occurs, the error message shown below will pop up in the middle of your installation.


SAFR Platform from RealNetworks
Installing
Please wait while SAFR is being installed.
SAFR from realnetworks
2019
S
SAFR fr
< Back
Close
Cancel

SAFR Platform from RealNetworks

Required port availability failed! Another application is using a required port!

Port check results:
SAFR service port availability

```
--------------------------------------------
8080   INUSE  CoVi API Service
```

Press OK to edit ports configuration and relaunch installer, or CANCEL to Quit.

If this happens, do the following:

1. Click **OK** to edit port configurations.
2. **Notepad** will open, displaying the *safrports.conf* file.
3. Edit any conflicting ports to new values. (e.g. CoviHTTP=18080)
4. Save and exit **Notepad**.

The Platform installer will then restart and the new port values will be used. You can find the modified *safrports.conf* file at `C:\Program Files\RealNetworks\SAFR\`.

5. After the installation finishes, a message will appear saying where you can locate the Genetec FaceReq plugin installer on your machine. Make a note of the installer location.
6. Immediately following installation, the installer opens the Desktop Client and prompts you to log in with your SAFR Account credentials. Make sure to log in; it's important in acquiring the SAFR license.
7. Go to the location specified in Step #5. Copy the installer (FaceRecSetup621.exe) to every machine running Genetec Security Center, and then run the installer on each of those machines.

Two icons will have appeared on your desktop: one labeled "SAFRActions" and another labeled "SAFR". *SAFRActions* launches SAFR Actions, while *SAFR* launches the Desktop Client. If you did a local deployment, SAFR Server will be automatically running as a collection of background services.

109.4.1 Connect SAFR to Genetec

1. Within your SAFR Desktop Client, select **Tools->Preferences->Genetec**.
   Note: If the Genetec preference tab is not showing, it means that the Genetec SDK was not properly installed on your machine.

2. Enter the following information in the Genetec preferences tab.
   - **Username**: Enter the SAFR user you created earlier.
   - **Password**: Enter the *Password* you created for the SAFR user.
   - **Directory**: IP address of the server running the Genetec Security Center server.
   - **Media Gateway**: Used for acquiring video streams.

- **Username**: Enter the SAFR user you created earlier.
- **Password**: Enter the *Password* you created for the SAFR user.
- **Port**: Enter the port on which to connect to the Media Gateway. You can use the default value of `654` unless that would create a port conflict.

This should cause your SAFR system to establish a connection with the Genetec system.

To verify that your SAFR system successfully connected to the Genetec system, do the following:

1. On the SAFR Desktop Client, open **Tools -> Preferences -> Camera**.
2. Cameras connected to Genetec system should be visible.
3. All cameras connected to Genetec have the *Genetec* prefix in their names.

109.5 Troubleshooting

109.5.1 How do I Resolve a Certificate Registration Error when Logging in from SAFR to Genetec?

This error is caused by a mismatch between the SAFR Genetec certificate and the Genetec Security Center license. SAFR builds have either a Genetec production certificate or a development certificate. The production certificate can be used only with Security Center installations that use a production or demonstration license. The development certificate can be used only with Security Center installations that use a development license.

Here are some steps you can take to try to diagnose the issue:

1. Use the Genetec Config Tool to connect to the Genetec Security Center server.
2. Click **About** on the left side.
3. Click the **Certificates** tab.
4. If you see a line that says, “Generic certificate for developers” then the Security Center server is using a developer license. You must use a SAFR build that uses a developer certificate. Builds with developer certificate are available only from SAFR build farm and should be used only by developers.
5. If that line is not present, then Security Center is using a production or demonstration license. You must use a SAFR build that uses a production certificate. Download SAFR build with production certificate from the SAFR Download Portal.
   - Click on the **Purchase Order** tab. Production or demonstration licenses must also have a license for SAFR attached to it. There should be a line with `Part #GSC-1SDK-RealN- FaceRec`. The quantity must be equal to or greater than the number of cameras that SAFR will be processing.

109.5.2 How do I Resolve a Connection Error when Logging in from SAFR to Genetec?

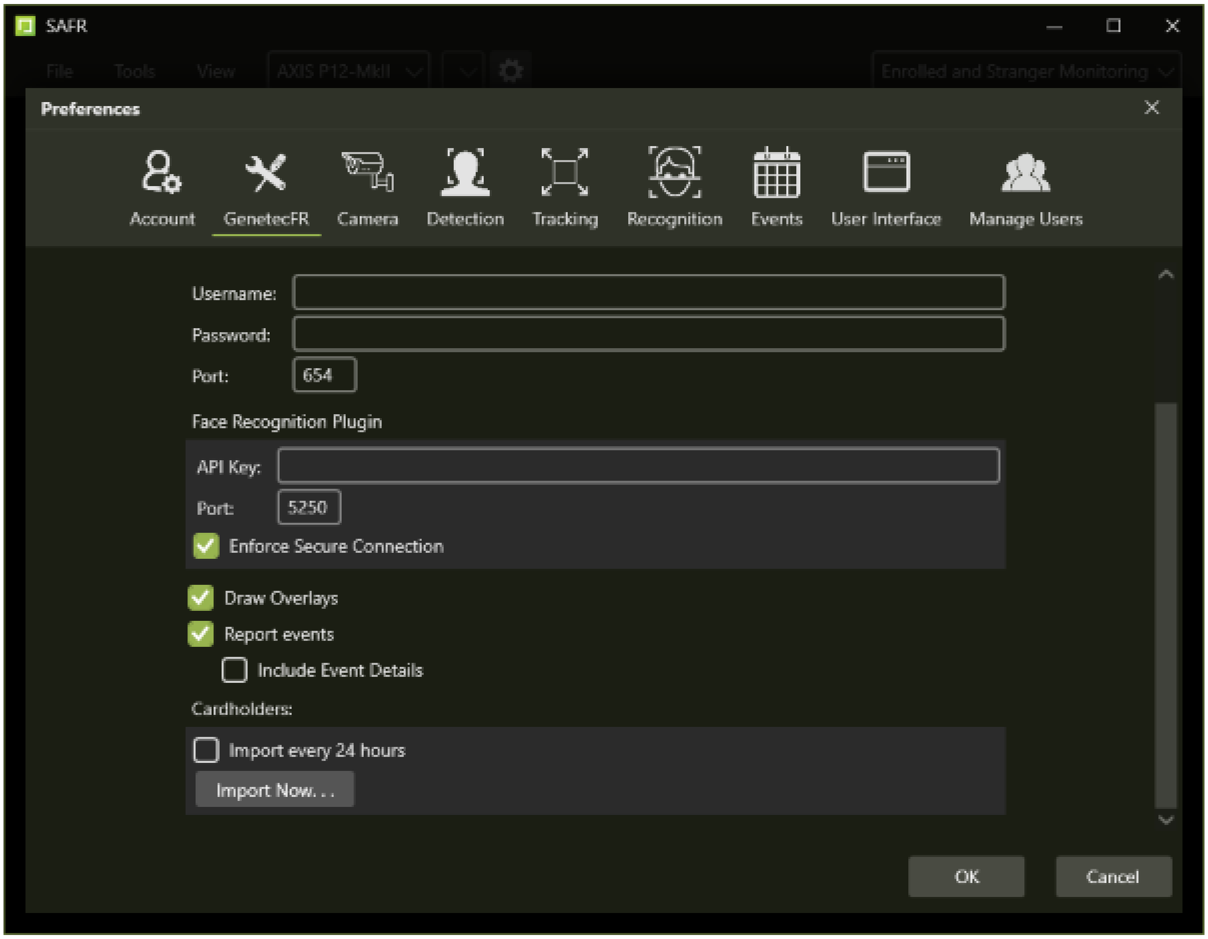
There can be many different causes for a Connection Timeout error from SAFR. However, if you are in a situation where this consistently happens and no cameras are connecting, then doing the following will most likely resolve the error:

1. Connect to the Security Center server using the Genetec Config Tool.
2. Go to the **Video** task.
3. In the left pane, right-click on the **Media Gateway** role.
4. Select the **Maintenance->Deactivate** role.
5. After the role turns gray, right-click on it again.
6. Select the **Maintenance->Activate** role.
7. The Media Gateway will go through a startup routine. It will turn red, yellow, and eventually white.
8. After it turns white, try connecting again.

110 SAFR Genetec FaceRec Operation Guide

110.1 SAFR Genetec Preferences

You can set several Genetec-specific preferences by opening the SAFR Desktop Client and clicking on **Tools -> Preferences -> Genetec**.



Username:

Password:

Port: 654

Face Recognition Plugin

API Key:

Port: 5250

☑ Enforce Secure Connection

☑ Draw Overlays

☑ Report events

☐ Include Event Details

Cardholders:

☐ Import every 24 hours

Import Now. . .

- **Username**: Person with the credentials to connect the SAFR system to the Genetec Security Center Server.
- **Password**: Password of a person with credentials to connect the SAFR system to a Genetec Security Center Server.
- **Directory**: IP address or hostname of the Genetec server.
- **Media Gateway**: Used for acquiring video streams.
  - **Username**: Person with the credentials to connect the SAFR system to a Genetec Security Center Server.
  - **Password**: Password of a person with credentials to connect the SAFR system to a Genetec Security Center Server.
  - **Port**: The port at which SAFR will connect to the Genetec Security Center Server. The default is 654.
- **Face Recognition Plugin**:
  - **API key**: The API key set within the *Face Recognition Activities* section of the Genetec Security Desk.
  - **Port**: The port value used by the *API key* specified within the Genetec Security Desk.
- **Enforce Secure Connection**: When enabled, it requires that a TSL/SSL connection be established with the FaceReq server. Although this setting is enabled by default, it is automatically disabled if the

user chooses “Switch to Insecure Connection” during an error dialogue that pops up if the client failes to establish a secure connection with the FaceReq server.

- **Draw Overlays**: Enables the drawing of ovals, names, and other details within Genetec camera video stream. The overlays match what would be shown in the SAFR Desktop Client, so SAFR settings affecting SAFR overlays also affect what is drawn in Genetec.
  Note: If you enable SAFR's overlays, you should disable Genetec's overlays. You can do this by opening the Genetec Security Desk, then going to **Options->Visual**, and then disabling the **Display overlay video controls** option.
- **Report Events**: Enables reporting SAFR events to Genetec. If this setting isn't checked, *Include Event Details* is automatically greyed out.
  - **Include Event Details**: When enabled, all of the technical details of the event are attached to events. This option is especially useful if an operator uses macros to handle events for decision making.
- **Cardholders**
  - **Import Every 24 Hours**: When enabled, all the Genetec cardholders not already in SAFR's Person Directory are imported and registered to SAFR every 24 hours.
  - **Import now. . .** : Clicking this causes all the Genetec cardholders not already in SAFR's Person Directory to be imported and registered to SAFR.

110.2 Connect and Use Cameras and Video Feeds

1. To connect cameras to Genetec, you need to add the cameras to the Genetec Video Archiver using the Genetec Config Tool. For details, please see the Genetec Security Center Administrator Guide.
2. After a camera has been added to the Video Archiver, it should be displayed as a Genetec camera in SAFR. If it's not, try closing and re-opening the SAFR Desktop Client.
   To get SAFR video feed overlays to be displayed on Genetec camera feeds, do the following:
   1. Open the SAFR Desktop Client.
   2. Select the Genetec version of the camera from the menu in the main windows (upper left). The word “Genetec” will be the first part of the camera name.
   3. After the client has successfully connected to the Genetec camera, video from the Genetec camera is displayed in the SAFR Desktop Client video feed window.
   4. Open the Genetec Security Desk.
   5. Go to the **Monitoring** Task.
   6. Drag and drop a camera from the left side into one of the tiles in the middle.
   7. The camera feed should appear and show the same video feed overlays that are in SAFR.

To connect additional cameras:

1. Open another instance of the Desktop Client by selecting **File > New** on the client.
2. Repeat steps 2-6 above.
3. You can keep repeat this procedure to add overlays to as many video feeds as desired.
   Note: Most machines can only support up to 16 video feeds. If you want to connect more feeds than that, you'll need to install the SAFR Desktop Client on additional machines.

By default, the SAFR Desktop Client operates in the *Enrolled Monitoring* video processing mode and generates events and bookmarks into the Genetec system for every registered person. If you want a different mode for a given camera, choose a different mode from that camera's *Camera* window **Mode Selector** menu.

110.3 Genetec Cardholders and SAFR People

Genetec cardholders can be registered to SAFR by doing the following:

1. Increase the Genetec Security Center setting for thumbnail size to make sure SAFR has access to high quality images to use for face recognition.
2. On the SAFR Desktop Client, click **Tools > Preferences > Genetec**.
3. In the Cardholders section click **Import Now. . .** . Pressing this button causes the following to occur:

- Each imported cardholder is given a *Person Type* based on their assigned group.
- If a cardholder has multiple group memberships, the cardholder group with the highest access privilege is used to define the group.
- After import, SAFR updates the events in Genetec to make sure Genetec has one event for each *Person Type*.

4. You can configure SAFR to import new cardholders every 24 hours by selecting the **Import Every 24 Hours** check box.

You can also register people to SAFR by using SAFR's native functionality. For more information, see Importing and Registering People. Although people registered with SAFR are never synchronized to Genetec, you may want to register people to SAFR anyways when you want to add threats, concerns, or other registered people who may not be suitable as Genetec cardholders.

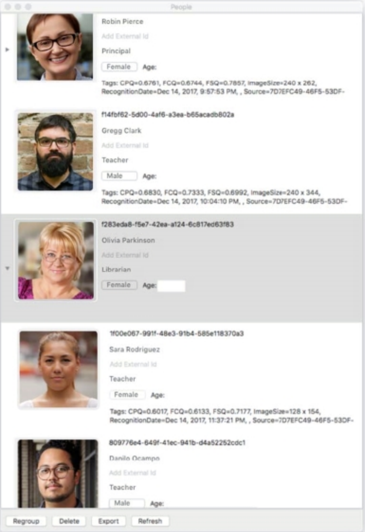
110.3.1 Edit Cardholder Data

You may want to edit people's properties to better manage which events get triggered when that person is recognized, For example, setting a person's *ID Class* to *Concern* or *Threat* triggers the corresponding alarms, while changing a cardholder group can allow you to trigger a VIP alert for specific cardholder groups. The most important people attributes are the *Name*, *Image*, *Person Type*, and *ID Class*.

Attributes should be edited through Genetec Security Center whenever possible. *Person Type* defines a person's role (for example, staff or visitor) while the *ID Class* defines the risk level (No-Concern, Stranger, Concern, or Threat). *Person Type* and *Image* can be edited in Security Center by changing the cardholder group a person belongs to.

To edit these attributes, open Cardholder Management in Genetec Config Tool and update the desired users. After making changes, make sure to either manually synchronize users or set automatic synchronization as described previously in the "Register Cardholders".

ID Class and any other attributes of a person must be edited in SAFR's People dialog accessed through the Desktop Client > Tools menu. All cardholders imported from Genetec Security Center are assigned an *ID Class* of *Normal*. To edit the *ID Class* of a person, click **Tools > People** in the Desktop Client. The following window is displayed:



Robin Pierce
Add External Id
Principal
Female Age:
Tags: CPQ=0.6761, FCQ=0.6744, FSQ=0.7857, ImageSize=240 x 262, RecognitionDate=Dec 14, 2017, 9:57:53 PM, , Source=7D7EFC49-46F5-53DF-

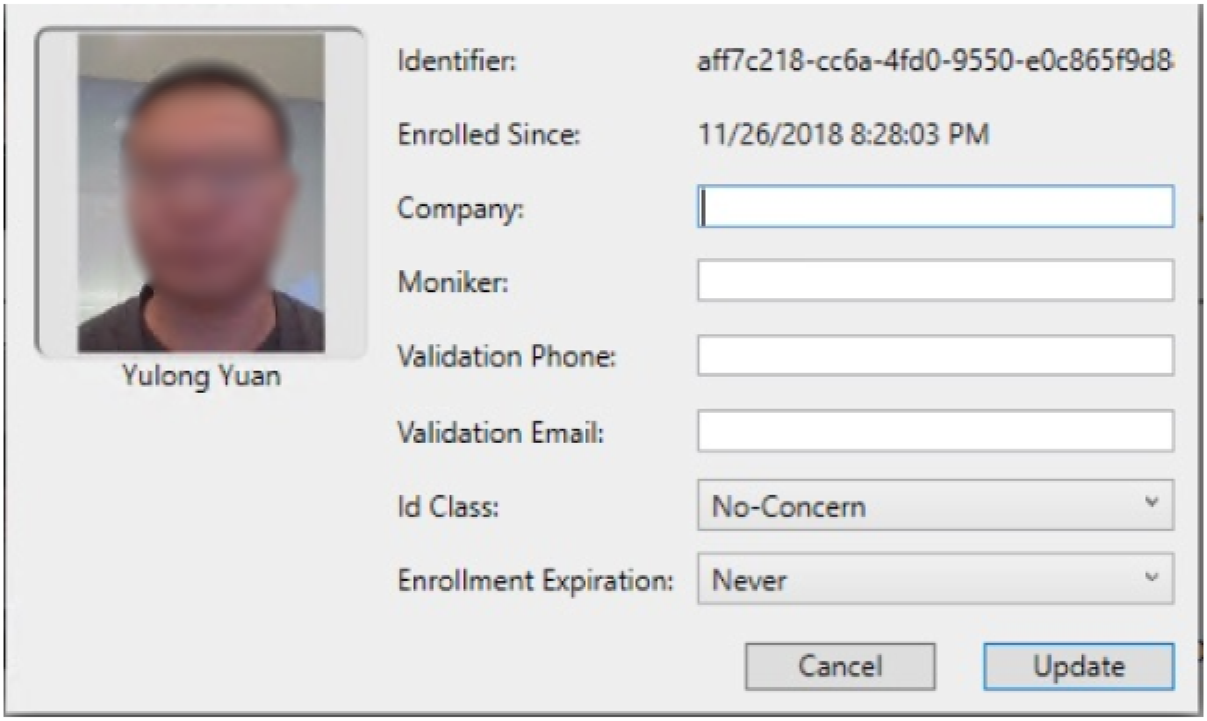
f14fbf62-5d00-4af6-a3ea-b65acadb802a
Gregg Clark
Add External Id
Teacher
Male Age:
Tags: CPQ=0.6830, FCQ=0.7333, FSQ=0.6992, ImageSize=240 x 344, RecognitionDate=Dec 14, 2017, 10:04:10 PM, , Source=7D7EFC49-46F5-53DF-

f283eda8-f5e7-42ea-a124-6c817ed63f83
Olivia Parkinson
Add External Id
Librarian
Female Age:

1f00e067-991f-48e3-91b4-585e118370a3
Sara Rodriguez
Add External Id
Teacher
Female Age:
Tags: CPQ=0.6017, FCQ=0.6133, FSQ=0.7177, ImageSize=128 x 154, RecognitionDate=Dec 14, 2017, 11:37:21 PM, , Source=7D7EFC49-46F5-53DF-

809776e4-649f-41ec-941b-d4a52252cdc1
Danilo Ocampo
Add External Id
Teacher
Male Age:

The *Person Type* and *Name* attributes can be edited by clicking their respective fields in the People window. To edit *ID Class*, in the **People Edit** dialog, double-click the user and choose the desired value from the *ID Class* menu as shown in the following image:



Yulong Yuan

Identifier: aff7c218-cc6a-4fd0-9550-e0c865f9d8

Enrolled Since: 11/26/2018 8:28:03 PM

Company:

Moniker:

Validation Phone:

Validation Email:

Id Class: No-Concern

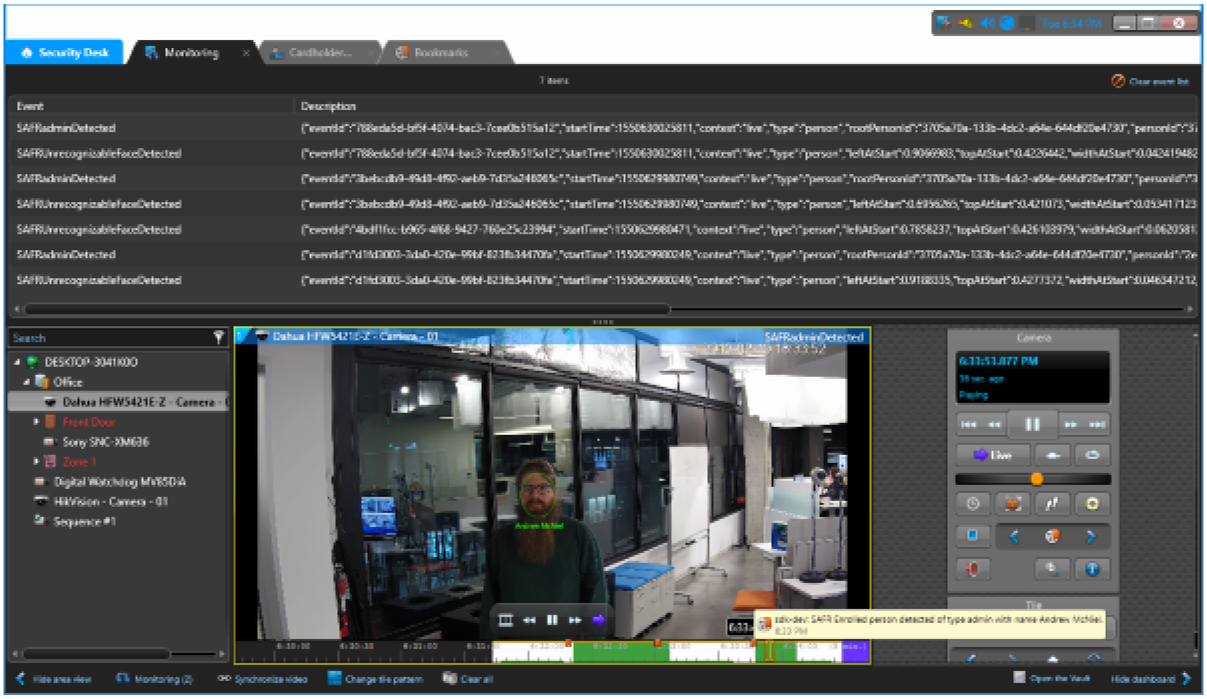
Enrollment Expiration: Never

110.4 SAFR Events

You can use SAFR to generate events. When enabled on the Events Preferences page, events occur when a person appears in the view of a connected camera. When a person appears in the camera's view, they're immediately assigned an *ID Class* attribute, although that *ID Class* may change if the system successfully recognizes them and assigns them a more appropriate *ID Class*. The types of *ID Classes* are listed below.

- Unrecognizable: A face was detected but it wasn't of sufficient quality to attempt recognition.
- Stranger. The person has been recognized, but they're not in Genetec's cardholder database nor in SAFR's Person Directory.
- Registered person without a name.
- Registered person with a name.
- Registered person marked as a *Concern*.
- Registered person marked as a *Threat*.

There are several different combinations of the conditions that are triggered. The following image shows multiple events populated in the Genetec alerts panel. Clicking any of the events allows the video from that event to be replayed:



7 items

Event	Description
SAFRadminDetected	{"eventId":"788eda5d-bf5f-4074-bac3-7cee0b515a12","startTime":1550630025811,"context":"live","type":"person","rootPersonId":"3705a70a-133b-4dc2-a64e-644df20e4730","personId":"3
SAFRUnrecognizableFaceDetected	{"eventId":"788eda5d-bf5f-4074-bac3-7cee0b515a12","startTime":1550630025811,"context":"live","type":"person","leftAtStart":0.9066983,"topAtStart":0.4226442,"widthAtStart":0.04241948
SAFRadminDetected	{"eventId":"3bebcdb9-49d8-4f92-aeb9-7d25a246065c","startTime":1550629980749,"context":"live","type":"person","rootPersonId":"3705a70a-133b-4dc2-a64e-644df20e4730","personId":"3
SAFRUnrecognizableFaceDetected	{"eventId":"3bebcdb9-49d8-4f92-aeb9-7d25a246065c","startTime":1550629980749,"context":"live","type":"person","leftAtStart":0.6956265,"topAtStart":0.421073,"widthAtStart":0.05341 7123
SAFRUnrecognizableFaceDetected	{"eventId":"4bdf1fcc-b965-4f68-9427-760e25c23994","startTime":1550629980471,"context":"live","type":"person","leftAtStart":0.7858237,"topAtStart":0.426103979,"widthAtStart":0.0620581
SAFRadminDetected	{"eventId":"d1fd3003-3da0-420e-99bf-823fb34470fa","startTime":1550629980249,"context":"live","type":"person","rootPersonId":"3705a70a-133b-4dc2-a64e-644df20e4730","personId":"2e
SAFRUnrecognizableFaceDetected	{"eventId":"d1fd3003-3da0-420e-99bf-823fb34470fa","startTime":1550629980249,"context":"live","type":"person","leftAtStart":0.9188335,"topAtStart":0.4277372,"widthAtStart":0.046347212,

sdic-dev: SAFR Enrolled person detected of type admin with name Andrew McNeil.
6:33 PM

The following table lists the available events that are SAFR makes available to Genetec.

Event Message	Id Class	Named	Person Type	Condition	People Attributes
Unrecognizable face detected	N/A	N/A	N/A	Face detected but insufficient information for recognition	idClass="unidentified"
Stranger detected	Stranger	N/A	N/A	Face detected but not found in registered people	idClass="stranger"
Registered person detected without name	Normal	No	None	Registered person without name or person type assigned	idClass="noconcern" && personType="" && name=""
Registered person detected with name <name>	Normal	Yes	None	Registered person with name but no person type	idClass="noconcern" && personType="" && name=<name>
Registered person detected of type <personType>	Normal	No	Defined	Registered person with person type but no name	idClass="noconcern" && personType=<personType> && name=""

Event Message	Id Class	Named	Person Type	Condition	People Attributes
Registered person detected of type <personType> with name <name>	Normal	Yes	Defined	Registered person with person type and name	idClass=“noconcern” && personType=<personType> && name=<name>
Concern person detected without a name	Concern	No	None	Same as above for Concern	idClass=“concern” && personType="" && name=""
Concern person detected with name <name>	Concern	Yes	None	Same as above for Concern	idClass=“concern” && personType="" && name=<name>
Concern person detected of type <personType>	Concern	No	Defined	Same as above for Concern	idClass=“concern” && personType=<personType> && name=""
Concern person detected of type <personType> with name <name>	Concern	Yes	Defined	Same as above for Concern	idClass=“concern” && personType=<personType> && name=<name
Threat person detected without a name	Threat	No	None	Same as above for Threat	idClass=“threat” && personType="" && name=""
Threat person detected with name <name>	Threat	Yes	None	Same as above for Threat	idClass=“threat” && personType="" && name=<name>
Threat person detected of type <personType>	Threat	No	Defined	Same as above for Threat	idClass=“threat” && personType=<personType> && name=""
Threat person detected of type <personType> with name <name>	Threat	Yes	Defined	Same as above for Threat	idClass=“threat” && personType=<personType> && name=<name>

110.4.1 SAFR Video Processing Modes

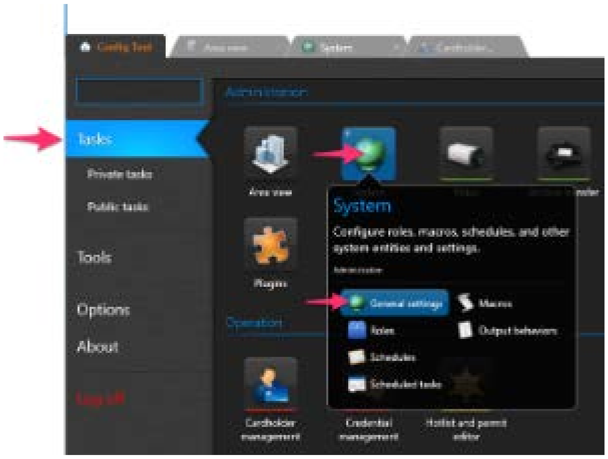
SAFR has different video processing modes that control what events are generated. Below is a short summary of the modes most relevant to Genetec Security Center integration. For a complete description, see Connect to a Video Feed in the *SAFR Documentation*.

- **Secure Access**: Only triggers events when cardholders and people registered in SAFR's Person Directory are identified with a high degree of certainty. This mode is useful when the system is being used to manage physical access. (i.e. when the system is responsible for unlocking doors for authenticated people)
- **Secure Access with Smile**: Similar to *Secure Access* mode, except that registered people must smile in order to cause the system to grant them access.
- **Enrolled Monitoring**: Similar to *Secure Access* mode, but events are triggered at a lower recognition confidence level.
- **Enrolled and Stranger Monitoring**: Similar to *Enrolled Monitoring* mode, but events are also triggered for strangers.

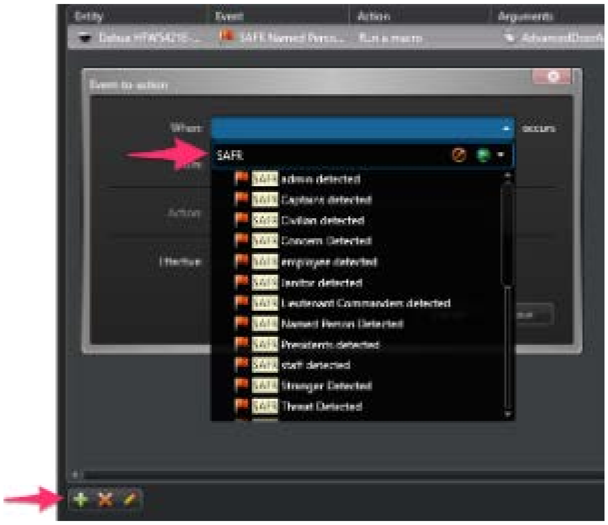
110.4.2 Add and Configure Alerts

To trigger the alert as a result of a SAFR-generated event, do the following:

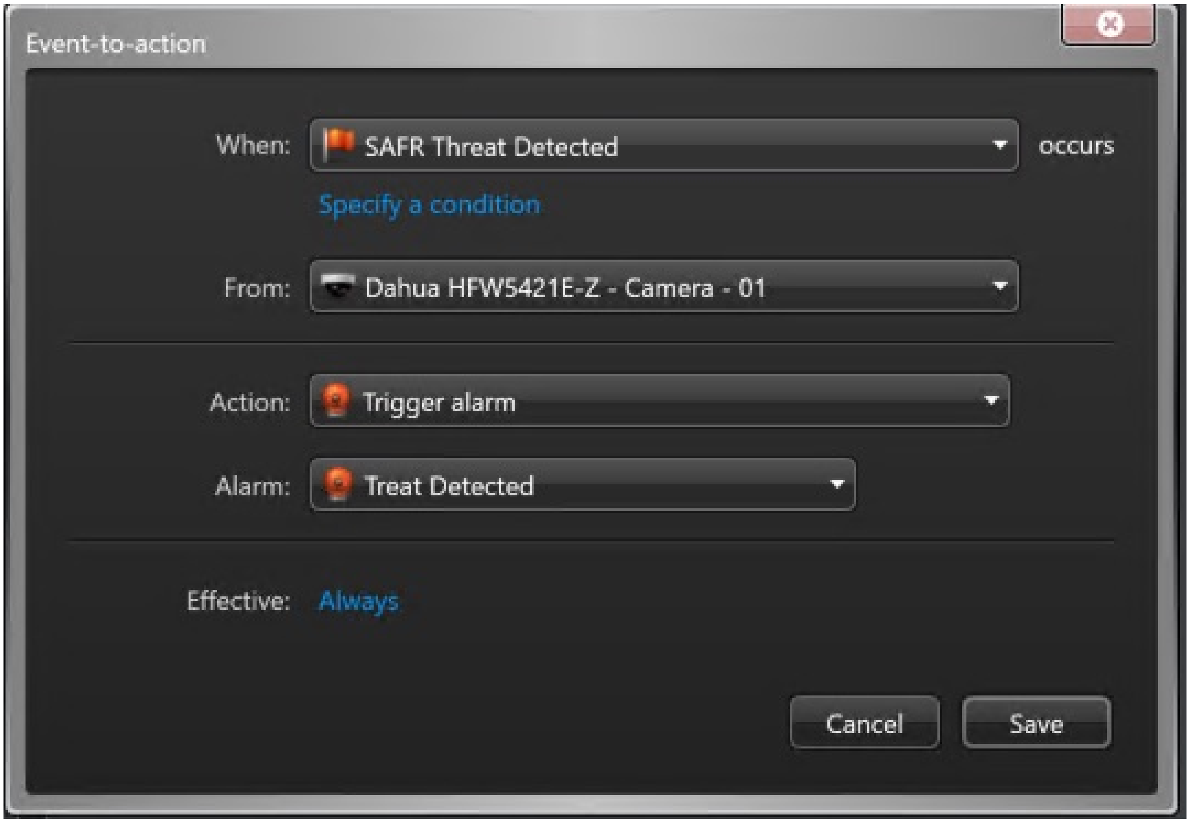
1. Open the Genetec Config Tool, and go to the System Panel.



2. Click the + icon to add a new alarm, and click the When menu. Type *SAFR* and press **Enter** to see the list of SAFR-enabled alarms.



3. Choose the desired entry from the list.
4. Under **From**, choose the camera you want to use to trigger the event. Under **Action**, choose a desired action. (e.g. Trigger Alarm)



Event-to-action

When: SAFR Threat Detected occurs

Specify a condition

From: Dahua HFW5421E-Z - Camera - 01

Action: Trigger alarm

Alarm: Treat Detected

Effective: Always

5. Click **Save** when done.

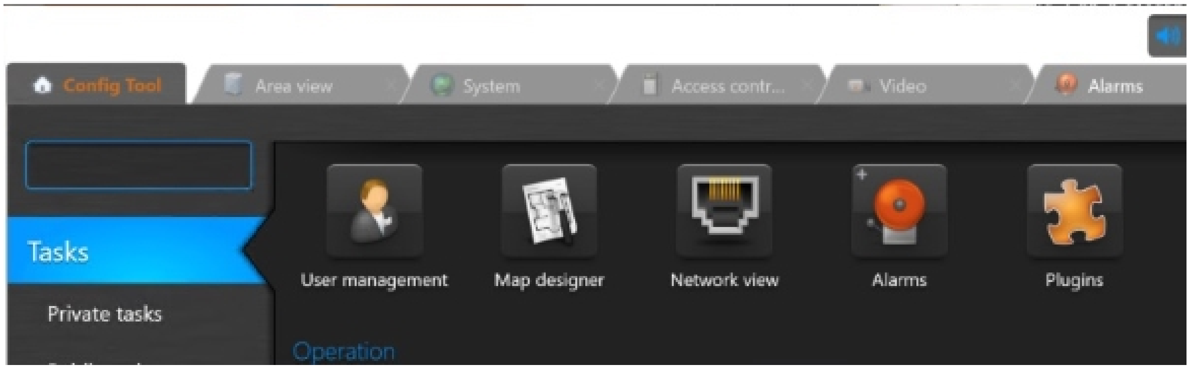
SAFR Events can be tied to Actions which can then trigger an Alarm. Initially create an alarm you want to trigger, and then use Genetec Event-to-Action dialog to tie SAFR Events to any action that can be defined in the Genetec system (for example, Trigger Alarm).

110.4.3 Add an Alarm

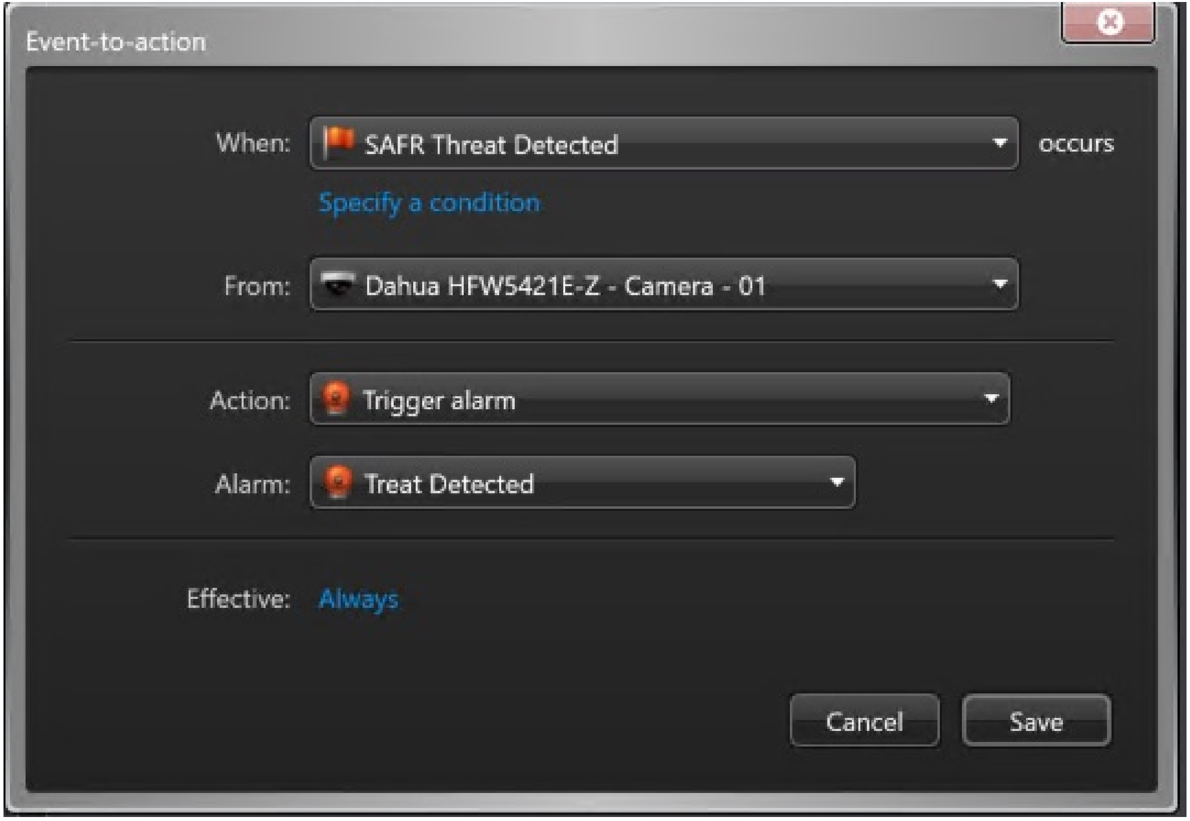
An alarm can be used to make sure an important event is noticed. In this example, we show how to create an alarm that is triggered when someone who has been marked as a threat is recognized on one of the cameras. For more information on triggering events, refer to Genetec support documentation.

To create an alarm, do the following:

1. Open the Genetec Config Tool, and open the Alarms screen.



2. Click **Add Alarm**, and type a *Name* as shown in the following image:


Event-to-action

When: SAFR Threat Detected occurs

Specify a condition

From: Dahua HFW5421E-Z - Camera - 01

Action: Trigger alarm

Alarm: Treat Detected

Effective: Always

3. Click **Save** to save the alarm.

110.4.4 Recommended Settings for Alarms

Properties task	Choose priorities based on circumstances and your organization guidelines (1=high, 255=low) • Stranger: 100 (If infrequent, set high) • Concern: 50 • Threat: 10 Video display option Set to Live to see the live view when alarm loads video Playback may be useful for short events where the subject may have walked off the screen by the time the video loads • If playback mode, set to at least 4 seconds to avoid buffering
Advanced task	Auto-Acknowledge: Good for stranger events; enter the number of seconds to stay in the view before returning to the view you were on prior to the event Choose color to match the SAFR colors (add ref to section in manual that describes colors) Reactivate threshold: Suppresses additional alarm if another similar alarm triggered within this time • Adjust as needed for use case.

110.4.5 Trigger Macros

When SAFR is configured to *Include Event Details* in reported events, highly customized actions can be programmed using macros in the Genetec system. Event details include all information associated with the detected face (e.g. *Name*, *Person Type*, *Age*, *Gender*, *Sentiment*, etc.). For more information on macros, refer to the Genetec support documentation.

110.5 Troubleshooting Tips

Note: When closing SAFR, use the **Quit SAFR** option on the **File** menu. Closing SAFR using the Window Close button will cause you to lose the SAFR state settings and connected cameras for that window.

- If detection or recognition is poor (not many faces found or recognized), make sure the Genetec video feeds are set for a sufficiently large frame size.
- If events are not being triggered, check the following:
  - Permissions are set correctly on Event-to-Actions.
  - Make sure the applicable SAFR Video Processing Mode is selected.

111 SAFR Geutebrueck Integration Guide

Integrated SAFR Geutebrueck is only available on Windows.

Integrating SAFR's facial recognition and analysis capabilities into Geutebrueck enables you to use SAFR's video feed information overlays within Geutebrueck camera video feeds, thus making it much easier to quickly and accurately separate unknown people from authorized people from known threats. You'll also have immediate access to additional infomation such as age, gender, sentiment, name, company, known associates, or any other configurable information you want to create.

Integrating the two systems also allows SAFR's information-rich events to trigger and populate Geutebrueck events, alerts, and videotagging/bookmarking.

A final benefit to integrating SAFR and Geutebrueck is that you'll be able to enroll individuals appearing in Geutebrueck cameras into SAFR's Identity Database.

111.1 Integration Overview

A typical deployment requires the following:

- A machine running Geutebrueck G-Core.
- One or more machines running the SAFR Desktop Client to process videos.
- If you're doing a SAFR on-premise deployment, you'll also need a machine running SAFR Server. SAFR Server can run on the same machine as one of the Desktop Clients, provided the host machine meets the system requirements.

Cameras are connected to the Geutebrueck G-Core. The SAFR Desktop Client can then connect to the G-Core to perform analysis of the video and add overlays. Depending on the number of cameras you need, one or more machines may be required to run the SAFR Desktop Client(s), each processing multiple video feeds. The Desktop Client is also used to perform various management activities.

111.2 Install and Configure Geutebrueck

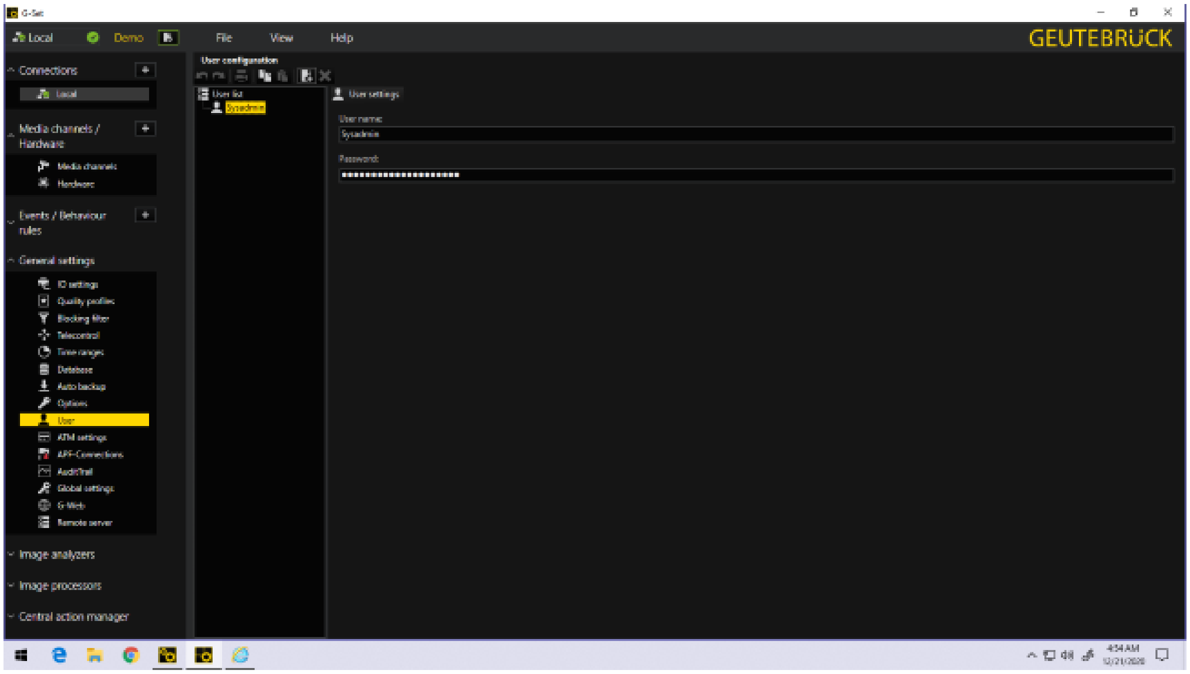
111.2.1 Install G-Core

1. Download the latest G-Core installer from the Geutebrueck Download Portal.
2. Run the installer, selecting the following options during installation:
   - Select all of the IP cameras.
   - Select **Plugin > GBF Streamer**.
   - Select **Plugin > Media Channel Simulator**.
   - Select **Plugin > Livestream Reader**.

111.2.2 Add a SAFR User Account

You'll need to create a user account within Geutebrueck that SAFR will use to connect to Geutebrueck. To do this, do the following:

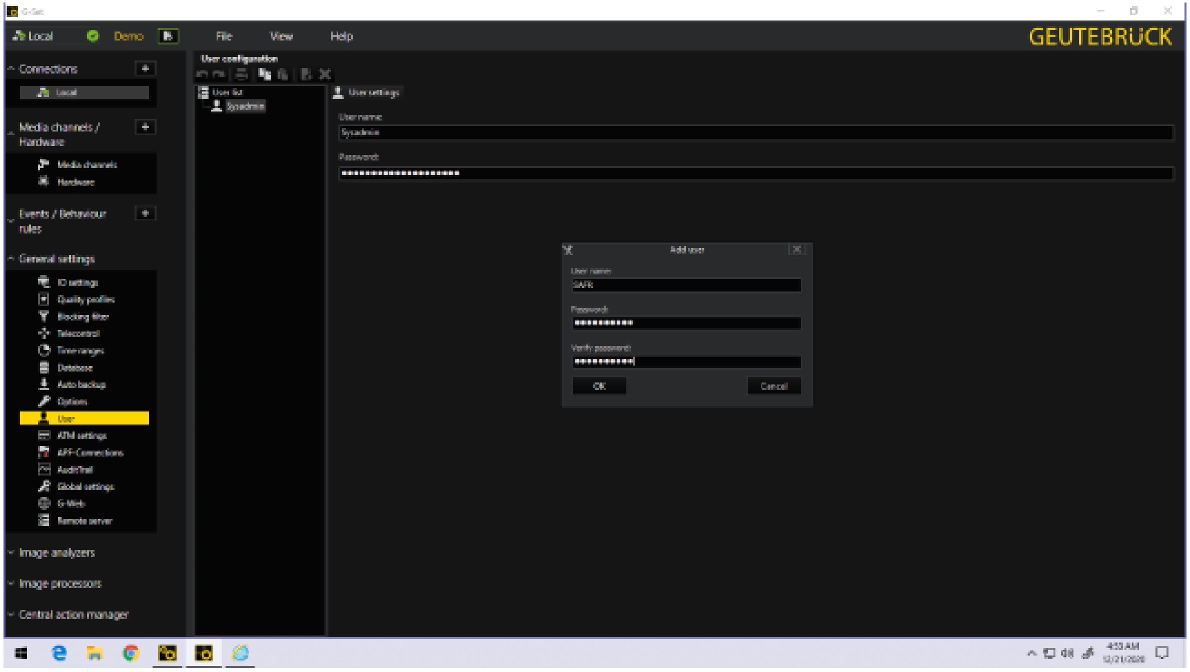
1. Open the **G-Set** application.
2. Go to **General settings -> User**.
3. On the **User configuration** window on the right, click on the **Add** button in the upper left corner of the window.



User name: Sysadmin

Password:

4. Enter a name and password for the new account.



User name: Sysadmin

Password:

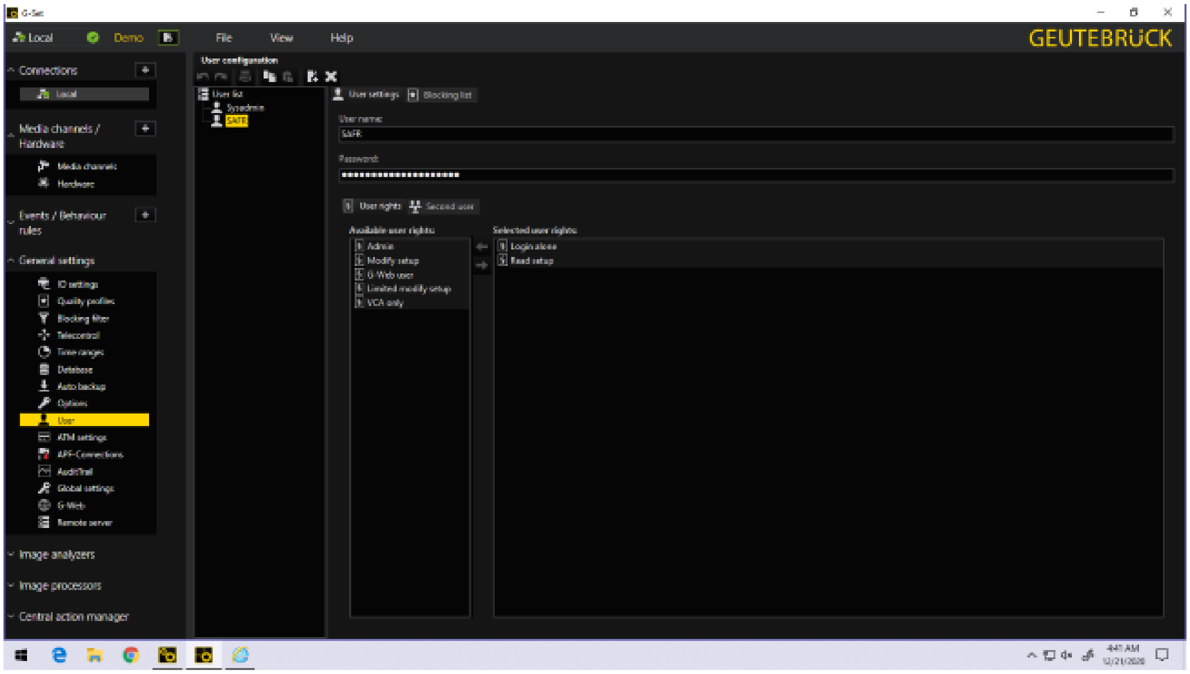
Add user

User name: SAFR

Password:

Verify password:

5. The new account will now be listed among the list of created user accounts.



User configuration

User name: SAFR

Password: ••••••••••••••••••••••

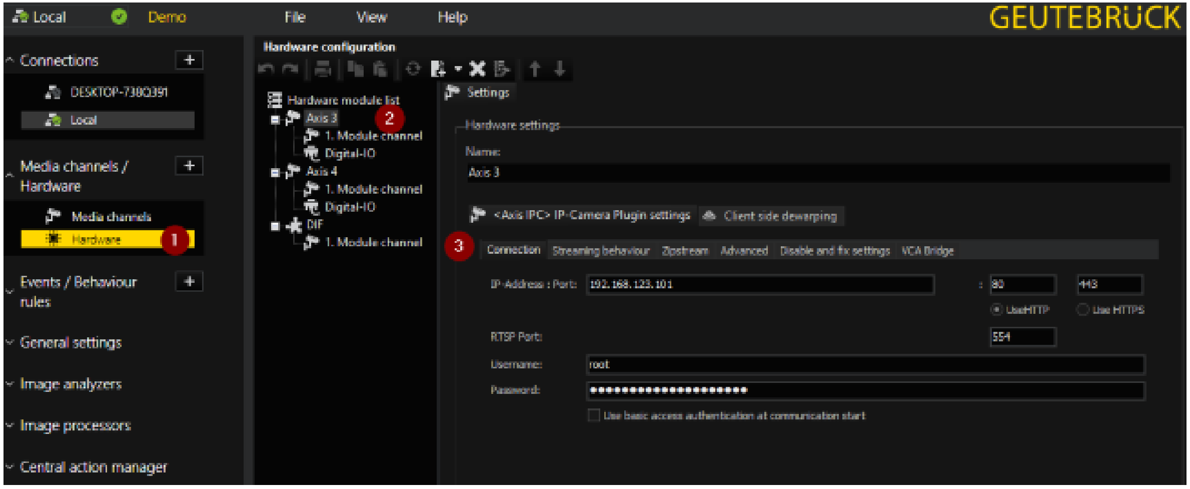
Available user rights: Admin, Modify setup, G-Web user, Limited modify setup, VCA only

Selected user rights: Login alone, Read setup

111.2.3 Add a Camera

To add and configure a new camera to Geutebrueck, do the following:

1. Open the **G-Set** application.
2. Go to **Media channels/Hardware -> Hardware**. (#1 below)
3. Click on the **Add** button at the top of the **Hardware configuration** window. Select the type of camera you want to add from the list of IP camera plugins. A new IP camera plugin entry will appear in the **Hardware module list**. (#2 below)
4. Click on your new camera plugin entry. The right pane wll now display various settings for the new plugin.
5. Enter the IP address for your camera as well as the credentials for its RTSP port, if applicable. (#3 below)



Hardware configuration

Name: Axis 3

IP-Address : Port: 192.168.123.101 : 80 443

○ UseHTTP ○ Use HTTPS

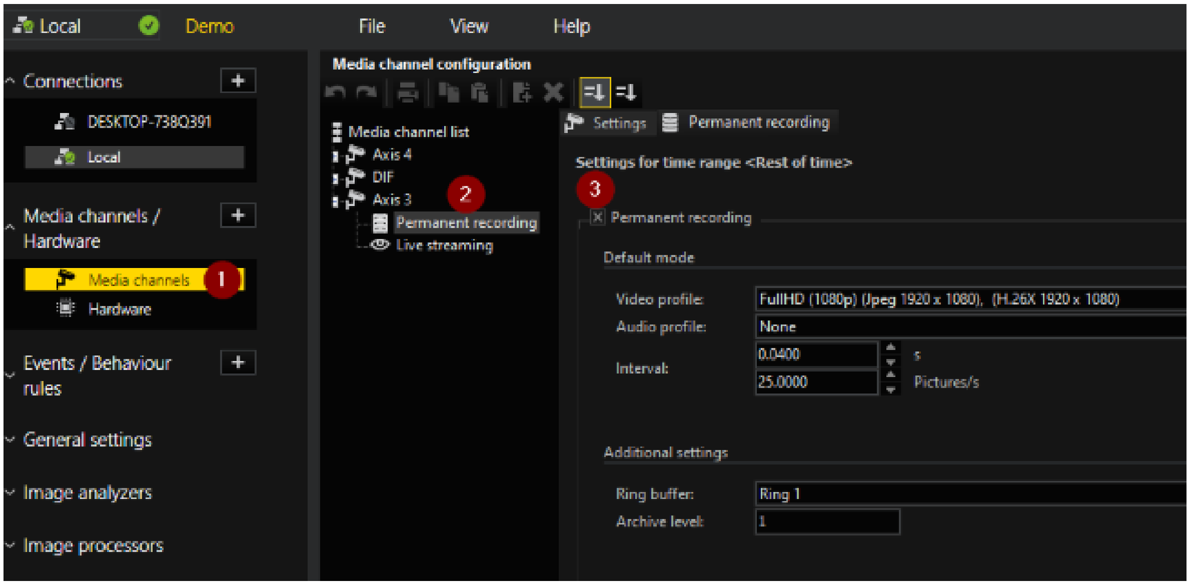
RTSP Port: 554

Username: root

Password: ••••••••••••••••••••

□ Use basic access authentication at communication start

6. Go to **Media channels/Hardware -> Media channels**. (#1 below)
7. Click on **Permanent recording** for the camera plugin you just added. (#2 below)
8. Tick the **Permanent recording** checkbox. (#3 below) This enables Geutebrueck to display reference images and event images.



Settings for time range <Rest of time>

☒ Permanent recording

Default mode

Video profile: FullHD (1080p) (Jpeg 1920 x 1080), (H.26X 1920 x 1080)

Audio profile: None

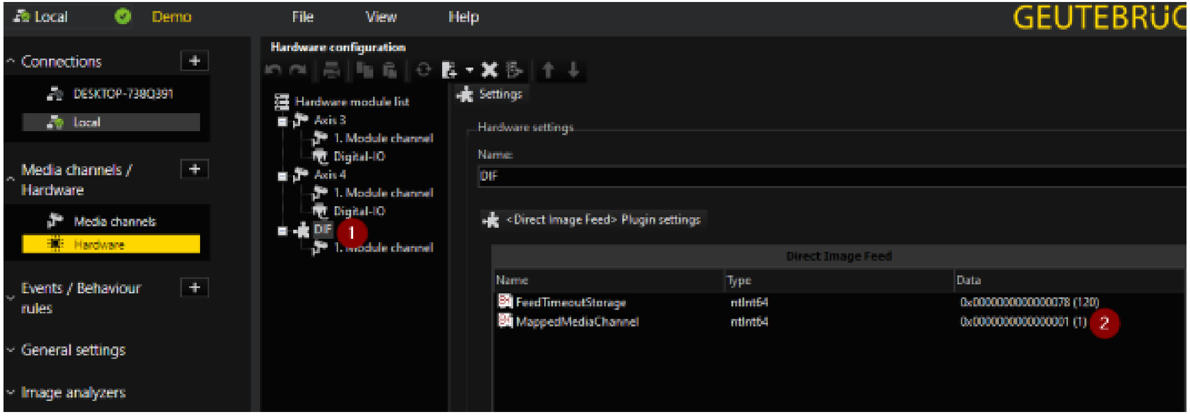
Interval: 0.0400 s
25.0000 Pictures/s

Additional settings

Ring buffer: Ring 1

Archive level: 1

9. Go to **Media channels/Hardware -> Hardware**.
10. Click on the **Add** button at the top of the **Hardware configuration** window. Select **Plugin <Direct Image Feed Plugin>** from the list of plugins, then press **Add**. A new IP plugin entry will appear in the **Hardware module list**. (#1 below)
11. Click on the new plugin.
12. Set the *MediaChannelID* of the *MappedMediaChannel* to the camera that's being used. (#2 below)



Hardware settings

Name: DIF

<Direct Image Feed> Plugin settings

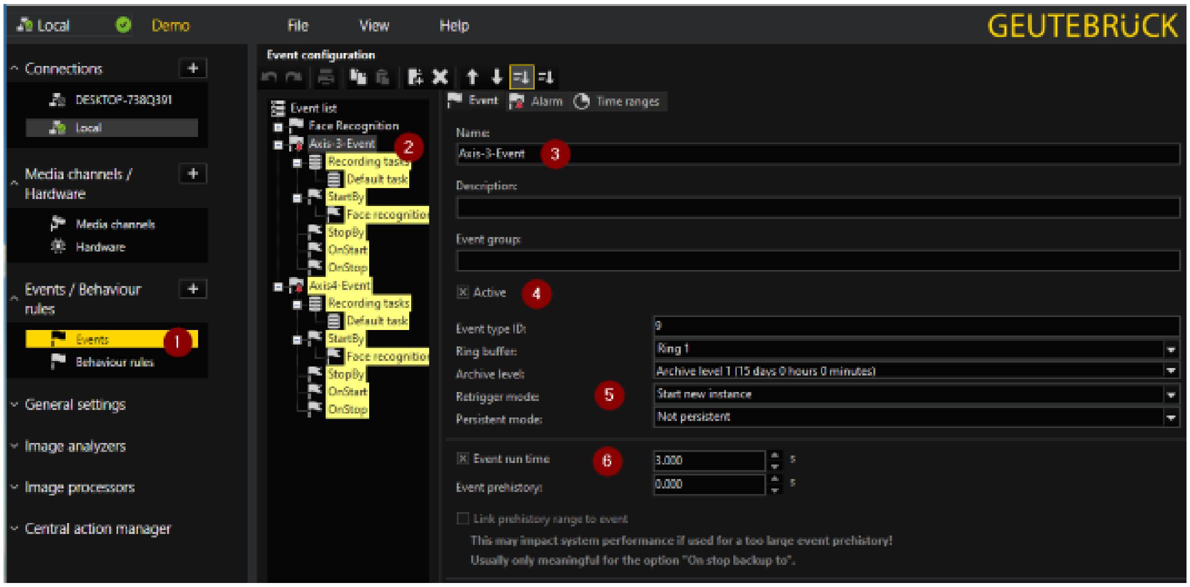
Direct Image Feed

Name	Type	Data
FeedTimeoutStorage	ntInt64	0x0000000000000078 (120)
MappedMediaChannel	ntInt64	0x0000000000000001 (1)

111.2.4 Configure Geutebrueck Events

To add and configure a new Geutebrueck event type, do the following:

1. Open the **G-Set** application.
2. Go to **Events/Behaviour rules > Events**, (#1 below)
3. Click on the **Add** button at the top of the **Event Configuraton** window. Your new event type will appear on the **Event list**. (#2 below)
4. Click on your new event type. The right pane wll now display various settings for your new event type.
5. Select the **Event** tab, if it's not already selected.
6. Enter a name for your new event type. (#3 below)
7. Tick the **Active** and **Event run time** checkboxes. (#4 and #6 below, respectively)
8. Select *Start new instance* from **Retrigger mode**'s drop-down menu. (#5 below)



Name: Axis-3-Event

Description:

Event group:

☒ Active

Event type ID: 9

Ring buffer: Ring 1

Archive level: Archive level 1 (15 days 0 hours 0 minutes)

Retrigger mode: Start new instance

Persistent mode: Not persistent

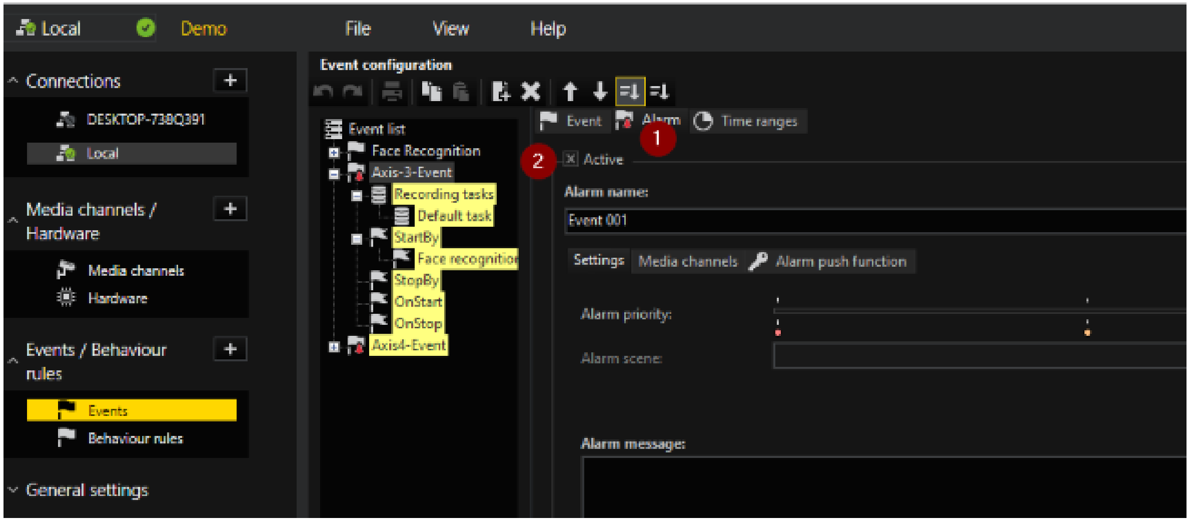
☒ Event run time 3.000 s

Event prehistory: 0.000 s

☐ Link prehistory range to event

This may impact system performance if used for a too large event prehistory!
Usually only meaningful for the option "On stop backup to".

9. Select the **Alarm** tab. (#1 below)
10. Tick the **Active** checkbox. (#2 below)



☒ Active

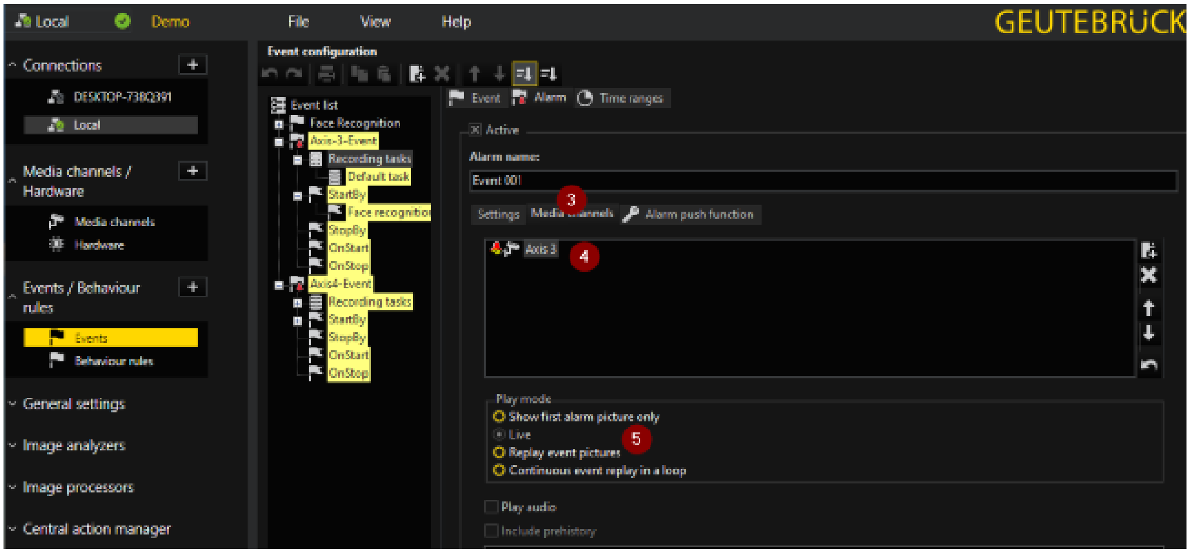
Alarm name: Event 001

Alarm priority:

Alarm scene:

Alarm message:

11. Select the **Media channels** tab. (#3 below)
12. Click on the **Add** button on the right of the **Media channels** window and select one of the listed cameras. The selected camera will now be a media channel for this alarm. (#4 below)
13. Tick the **Live** checkbox. (#5 below)

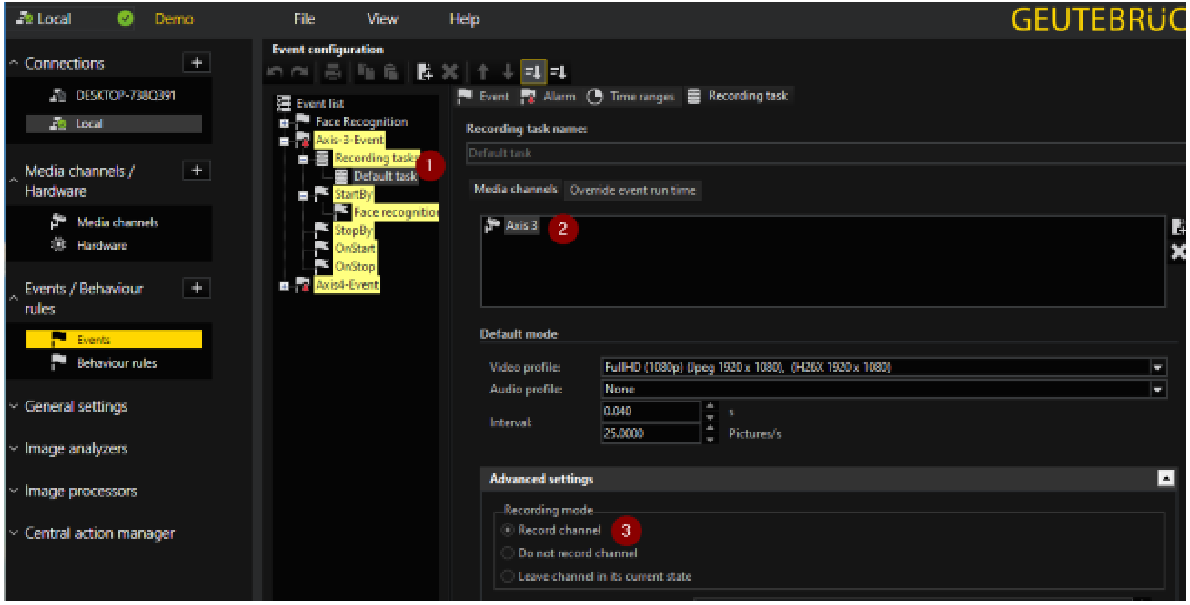


☒ Active

Alarm name: Event 001

Play mode: ● Live

14. In the **Event list**, click on **Recording tasks** for your new event type. (#1 below) This will cause a new **Recording task** tab to appear on the right pane. Select that new tab.
15. Select the **Media channels** tab, if not already selected.
16. Click on the **Add** button on the right of the **Media channels** window and select one of the listed cameras. The selected camera will now be listed in the **Media channels** window. (#2 below)
17. Tick the **Record channel** checkbox. (#3 below)



Recording task name: Default task

Video profile: FullHD (1080p) (Jpeg 1920 x 1080), (H26X 1920 x 1080)

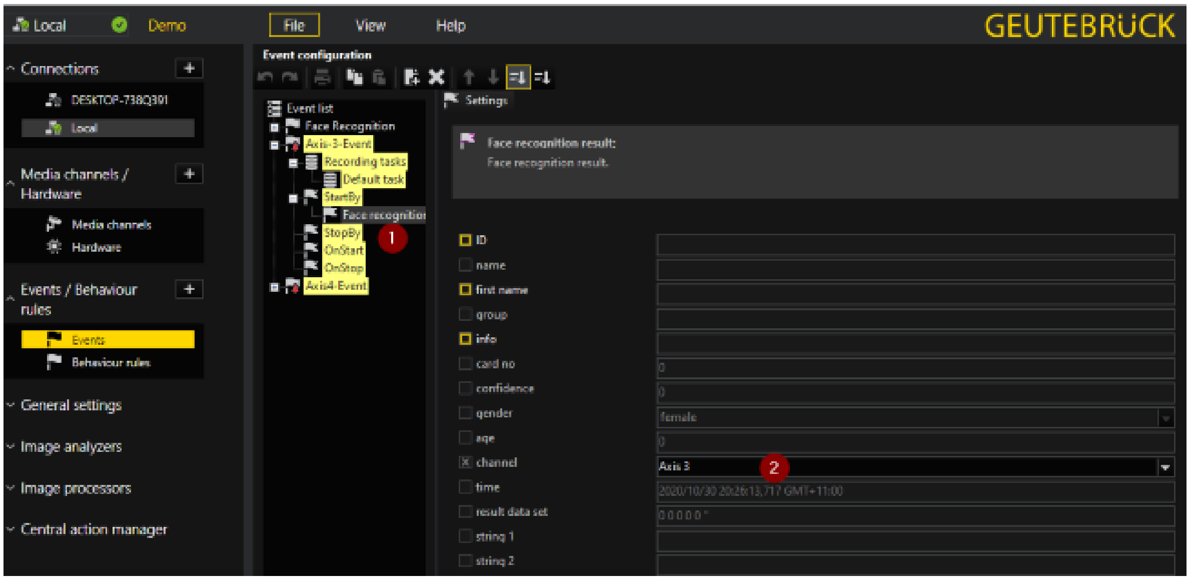
Audio profile: None

Interval: 0.040 s

25.0000 Pictures/s

Recording mode: ● Record channel

18. In the **Event list**, click on **StartBy** for your new event type. (#1 below)
19. Click on the **Add** button near the top of the **Event configuration** window.
20. You will be prompted to select whivch action to add. Choose *Face Recognition Result*.
21. In the **Event list**, click on the newly added *Face Recognition Result*.
22. In the **Settings** window in the right pane, tick the **channel** checkbox.
23. Select the media channel that you added in the previous steps from **channel**'s drop-down menu. (#2 below)



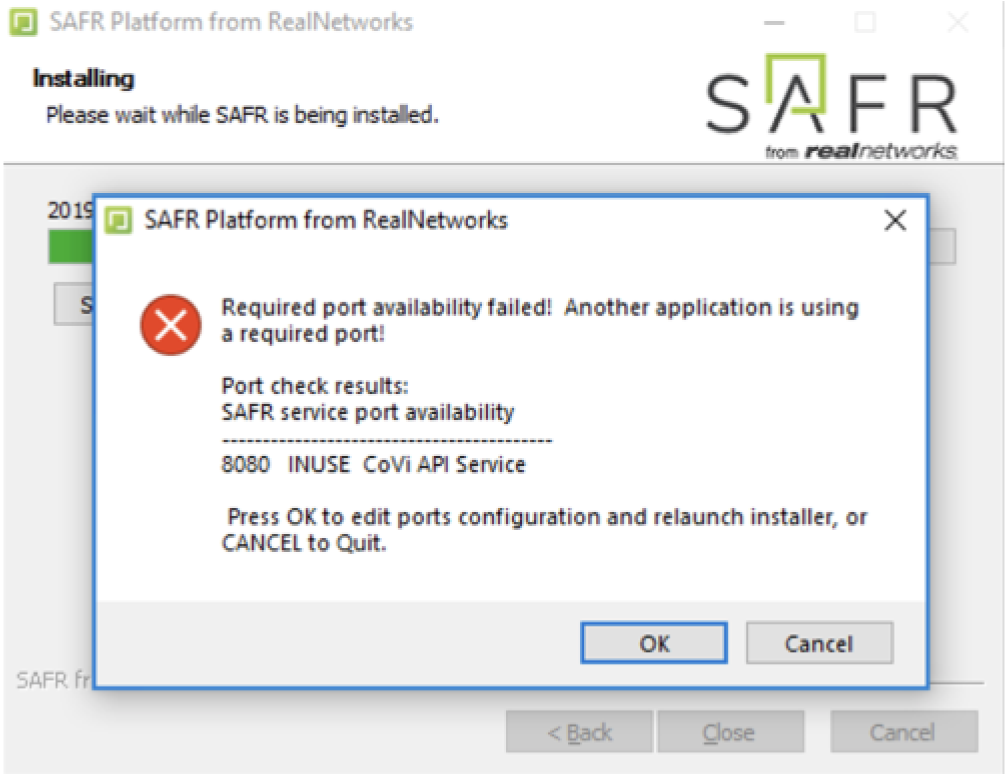
Face recognition result:
Face recognition result.

□ ID
□ name
□ first name
□ group
□ info
□ card no: 0
□ confidence: 0
□ gender: female
□ age: 0
☒ channel: Axis 3
□ time: 2020/10/30 20:26:13,717 GMT+11:00
□ result data set: 0 0 0 0 0 *
□ string 1
□ string 2

111.3 Install and Configure SAFR

To install SAFR, do the following:

1. Go to the SAFR Download Portal.
2. If you're doing a cloud deployment, download and install Windows SAFR Desktop. Make sure to select the Geutebrueck install option.
3. If you're doing an on-premise deployment, download and install Windows SAFR Platform. Make sure to select the Geutebrueck install option.
    - When installing the SAFR Platform, the default SAFR port assignments sometimes conflict with other software port assignments. If a port conflict occurs, the error message shown below will pop up in the middle of your installation.



SAFR Platform from RealNetworks
Installing
Please wait while SAFR is being installed.
SAFR from realnetworks
2019
SAFR fr
< Back
Close
Cancel
SAFR Platform from RealNetworks
OK
Cancel

Required port availability failed! Another application is using a required port!

Port check results:
SAFR service port availability

```
---------------------------------------------
8080   INUSE  CoVi API Service
```

Press OK to edit ports configuration and relaunch installer, or CANCEL to Quit.

If this happens, do the following:

1. Click **OK** to edit port configurations.
2. **Notepad** will open, displaying the *safrports.conf* file.
3. Edit any conflicting ports to new values. (e.g. CoviHTTP=18080)
4. Save and exit **Notepad**.

The Platform installer will then restart and the new port values will be used. You can find the modified *safrports.conf* file at `C:\Program Files\RealNetworks\SAFR\`.

After the installation finishes, two icons will appear on your desktop: one labeled *SAFRActions* and another labeled *SAFR*. *SAFRActions* launches SAFR Actions, while *SAFR* launches the Desktop Client. The SAFR Server (when installed as part of a local deployment) automatically runs as a collection of background services.

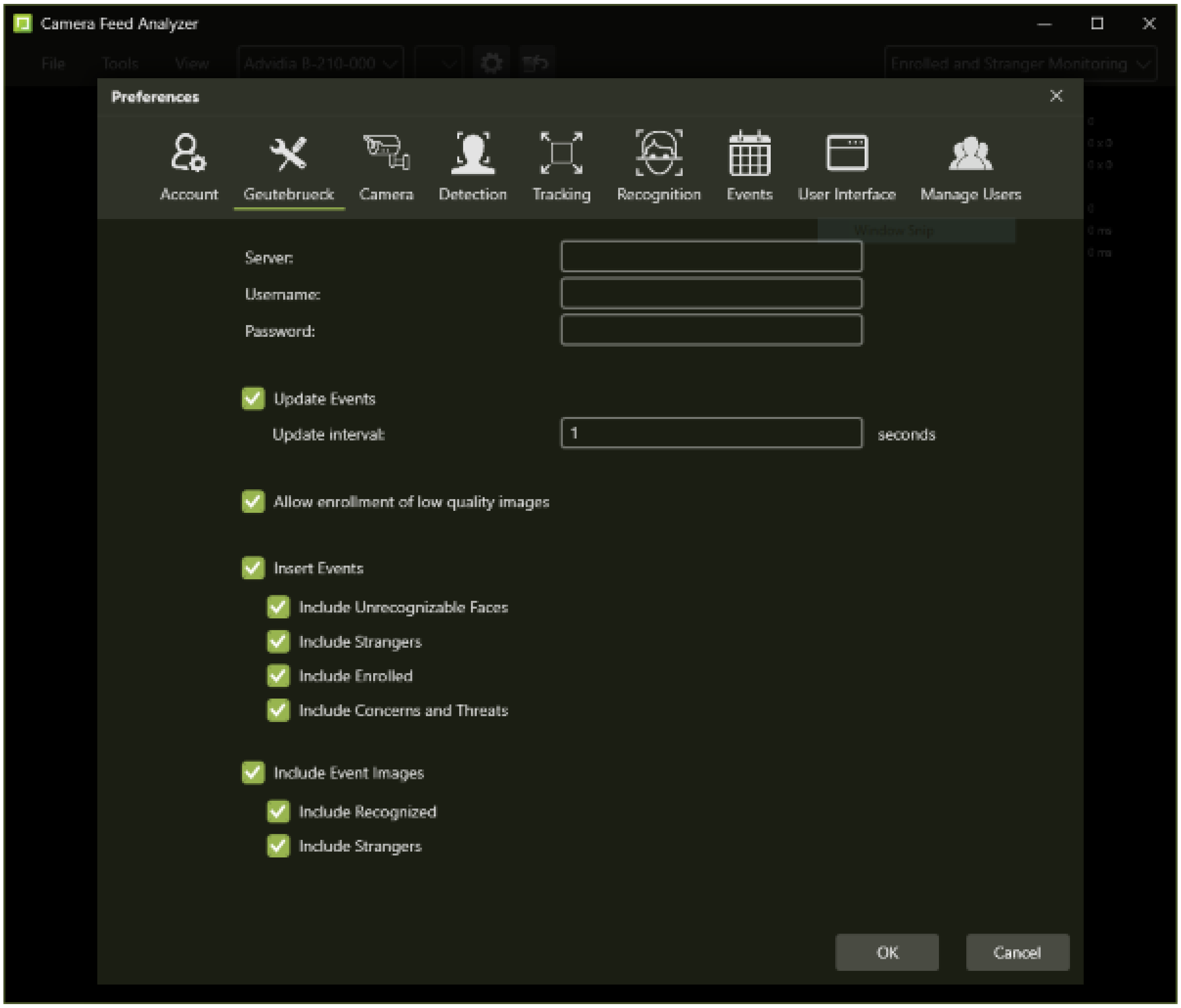
Immediately following installation, the installer opens the Desktop Client and prompts you to log in with your SAFR Account credentials. Make sure to log in; it's important in acquiring the SAFR license.

111.3.1 Connect SAFR to Geutebrueck

To connect SAFR to Geutebrueck, do the following:

1. Within your SAFR Desktop Client, select **Tools->Preferences->Geutebrueck**.
2. Enter information for the first three settings on the SAFR Geutebrueck Preferences menu: *Server*, *Username*, and *Password*. See below for information about the available SAFR Geutebrueck preference settings.

111.3.1.1 SAFR Geutebrueck Preferences You can configure several Geutebrueck-specific settings from the **Tools->Preferences->Geutebrueck** menu.



Server:

Username:

Password:

☑ Update Events

Update interval: 1 seconds

☑ Allow enrollment of low quality images

☑ Insert Events

☑ Include Unrecognizable Faces

☑ Include Strangers

☑ Include Enrolled

☑ Include Concerns and Threats

☑ Include Event Images

☑ Include Recognized

☑ Include Strangers

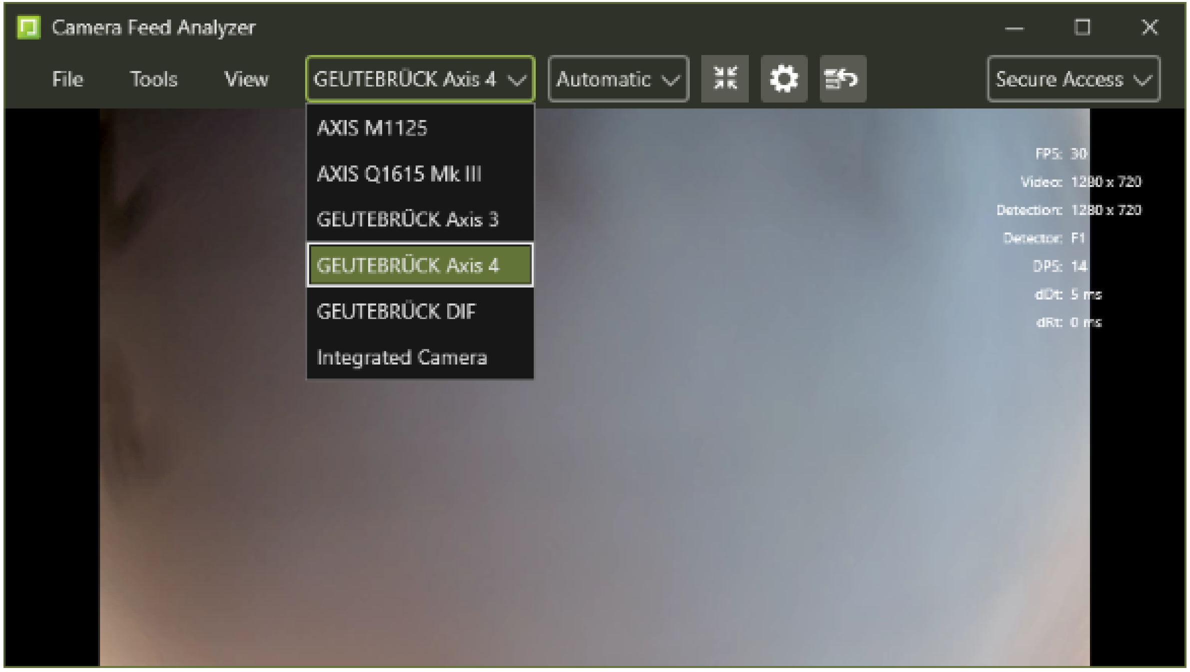
- **Server**: IP address or hostname of a Geutebrueck server.
- **Username**: Username of a Geutebrueck user that has sufficient permissions to allow SAFR to connect to the Geutebrueck system. See the Add a SAFR User Account section above for information about how to create and find this information.
- **Password**: Password of a Geutebrueck user that has sufficient permissions to allow SAFR to connect to the Geutebrueck system. See the Add a SAFR User Account section above for information about how to create and find this information.
- **Update Events**: Specifies if generated events should be updated if additional information is obtained. (e.g. A previously unidentified face becomes recognized, possibly due to improved lighting conditions, thus changing a Stranger Event to a Known Person Event.)
  - **Update Interval**: If *Update Events* is enabled, this setting specifies how frequently events are updated.
- **Allow enrollment of low quality images**: Enables enrollment of people with low quality facial images. See the Image Quality Metrics Guidance page in the SAFR documentation for information about face image quality.
- **Insert Events**: Enables insertion of events into Geutebrueck.
  - **Include Unrecognizable Faces**: Enables the inclusion of events triggered by unrecognizable faces.
  - **Include Strangers**: Enables the inclusion of events triggered by strangers.
  - **Include Enrolled**: Enables the inclusion of events triggered by enrolled persons that aren't persons of concern or known threats.

  - **Include Concerns and Threats**: Enables the inclusion of events triggered by persons of concern or known threats.
- **Include Event Images**: Enables the inclusion of event images.
  - **Include Recognized**: Enables reference and event face images for events triggered by recognized persons.
  - **Include Strangers**: Enables events face images for events triggered by strangers.

111.3.2 Verify the Connection to Geutebrueck

To verify that your SAFR system successfully connected to the Geutebrueck system, do the following:

1. Open the Camera Feed Analyzer in the SAFR Desktop Client.
2. Click on the camera selection drop-down menu at the top of the *Camera Feed Analyzer*.



3. You should see the cameras connected to Geutebrueck among the available camera. **Note**: All Geutebrueck cameras will have "Geutebrueck" appended to the beginning of the camera name.
   - If you don't see any Geutebrueck cameras listed, try shutting down and then restarting your SAFR Desktop Client.

112 SAFR Geutebrueck Operation Guide

112.1 Person Enrollment

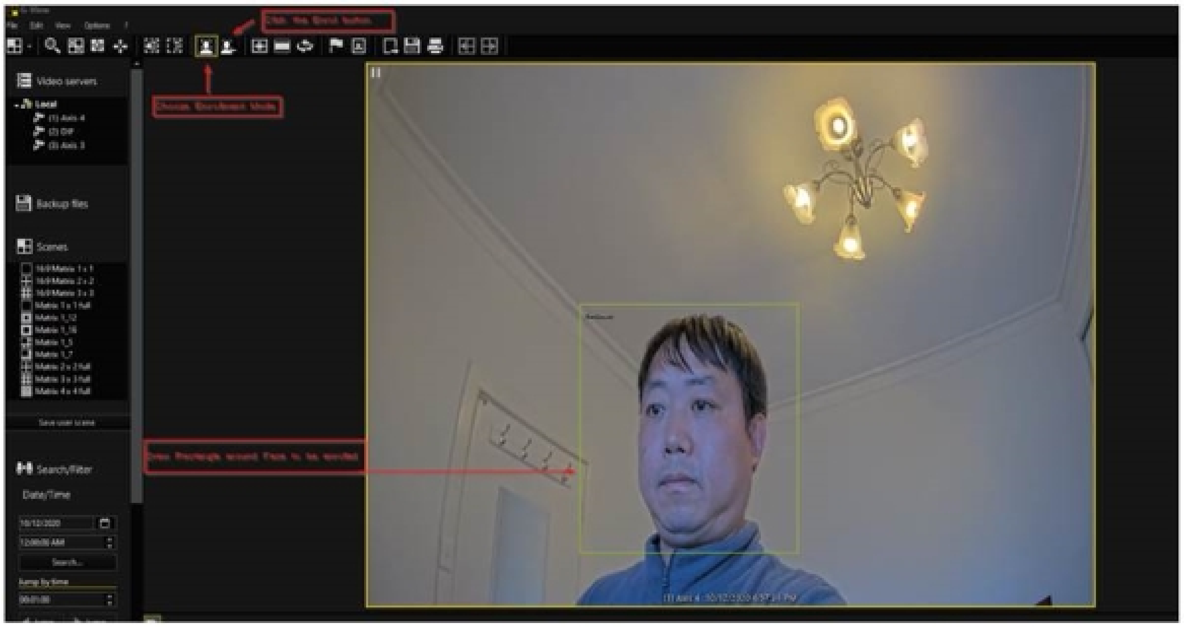
Most of the integrated SAFR-Geutebrueck functionality requires that people are first enrolled in SAFR's Identity Database. There are 3 main ways that people can become enrolled in your SAFR system:

- Import people into SAFR using photos of faces.
- Import people into SAFR using video files. (e.g. saved CCTV footage)
- Enroll people into SAFR from within the Geutebrueck G-View client, as described in the section below.

112.1.1 Enroll People from Within Geutebrueck

To enroll people into SAFR from within Geutebrueck, do the following:

1. Open the Geutebrueck G-View client and run a recorded CCTV video file.



Click the Enroll button.

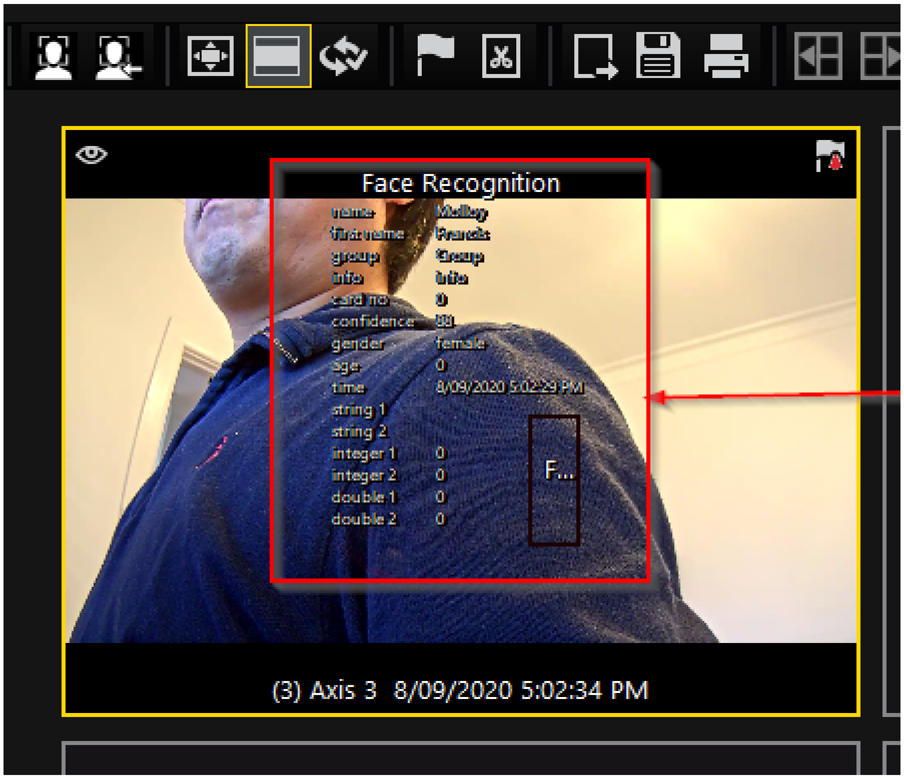
Choose 'Enrollment' Mode

Draw Rectangle around Face to be enrolled

2. Click the button to enter *Face Enrollment* mode.
3. Draw a rectangle around the face to be enrolled.
4. Click the **Enroll** button. This will cause the person's face to be enrolled in SAFR.

112.2 Video Overlays

Geutebrueck supports displaying real-time overlays in the G-View Client and archived video on a periodic basis. The frequency and duration of display is managed in the G-View preferences. When SAFR and Geutebrueck are integrated, Geutebrueck's video overlays display additional information.



Face Recognition

name	Medley
first name	Francis
group	Group
info	info
card no	0
confidence	88
gender	female
age	0
time	8/09/2020 5:02:29 PM
string 1	
string 2	
integer 1	0
integer 2	0
double 1	0
double 2	0

(3) Axis 3 8/09/2020 5:02:34 PM

Video Insight Overlay Item	SAFR Equivalent	Additional Notes
ID	PersonId	If the person isn't enrolled in SAFR, this field is blank. (Persons who aren't enrolled don't have PersonIds.)
Name	Last Name	
FirstName	First Name	
Group	Person Type	
Info	Tags	This item is a boolean. It's set to `true` if the viewed person is enrolled in SAFR and has one or more tags defined. Otherwise, it's set to `false`.
CardNo	N/A	This item currently isn't used for anything.
Confidence	similarityScore	
Gender	Gender	This item is an integer; its possible values represent the following: 0 = Female 1 = Male 2 = Unknown
Age	Age	
Time	DateTimeOffset	
String 1	External Id	
String 2	Id Class	This item is a text string.

Video Insight Overlay Item	SAFR Equivalent	Additional Notes
Integer1	Id Class	This item is an integer; its possible values represent the following SAFR Id Classes: 0 = Unknown 1 = Unidentified 2 = Stranger 3 = NoConcern 4 = Concern 5 = Threat
Integer2	Sentiment value	
Double1	Mask	This item is an integer; its possible values represent the following: 0 = no mask 1 = masked face
Double2	SAFR Event Type	This item is an integer; its possible values represent the following SAFR event types: 0 = Unknown 1 = Person 2 = Badge 3 = Action 4 = RecognizedObject

See the SAFR documentation for details about the additional overlay information.

113 SAFR Milestone Integration Guide

Integrated SAFR Milestone is only available on Windows.

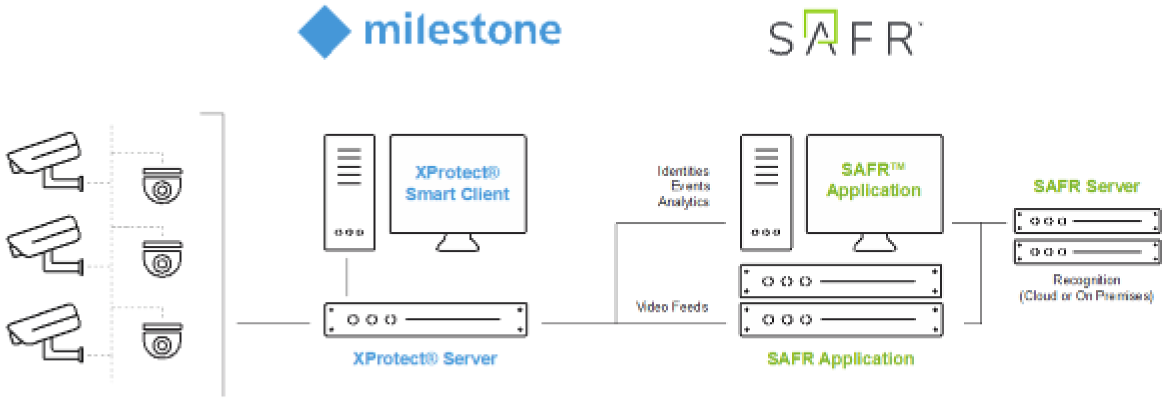
Integrating SAFR's facial recognition and analysis capabilities into Milestone enables you to use SAFR's video feed information overlays within Milestone camera video feeds, thus making it much easier to quickly and accurately separate unknown people from authorized people from known threats. You'll also have immediate access to additional infomation such as age, gender, sentiment, name, company, known associates, or any other configurable information you want to create.

Integrating the two systems also allows SAFR's information about individuals to trigger Milestone alerts and other actions within the Milestone system. Milestone's metadata within bookmarks are enriched with SAFR's additional information, allowing you to more easily find relevant bookmarks.

113.1 Integration Overview and Requirements

A typical deployment requires the following:

- A machine running Milestone XProtect. (Bookmarks are only supported on XProtect Expert and Corporate.)
- Machines running Milestone XProtect Smart Client and Milestone Admin Tool for monitoring and administration of Milestone.
- One or more machines running the SAFR Desktop Client to process videos.
- If you're doing a SAFR local deployment, you'll also need a machine running SAFR Server. SAFR Server can run on the same machine as one of the Desktop Clients, provided the host machine meets the system requirements.



Cameras are connected to Milestone XProtect. SAFR can then connect to Milestone XProtect to perform analysis of the video and add overlays. Depending on the number of cameras you need, one or more machines can run the SAFR Desktop Client, with each client processing multiple video feeds. SAFR processes the video and returns information to Milestone to overlay the video feeds and generate events. The SAFR Desktop Client is also used to perform various management activities.

113.1.1 System Requirements

Milestone has the following requirements:

- Milestone XProtect 2019 R1 or later must be installed.
- Each camera connected to Milestone requires a Milestone license.
- For each camera connected to Milestone on which you want to run SAFR overlays, you'll need a second Milestone license. Thus, each Milestone camera running SAFR overlays requires 2 Milestone licenses total.

SAFR has the following requirements:

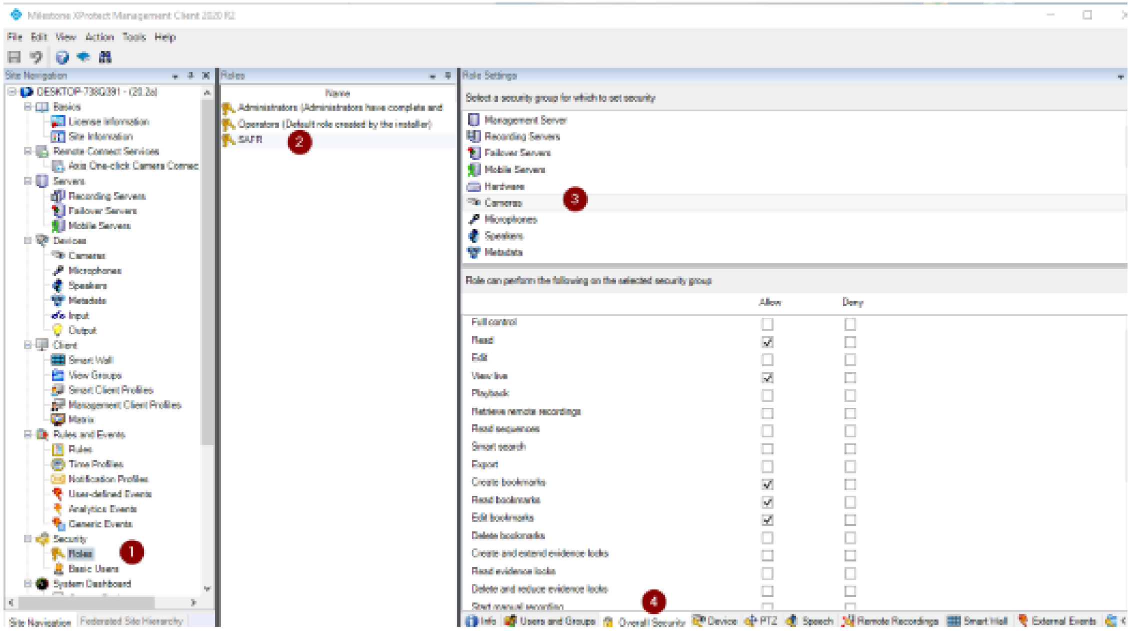
- Each camera running SAFR must have a SAFR license.
- Each machine running the SAFR Desktop Client must meet the following requirements:
  - The Desktop Client must be version 1.4.142 or later.
  - The system requirements described here.
- Local SAFR deployments require at least one machine running SAFR Platform 1.4.140 or later.
- Each machine running SAFR Platform must meet the system requirements described here.

113.2 Install and Configure Milestone XProtect

Download and install the latest Milestone installer package from the Milestone Download Portal.

To create a SAFR user in Milestone and set its permissions, do the following:

1. Add a SAFR user (e.g. `safr-roles`) in the Milestone XProtect Management Client by going to the Site Navigation pane and selecting **Security > Roles**.
2. Highlight **Administrator** in the **Roles** pane.
3. At the bottom of the GUI click **Users and Gr** and then **Add**. (Select **Basic** to add the `safr-roles`.)
4. Open the Milestone XProtect Management Client, go to **Security > Roles > SAFR > Camera**, and check **Allow** for the following checkboxes:
   - Read
   - View Live
   - Create bookmarks (only available on XProtect Expert and Corporate)
   - Read bookmarks (only available on XProtect Expert and Corporate)
   - Edit bookmarks (only available on XProtect Expert and Corporate)



Role can perform the following on the selected security group

	Allow	Deny
Full control	☐	☐
Read	☑	☐
Edit	☐	☐
View live	☑	☐
Playback	☐	☐
Retrieve remote recordings	☐	☐
Read sequences	☐	☐
Smart search	☐	☐
Export	☐	☐
Create bookmarks	☑	☐
Read bookmarks	☑	☐
Edit bookmarks	☑	☐
Delete bookmarks	☐	☐
Create and extend evidence locks	☐	☐
Read evidence locks	☐	☐
Delete and reduce evidence locks	☐	☐
Start manual recording	☐	☐

5. Click **Save** to save the changes.

113.2.1 Update Milestone XProtect Operator Permissions

To enable the Milestone operators to view SAFR-created overlays, do the following:

1. Open the Milestone XProtect Management Client, go to **Security > Roles > Operators > Metadata**, and check **Allow** for the following checkboxes:
   - Read
   - View Live
   - Playback

2. Go to **Security > Roles > Operators > Analytics Events** and check **Allow** for the following checkbox:
   - Read
3. Go to **Security > Roles > Operators > Alarms** and check **Allow** for the following checkboxes:
   - View
   - Receive notifications
4. Click **Save** to save the changes.

Note: Overlays are not visible if the live permission is not added to the Operator role.

113.3 Install and Configure SAFR

1. From the SAFR Download Portal, download and install either SAFR Platform or SAFR Desktop, depending on your deployment type. Make sure to select the Milestone VMS extension install option.
2. Start the Desktop Client and go to **Tools > Preferences**.
3. Click the **Milestone** tab.
   Note: If the **Milestone** Preferences tab is not displayed, it's possible that you didn't select the Milestone VMS Extension when you installed SAFR.
4. To connect to the Milestone XProtect server and enable access to the cameras connected to Milestone server enter the following.
   - **Username**: SAFR user created when you installed and configured Milestone above. (e.g. `safr-roles`)
   - **Password**: Password created for the SAFR user.
   - If you created the user (in this example `safr-roles`) as a Windows user versus basic user, check the "Windows credentials" box.
   - **Directory**: IP address or domain address of server running Milestone XProtect. If all in one server for a small deployment or PoC, "localhost" should work.

See the Operation Guide for a complete description of the settings on the Milestone Preferences tab.

113.3.1 Customizing Ports for SAFR Server Services

For smaller deployments in which you want to run both SAFR and Milestone XProtect on the same machine, you must customize the port assignments in SAFR to ensure SAFR and Milestone XProtect do not conflict.

SAFR uses the following ports by default:

- **COVI**: 8080
- **Event**: 8082
- **VIRGA**: 8084
- **CVOS**: 8086

To customize ports, do the following:

1. Stop or disable any conflicting software using the required ports.
2. Install the SAFR Platform.
3. Open *Notepad* as Administrator and open `C:\Program Files\RealNetworks\SAFR\safrports.conf`.
4. Edit any conflicting ports to new values. (e.g. CoviHTTP=18080)
5. Run `C:\Program Files\RealNetworks\SAFR\bin\configure-ports.bat`. Running this batch script stops, reconfigures, and re-starts SAFR services.
6. Run `C:\Program Files\RealNetworks\SAFR\bin\check.bat`. `check.bat` displays the new port settings.

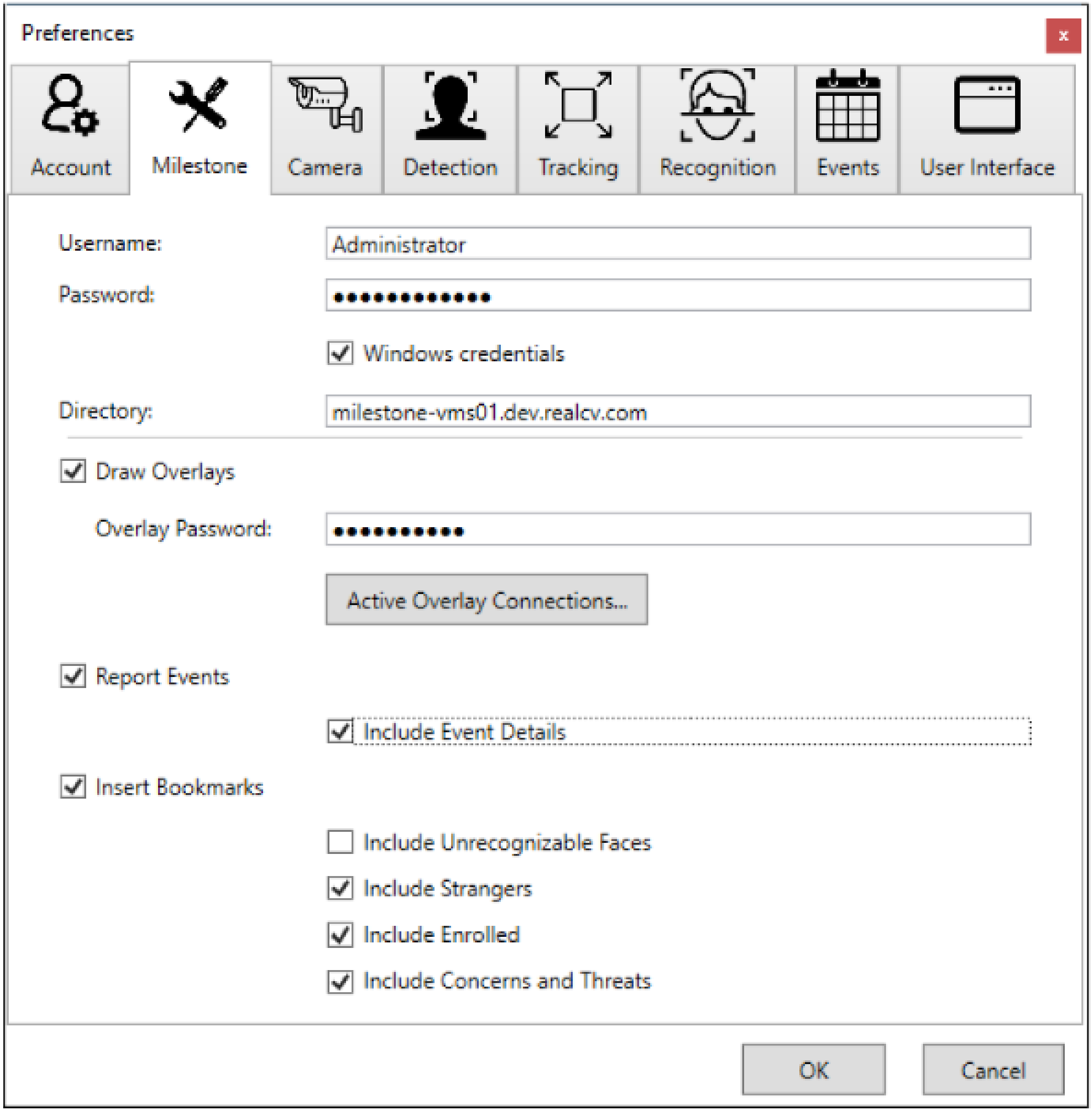
113.3.2 Verify your Connection

To verify successful connection to the Milestone system, open the **Preferences > Camera** tab. Cameras connected to the Milestone system should be visible. All cameras connected to the Milestone system have a

Milestone prefix in their names.

114 SAFR Milestone Operation Guide

114.1 SAFR Milestone Preferences



Preferences

Username: Administrator

Password: •••••••••••••

☑ Windows credentials

Directory: milestone-vms01.dev.realcv.com

☑ Draw Overlays

Overlay Password: •••••••••••

☑ Report Events

☑ Include Event Details

☑ Insert Bookmarks

☐ Include Unrecognizable Faces

☑ Include Strangers

☑ Include Enrolled

☑ Include Concerns and Threats

- **Username**: Name of the user with the necessary permissions to connect SAFR to Milestone.
- **Password**: Password of the user with the necessary permissions to connect SAFR to Milestone.
- **Windows credentials**: Indicates whether the user is a Windows user or a basic user.
- **Directory**: IP address or hostname of a Milestone server.
- **Draw Overlays**: Enables the use of SAFR overlays on Milestone cameras.
  - **Overlay Password**: This is the password that should be configured/used in Milestone to access SAFR to receive overlay data. See the Interpret Video Feed Overlays page for information about SAFR overlays. After you add the *overlay password*, you need to restart the Milestone XProtect Smart Client and the SAFR Desktop Client.
- **Report Events**: Enable to allow the reporting of SAFR events to Milestone. Note that only events reported to the SAFR Events server are reported to Milestone.
  - **Include Event Details**: Increases the verbosity of events in attaching JSON to the event that

includes all of the technical details of the event. This is useful if an operator is using macros to handle the event for decision-making.

- **Insert Bookmarks**: Adds bookmarks to the video stream related to events. Allows operators to search video for events or recognized person names. Use caution when deciding what to include since many faces can cause many bookmarks to be created.
  - **Include Unrecognizable Faces**: When enabled, adds bookmarks when a face is detected but SAFR does not have enough information to determine if they are a stranger or a known person. This can result in an overwhelming number of bookmarks, so it's disabled by default. However, this setting can be useful when monitoring areas with very few people.
  - **Include Strangers**: When enabled, adds bookmarks when a face is recognized and determined to be a stranger. This option is generally useful for secured areas where only known people should be.
  - **Include Enrolled**: When enabled, adds bookmarks when a face is recognized and determined to be a known person.
  - **Include Concerns and Threats**: When enabled, adds bookmarks when a face is recognized and determined to be a known concern or threat.

114.2 Connect Cameras to Milestone

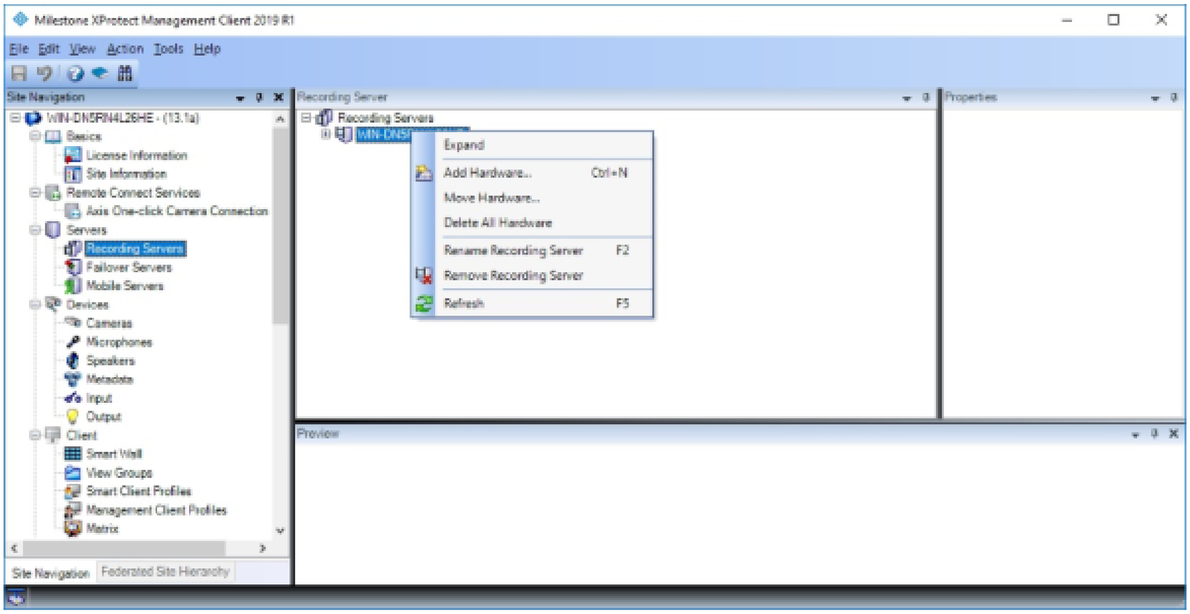
It is very likely that cameras are already connected to the Milestone server but if additional cameras are to be added use the Milestone Management client. The best information on how to do this is in the *Milestone XProtect Administration Guide*. However, what follows is a brief explanation of the procedure:

1. Open the Administration client.
2. Go to Recording Servers.
3. Right-click the Recording Server to which you want to add a camera.
4. Select **Add Hardware**.
5. Select a method to auto-discover the camera. For this example, use auto-discover.
   - Enter the username and password for the camera.
6. Check the box next to each manufacturer of the cameras you want to add. Click **Next**.
7. Add any username and password combination used to connect to the camera.
8. Cameras appear in the list as they are discovered. Check the box next to each camera you want to add.
9. The hardware for each camera is listed. Check the box for each part to be added. Types of parts are camera feeds, microphones, speakers, and others.
10. Hardware parts need to be added to groups. At the left, select the default group for each type of hardware. At the right, you can override the group for individual hardware parts.
11. Click **Finish** to add the selected hardware.

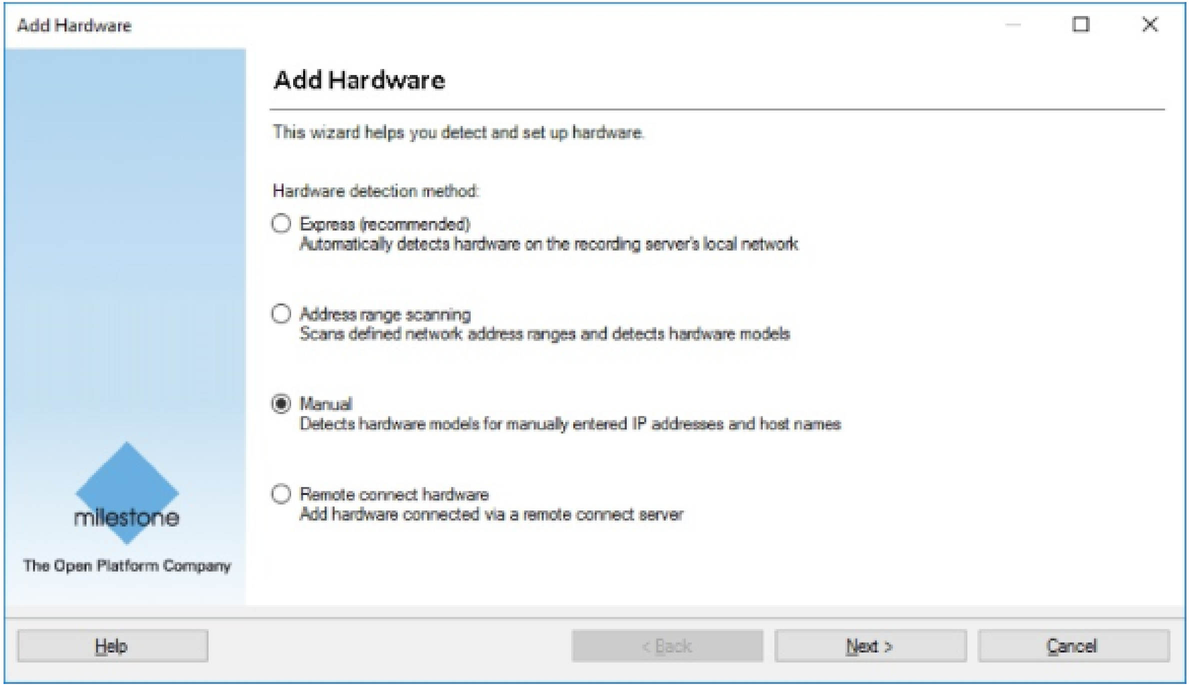
Note: The *XProtect Smart Client* has to be restarted to pick up newly added cameras.

114.3 Create Overlay Metadata Driver on Milestone Server for each Camera

1. Open the Milestone XProtect Management Client. In the left pane, under Servers, click **Recording Servers**.
2. In the center pane, right-click the appropriate server name, and click **Add Hardware**.



3. In the dialog, click the **Manual** option, and click **Next**.



Add Hardware

Add Hardware

This wizard helps you detect and set up hardware.

Hardware detection method:

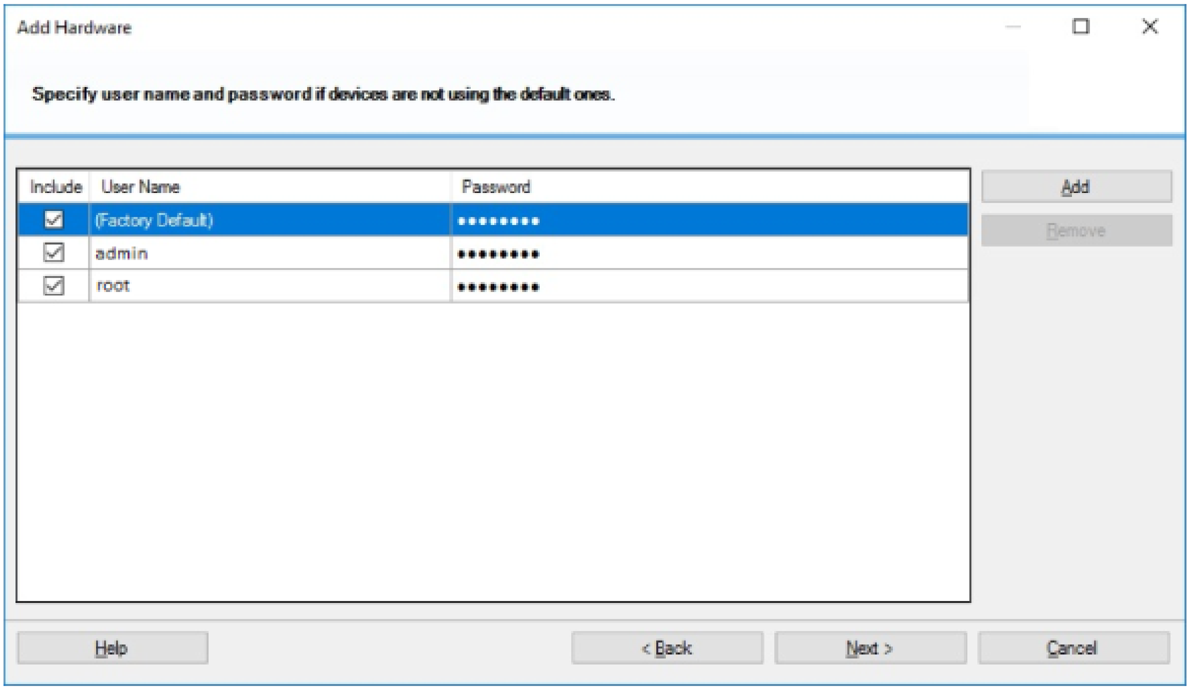
○ Express (recommended)
Automatically detects hardware on the recording server's local network

○ Address range scanning
Scans defined network address ranges and detects hardware models

● Manual
Detects hardware models for manually entered IP addresses and host names

○ Remote connect hardware
Add hardware connected via a remote connect server

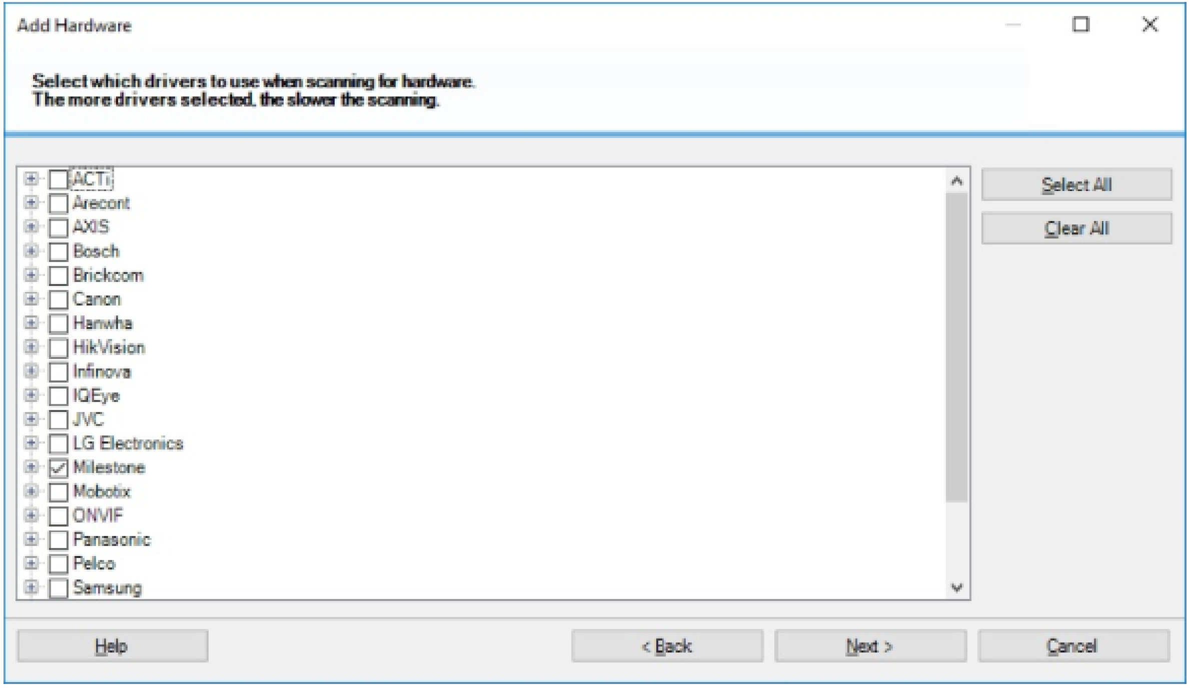
4. If not already in the list, click **Add** to include the password from the SAFR > Preferences > Milestone tab **Overlay Password** field. Include a username of your choice. (e.g. *SAFR*)
   - If you make changes to the overlay password on the **SAFR > Milestone** tab, make sure that you click **OK** at the bottom of the window. Then make sure the password on the Add Hardware username/password page includes the changed password. If not, the add hardware operation will fail.
5. After you have edited the username and password table, click **Next**.



Specify user name and password if devices are not using the default ones.

Include	User Name	Password
☑	(Factory Default)	••••••••
☑	admin	••••••••
☑	root	••••••••

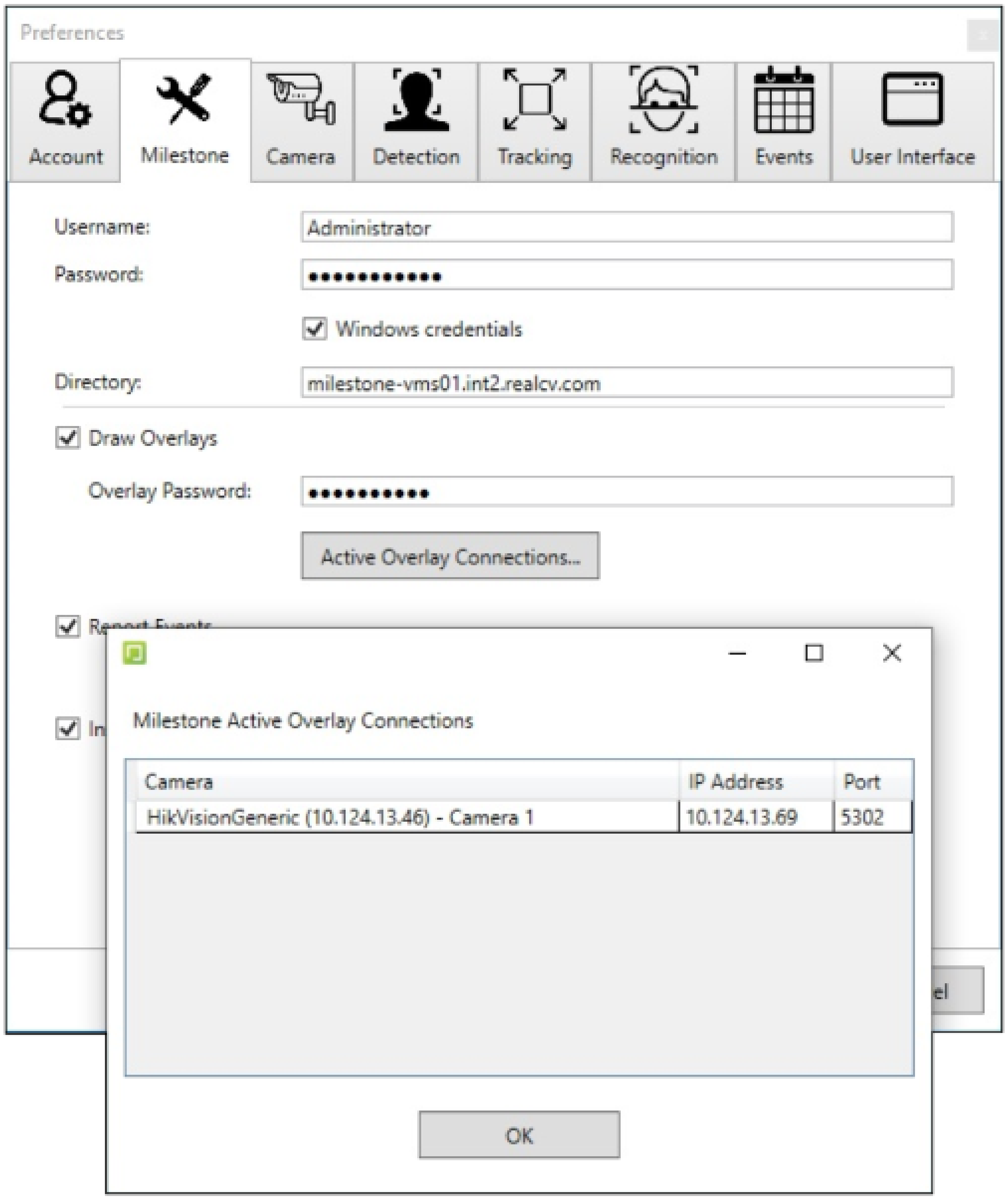
6. Click **Clear All**, select Milestone only, and click **Next**.



Select which drivers to use when scanning for hardware.
The more drivers selected, the slower the scanning.

- ☐ ACTi
- ☐ Arecont
- ☐ AXIS
- ☐ Bosch
- ☐ Brickcom
- ☐ Canon
- ☐ Hanwha
- ☐ HikVision
- ☐ Infinova
- ☐ IQEye
- ☐ JVC
- ☐ LG Electronics
- ☑ Milestone
- ☐ Mobotix
- ☐ ONVIF
- ☐ Panasonic
- ☐ Pelco
- ☐ Samsung

7. Add the IP address and port number of the SAFR client active overlay connections. To find this information, go to SAFR > Properties > Milestone tab, and click **Active Overlay Connections**. The dialog provides a table of camera names, IP addresses, and port numbers. Please note the IP address of the SAFR Desktop Client and not the IP address of the camera itself.
   - The password in the *Overlay Password* field must be included in the Add Hardware username/-password table or the add hardware operation will fail.



Username: Administrator

Password: ●●●●●●●●●●●●

☑ Windows credentials

Directory: milestone-vms01.int2.realcv.com

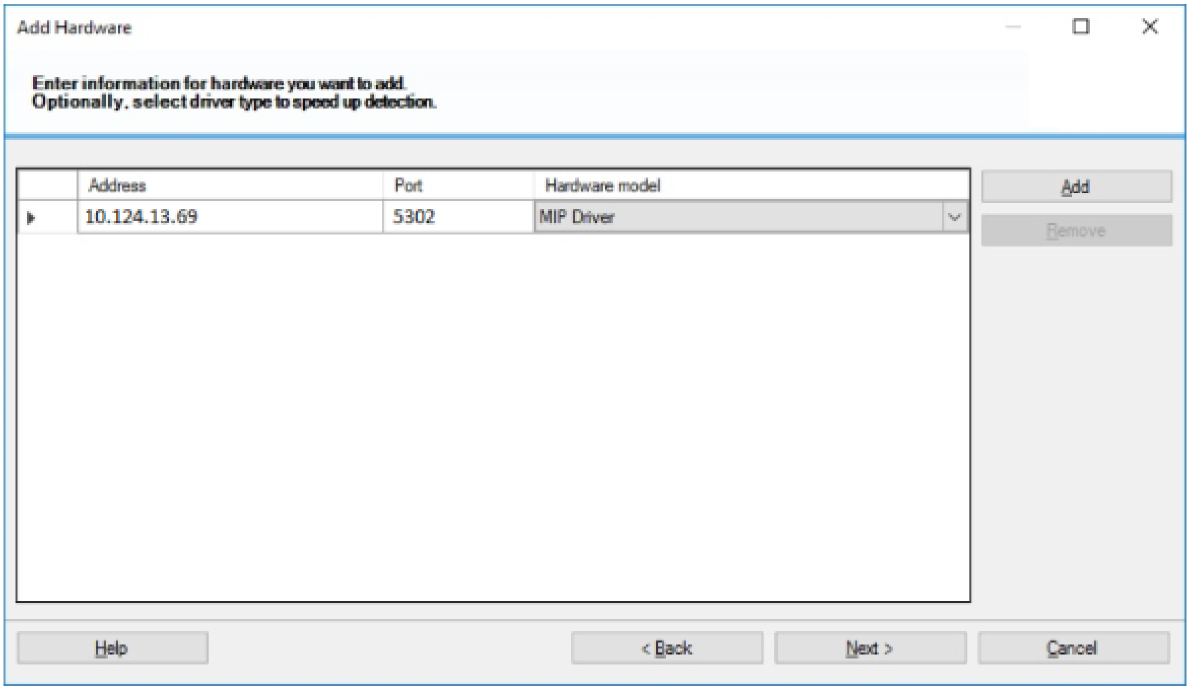
☑ Draw Overlays

Overlay Password: ●●●●●●●●●●●

Milestone Active Overlay Connections

Camera	IP Address	Port
HikVisionGeneric (10.124.13.46) - Camera 1	10.124.13.69	5302

8. From the **Hardware Model** menu, select **MIP Driver**. Click **Add** to create more rows for additional MIP driver addresses and ports if needed. Click **Next**.

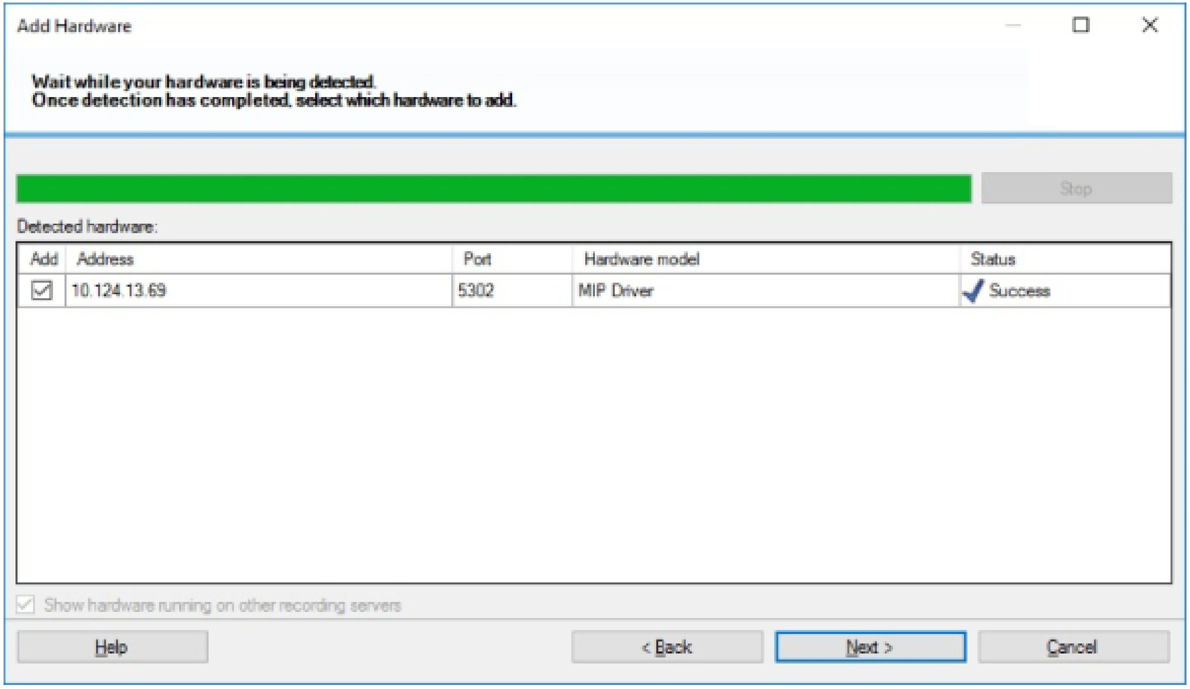


Add Hardware

Enter information for hardware you want to add.
Optionally, select driver type to speed up detection.

	Address	Port	Hardware model
▸	10.124.13.69	5302	MIP Driver

9. On connection success, click **Next**.



Add Hardware

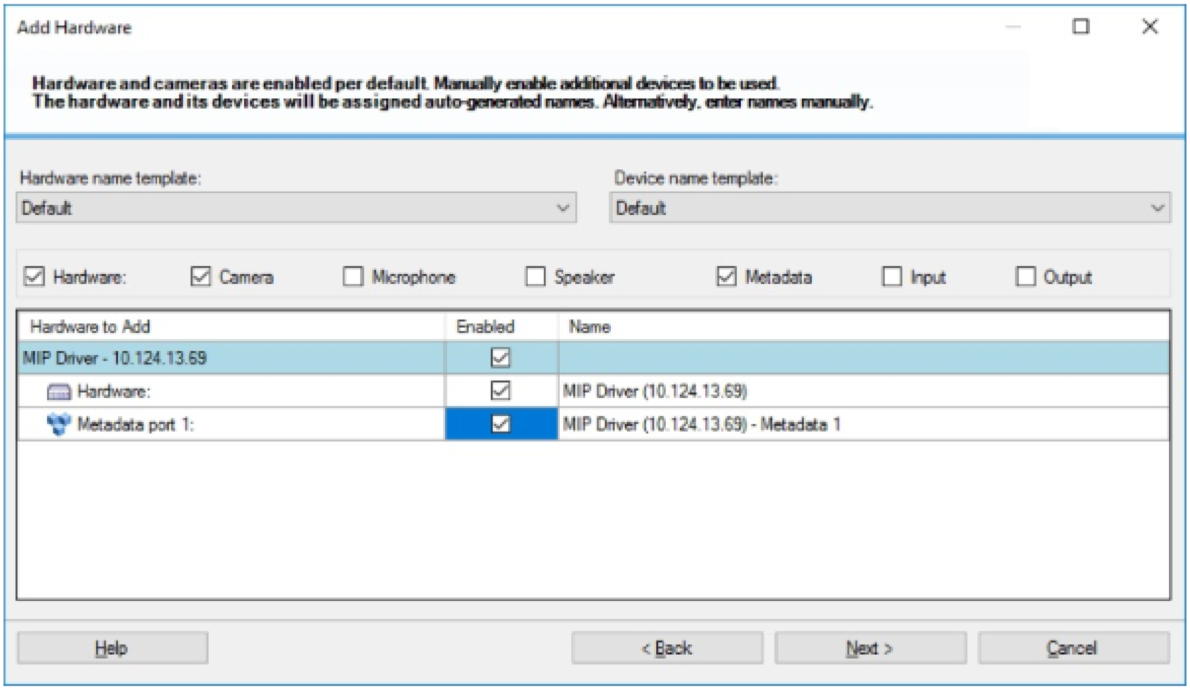
Wait while your hardware is being detected.
Once detection has completed, select which hardware to add.

Detected hardware:

Add	Address	Port	Hardware model	Status
☑	10.124.13.69	5302	MIP Driver	✓ Success

☑ Show hardware running on other recording servers

10. Select the **Metadata** check box. We recommended that you click the **Name** field for the MIP driver (Metadata Port 1) and rename it to include the name of the camera to assist you when associating the camera with the MIP driver later. Click **Next** to add the hardware.



Add Hardware

Hardware and cameras are enabled per default. Manually enable additional devices to be used.
The hardware and its devices will be assigned auto-generated names. Alternatively, enter names manually.

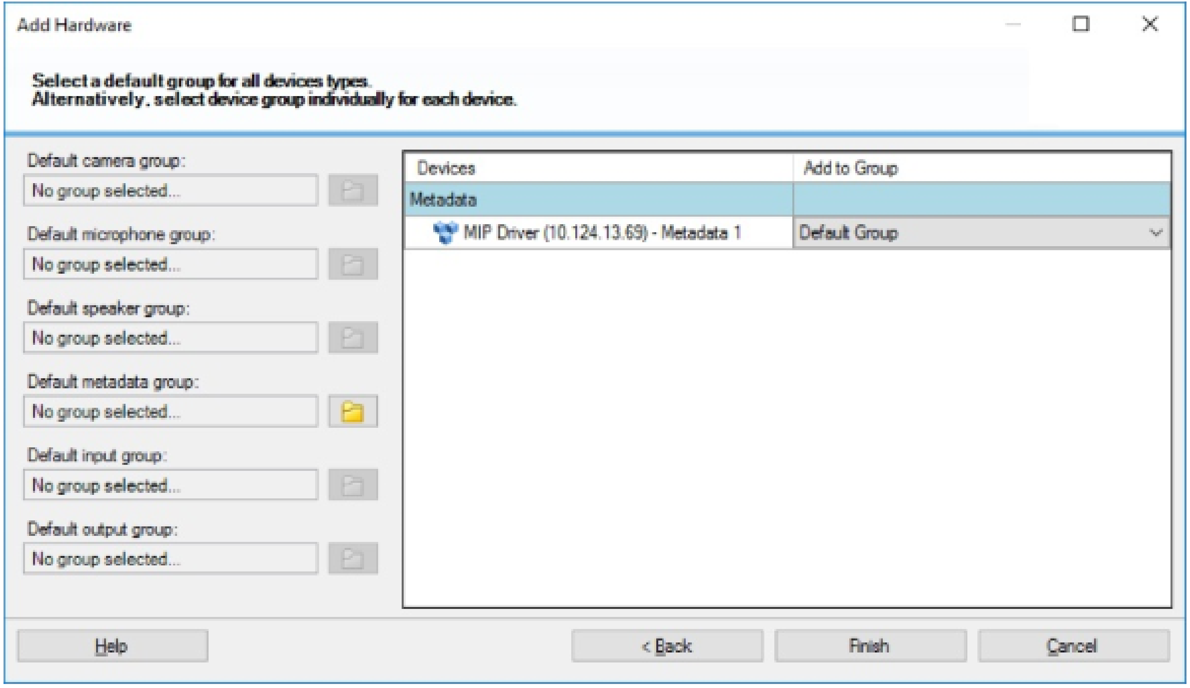
Hardware name template: Default

Device name template: Default

☑ Hardware: ☑ Camera □ Microphone □ Speaker ☑ Metadata □ Input □ Output

Hardware to Add	Enabled	Name
MIP Driver - 10.124.13.69	☑	
Hardware:	☑	MIP Driver (10.124.13.69)
Metadata port 1:	☑	MIP Driver (10.124.13.69) - Metadata 1

11. From the **Add to Group** menu, associate the MIP driver with a specific metadata group to make it easier to locate. Click **Finish**.



Add Hardware

Select a default group for all devices types.
Alternatively, select device group individually for each device.

Default camera group: No group selected...

Default microphone group: No group selected...

Default speaker group: No group selected...

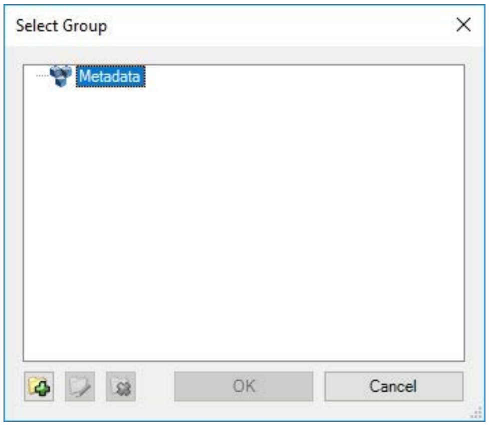
Default metadata group: No group selected...

Default input group: No group selected...

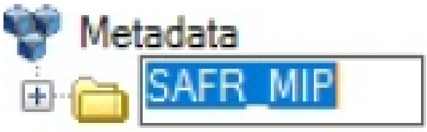
Default output group: No group selected...

Devices	Add to Group
Metadata	
MIP Driver (10.124.13.69) - Metadata 1	Default Group

12. If the menu does not include a choice, or if you receive an *Assign Device to a Device Group* message, do the following:
    1. In the list in the left pane, to the right of the Default Metadata Group field, click the folder icon.
    2. In the Select Group dialog, in the lower-left corner, click the folder-plus icon to add a group.



3. Provide a name for the device group, and click **OK**.

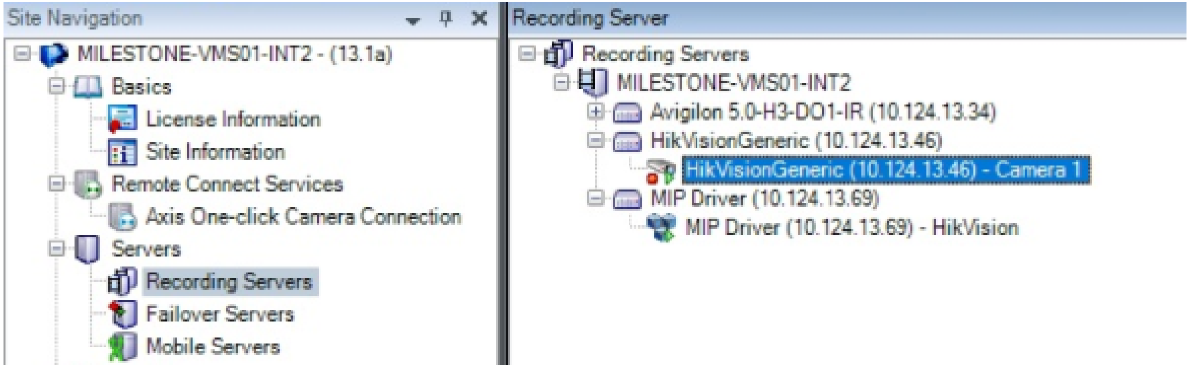


4. From the **Add to Group** menu, you can now select the new device group.

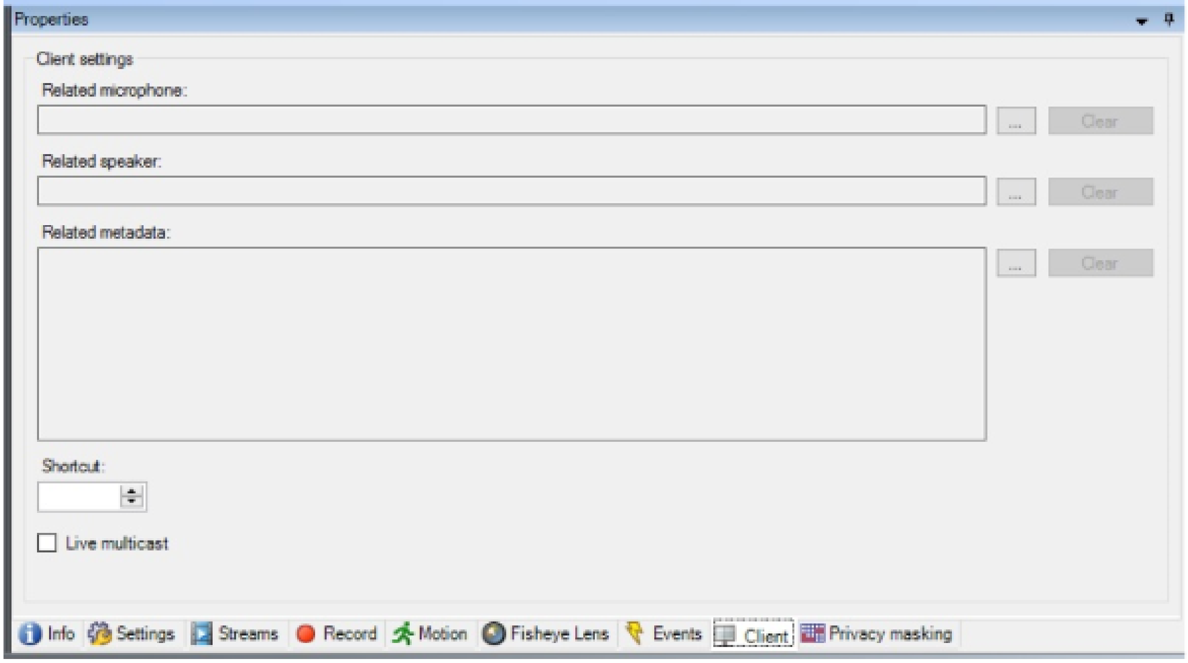
13. After the driver has an associated device group, click **Finish**.

114.4 Associate the Milestone Integration Platform (MIP) Device with the Camera

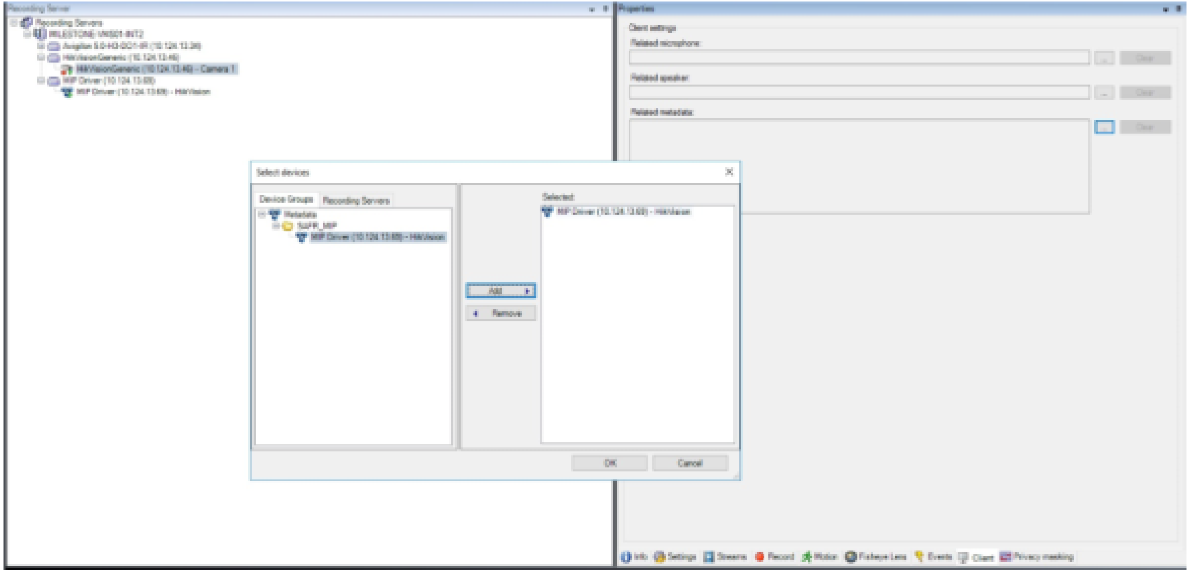
1. In the Milestone XProtect Management Client, in the left pane, click **Recording Servers** and expand the tree view to display the cameras and MIP drivers.


Site Navigation
MILESTONE-VMS01-INT2 - (13.1a)
Basics
License Information
Site Information
Remote Connect Services
Axis One-click Camera Connection
Servers
Recording Servers
Failover Servers
Mobile Servers
Recording Server
Recording Servers
MILESTONE-VMS01-INT2
Avigilon 5.0-H3-DO1-IR (10.124.13.34)
HikVisionGeneric (10.124.13.46)
HikVisionGeneric (10.124.13.46) - Camera 1
MIP Driver (10.124.13.69)
MIP Driver (10.124.13.69) - HikVision

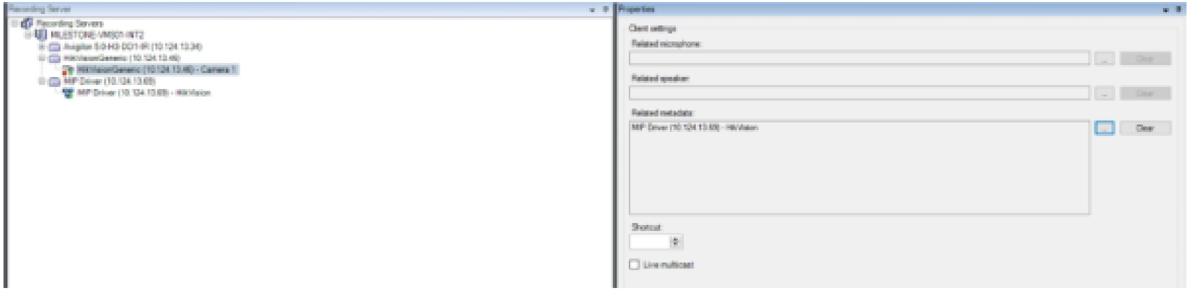
2. With the camera highlighted, in the right-pane, at the bottom of the window, click **Client**.



3. Under Client Settings > Related Metadata, click the ... button.



4. In the Select Devices dialog, on the Device Groups tab, click to highlight the MIP device you renamed earlier for easy identification. Click Add so that the device appears in the Selected pane. Click OK.


Recording Server
Recording Servers
MILESTONE-VMS01-INT2
Avigilon 5.0-H3-DO1-IR (10.124.13.34)
HikVisionGeneric (10.124.13.46)
HikVisionGeneric (10.124.13.46) - Camera 1
MIP Driver (10.124.13.69)
MIP Driver (10.124.13.69) - HikVision
Properties
Client settings
Related microphone
Related speaker
Related metadata
MIP Driver (10.124.13.69) - HikVision
Clear
Shortcut
Live multicast

5. Click **Save** to save the changes.

In the right Related Metadata field, the MIP device name is displayed and is now associated with the highlighted camera in the left pane.

114.5 Connect SAFR to Milestone Video Feeds

Once the camera is in the Video Archiver, it shows up as a Milestone camera in SAFR. If it does not, try closing and re-opening SAFR.

To get the SAFR enhancements to show up on the Milestone camera feed:

1. Open the SAFR Desktop Client.
2. Select the Milestone version of the camera (It has Milestone as the first part of the camera name.) from the menu in the main window (upper left).

On successful connection to the Milestone camera, video from the camera plays in the SAFR Desktop Client window.

If overlays have been configured, you can see the enhancements by watching the camera's video feed in XProtect:

1. Open XProtect.
2. Go to the Live tab.
3. Drag and drop a camera from the left side into one of the tiles in the middle.

The camera feed should start up and show the same overlays that are in SAFR (for example, ovals, names).

To connect to additional cameras:

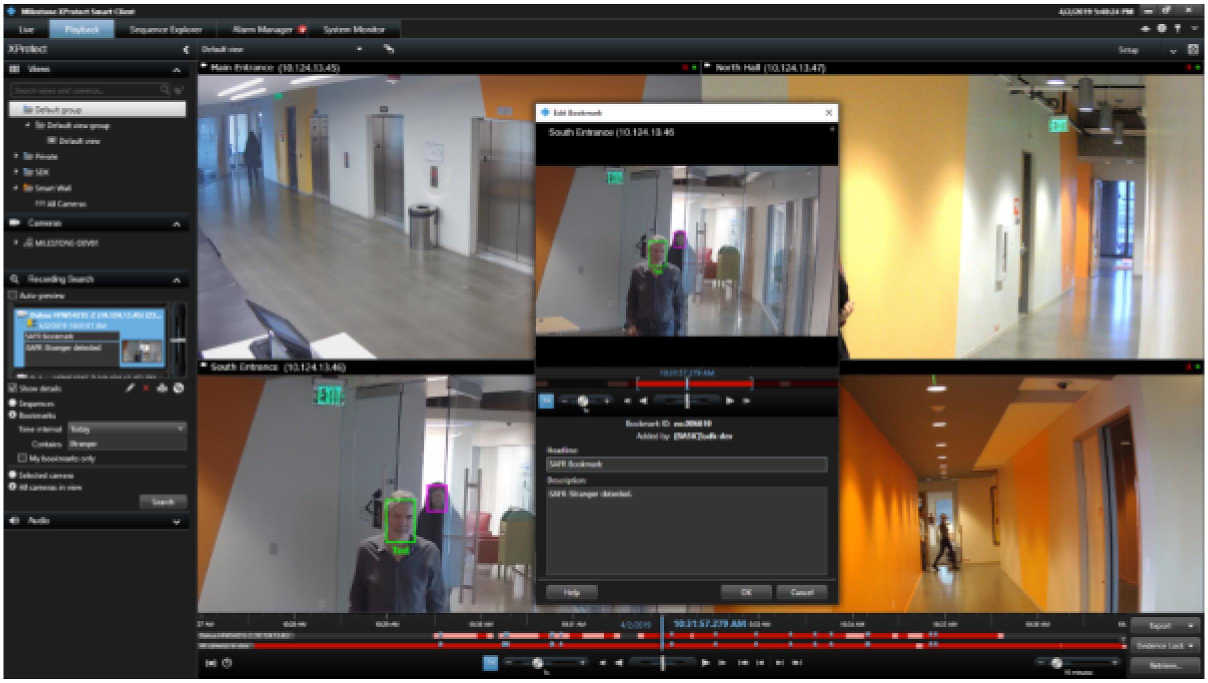
1. Open another instance of the SAFR Desktop Client by selecting **New** from the **File** menu.
2. Repeat the previous steps to connect to a different Milestone camera feed.
   On successful connection to the Milestone camera, video from the camera plays in the SAFR Desktop Client window.
3. Repeat this process for as many video feeds as desired (up to the capacity of the machine SAFR is installed on).
4. If more capacity is needed, install SAFR on additional machines and repeat the setup process.

Note: By default, the SAFR Desktop Client for Milestone operates in *Enrolled Monitoring* video processing mode and generates events and bookmarks into the Milestone system for every enrolled person. If a different mode is desired for a selected camera, choose a different mode from the video window mode selector menu.

114.6 Automatic Bookmarks

Milestone creates bookmarks to help locate important events. Bookmarks are populated with *person type*, *ID class*, and *name*. They can also provide more detailed information with even more search attributes, such as age and gender.

The following illustration shows how bookmarks can be used to review important events, such as the detection of a stranger tailgating behind a registered user.


Milestone XProtect Smart Client
Live
Playback
Sequence Explorer
Alarm Manager
System Monitor
Main Entrance (10.124.13.45)
North Hall (10.124.13.47)
South Entrance (10.124.13.46)
Edit Bookmark
South Entrance (10.124.13.46
Help
OK
Cancel
10:31:57.279 AM

Headline: SAFR Bookmark

Description: SAFR Stranger detected.

Note: The purple indicator identifies the person is a stranger.

114.7 SAFR Identities

To add people through the SAFR Desktop Client from an image or video file, do the following:

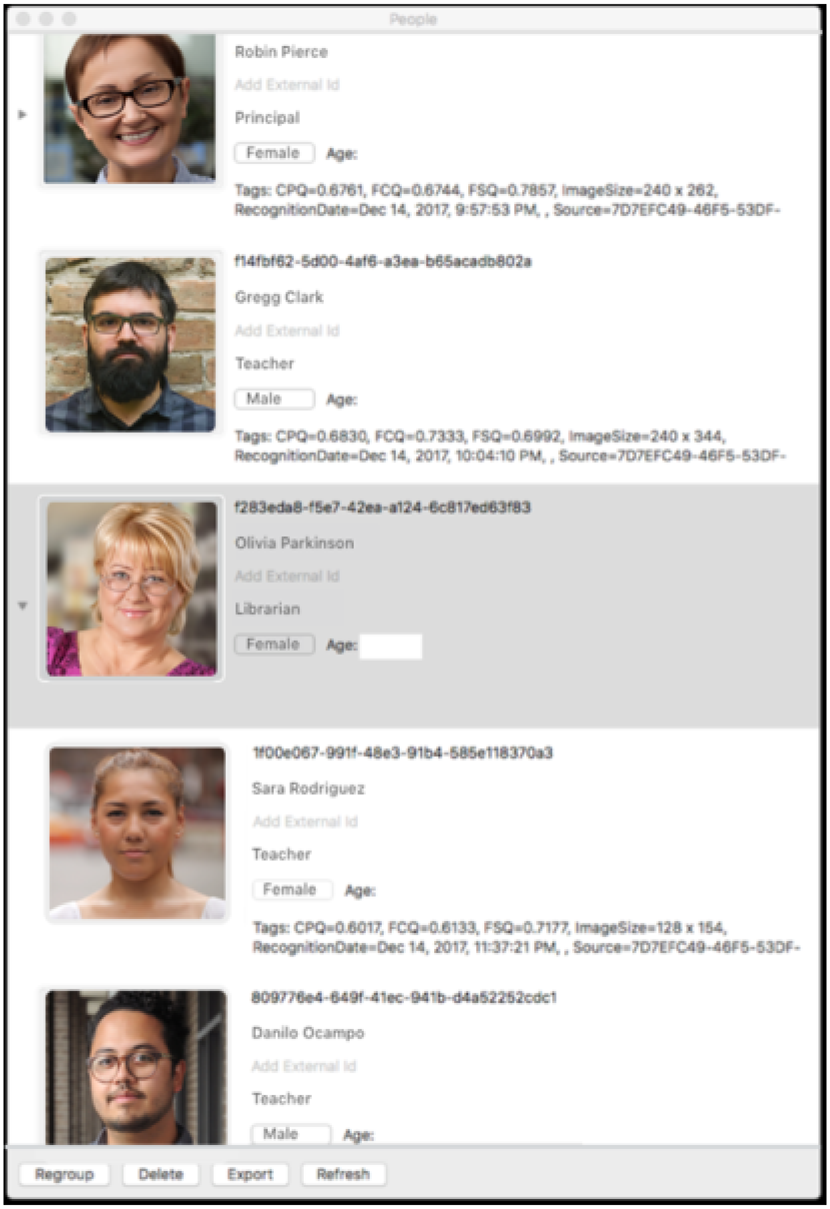
1. Open the Desktop Client.
2. Click **File > Import Faces**.
3. Select the image.
   - For an image, each recognized face is enclosed by a box, and you have the option to type a name.
   - For a video, each recognized person is learned automatically as long as the faces meet the minimum criteria for recognition.
4. If faces are not learned, check the settings in the Detection and Recognition tabs under Preferences to ensure faces meet minimum criteria.
   - Detection > Minimum searched face size
   - Recognition > To allow identification

Users added to SAFR are not synchronized to Milestone. These users exist only in SAFR.

It may be desirable to edit people properties to control which events get triggered when that person is recognized. For example, setting a person's *ID Class* to *Concern* or *Threat* triggers the respective alarms. The most important people attributes are *Name*, *Image*, *Person Type*, and *ID Class*.

The *Name*, *Image*, and *Person Type* should be edited through SAFR. *Person Type* defines a person's role (e.g. staff or visitor), while the *ID Class* defines the risk level (No-Concern, Concern, or Threat). *Person Type* and *Image* can be edited in the Desktop Client by changing the *Person Type* on the People screen.

ID Class and all other attributes of a person are also edited within SAFR People dialog, accessed through the SAFR Desktop Client **Tools** menu. All identities are created by default with an *ID Class* of *No Concern*. To edit a person's *ID Class*, open the People window from the SAFR Desktop Client **Tools** menu as follows:



Robin Pierce

Add External Id

Principal

Female Age:

Tags: CPQ=0.6761, FCQ=0.6744, FSQ=0.7857, ImageSize=240 x 262, RecognitionDate=Dec 14, 2017, 9:57:53 PM, , Source=7D7EFC49-46F5-53DF-

f14fbf62-5d00-4af6-a3ea-b65acadb802a

Gregg Clark

Add External Id

Teacher

Male Age:

Tags: CPQ=0.6830, FCQ=0.7333, FSQ=0.6992, ImageSize=240 x 344, RecognitionDate=Dec 14, 2017, 10:04:10 PM, , Source=7D7EFC49-46F5-53DF-

f283eda8-f5e7-42ea-a124-6c817ed63f83

Olivia Parkinson

Add External Id

Librarian

Female Age:

1f00e067-991f-48e3-91b4-585e118370a3

Sara Rodriguez

Add External Id

Teacher

Female Age:

Tags: CPQ=0.6017, FCQ=0.6133, FSQ=0.7177, ImageSize=128 x 154, RecognitionDate=Dec 14, 2017, 11:37:21 PM, , Source=7D7EFC49-46F5-53DF-

809776e4-649f-41ec-941b-d4a52252cdc1

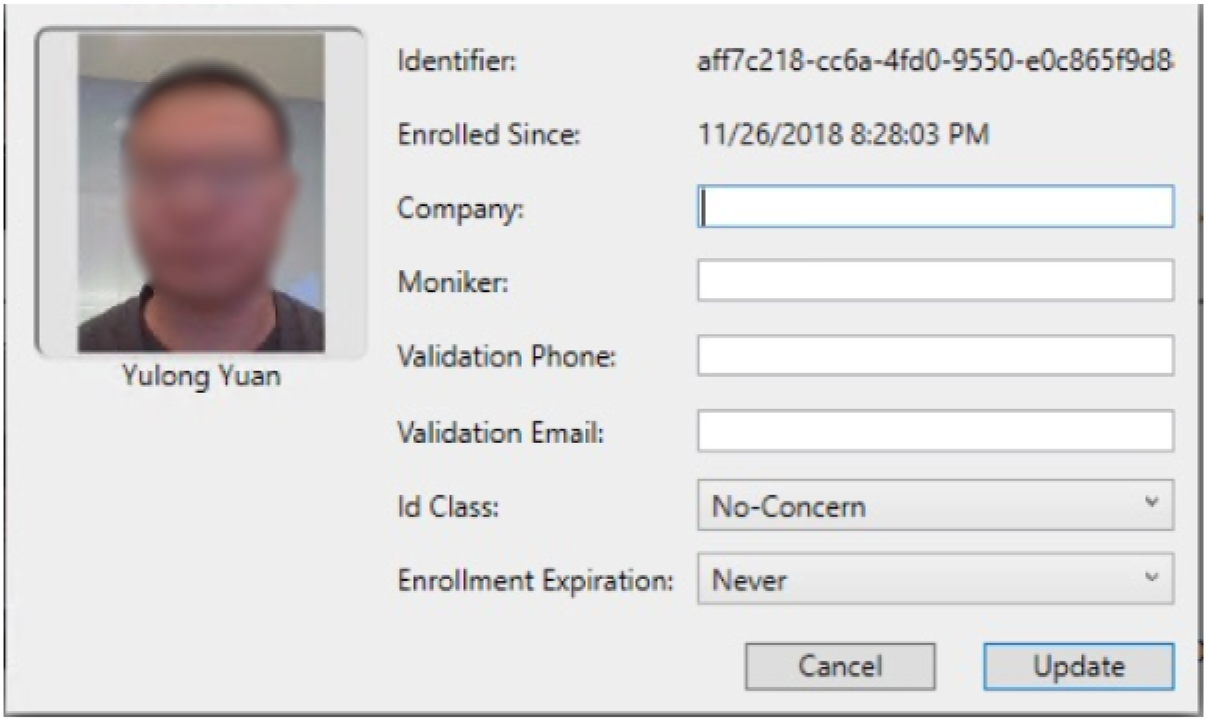
Danilo Ocampo

Add External Id

Teacher

Male Age:

The *Person Type* and *Name* can be edited by clicking the respective fields on the People screen. To edit *ID Class*, double-click the person and choose the desired value from the ID Class menu in the People Edit dialog as shown in the following illustration:



Yulong Yuan

Identifier: aff7c218-cc6a-4fd0-9550-e0c865f9d8

Enrolled Since: 11/26/2018 8:28:03 PM

Company:

Moniker:

Validation Phone:

Validation Email:

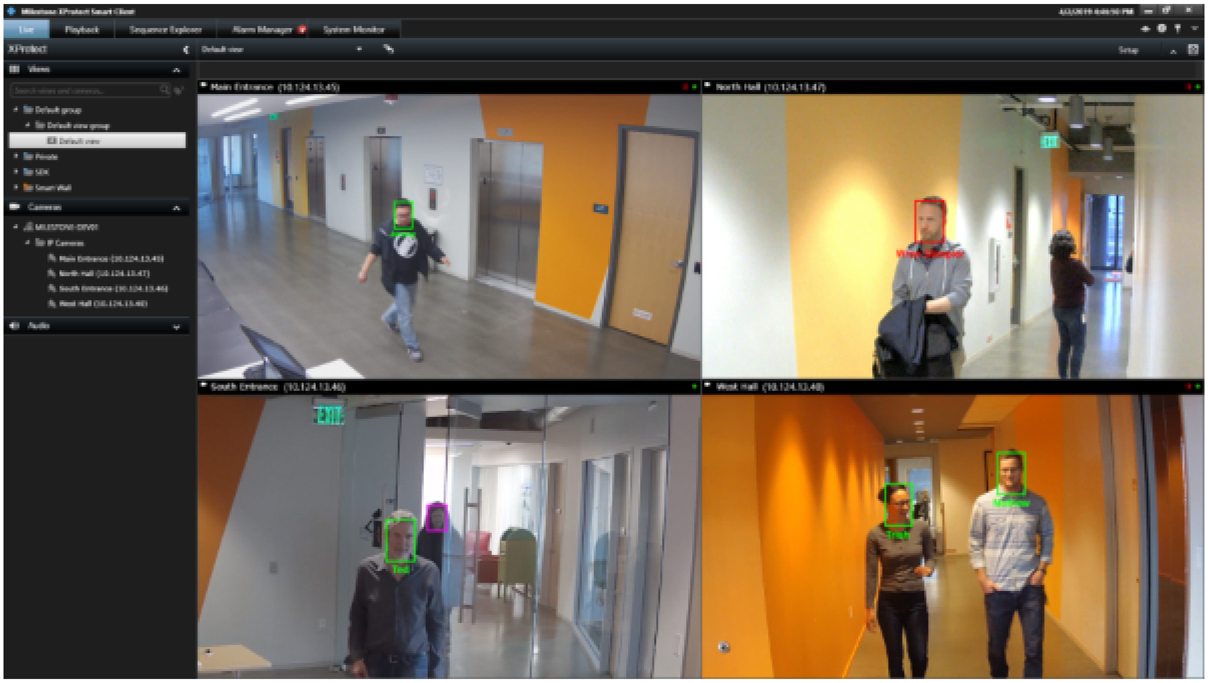
Id Class: No-Concern

Enrollment Expiration: Never

114.8 SAFR Overlays

SAFR for Milestone enhances video monitoring by providing overlays that gives the security personnel more information, including person types, threat classification, and name. It can even be used to augment the video views with age and gender information that may be useful in reporting suspects.

The following illustration shows how overlays give more information about the subjects in view:



The person in the top left is a stranger, the person in the top right is a recognized person with low probability, and the person in the bottom left is a known threat. The information is conveyed by the color of their overlays. The following list describes the default colors used in SAFR overlays:

- **Gray**: Unrecognizable. A face was detected but it wasn't of sufficient quality to attempt recognition.
- **Purple**: Stranger. The person has been recognized, but they're not in Genetec's cardholder database nor in SAFR's Person Directory.
- **Blue**: Registered person without a name. The face was recognized as matching one already in the database.
- **Green**: Registered person with a name.
- **Yellow**: Concern. The registered face has been tagged as a concern.
- **Red**: Threat. The registered face has been tagged as a threat.

114.9 SAFR Video Processing Modes

SAFR has different video processing modes to control what events are generated. The following is a short summary of the modes most relevant to Milestone XProtect integration. For a complete description, see Connect to a Video Feed in the *SAFR Documentation*.

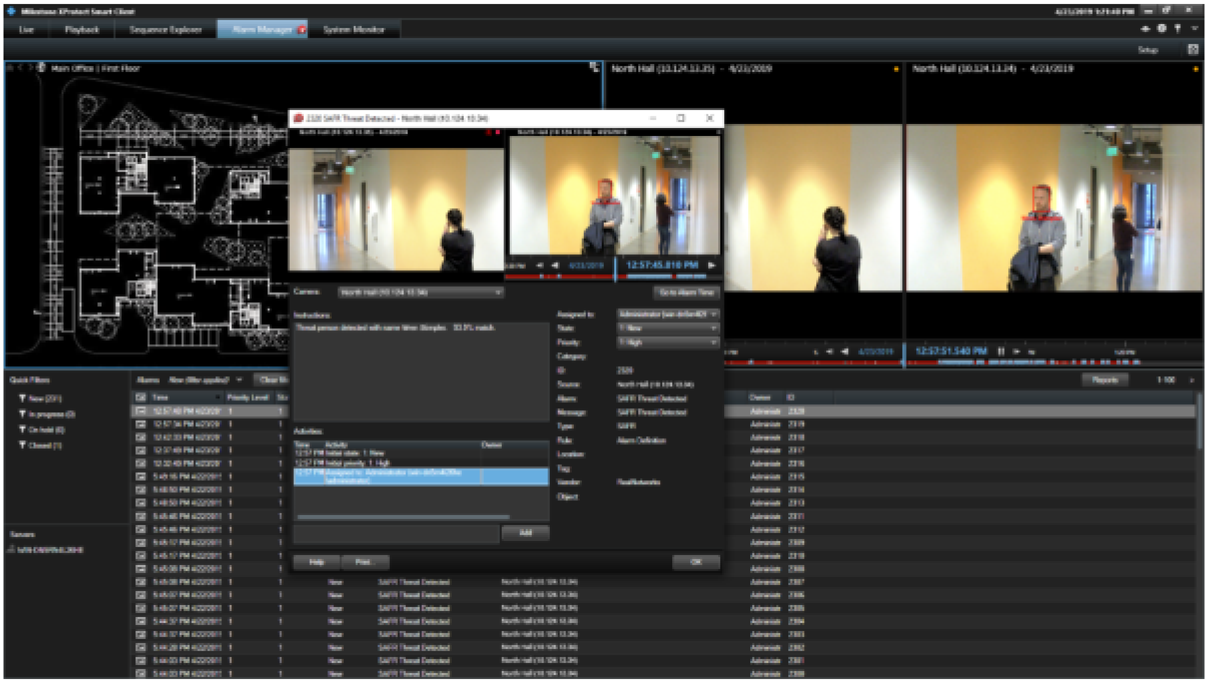
- **Secure Access**: Trigger events only for cardholders (or persons registered directly in SAFR) only when there is a high confidence match. Useful for secure entry (for example, unlocking doors).
- **Smile to Unlock**: Same as *Secure Access* but includes an additional trigger event if the user smiles.
- **Enrolled Monitoring**: Trigger events for cardholders (or persons registered directly in SAFR) but with lower confidence.
- **Enrolled and Stranger Monitoring**: Same as *Enrolled Monitoring* but triggers events for strangers also.

114.10 Alarms and Notifications

You can also use SAFR to view recognition events. Recognition events occur when a known, unknown, or unrecognized person appears in the view of a camera. The types of recognized persons are:

- Unrecognizable: A face was detected but it wasn't of sufficient quality to attempt recognition.
- Stranger.
- Registered person without a name.
- Registered person with a name.
- Registered person marked as a *Concern*.
- Registered person marked as a *Threat*.

There are several combinations of these conditions that can be triggered. The following shows multiple events populated in the Milestone alerts panel:



Note: Clicking any of the events on this screen allows the video from that event to be replayed.

114.11 Troubleshooting Tips

Note: When closing SAFR, use the **Quit SAFR** option on the **File** menu. Closing SAFR using the Window Close button will cause you to lose the SAFR state settings and connected cameras for that window.

- If detection or recognition results in not many faces found or recognized, check that the Milestone video feeds are of a sufficiently large frame size.
- If events are not being triggered, ensure the correct SAFR video processing mode is selected.
- If overlays are not displayed after configuration, the issue may involve a Windows Defender firewall security alert either before or after your system has been rebooted. The result is that the SAFR application is blocked. Configure Windows Defender Firewall Inbound Rules to enable SAFR and Client.

115 SAFR Panasonic Video Insight Integration Guide

Integrated SAFR Video Insight is only available on Windows.

Integrating SAFR's facial recognition and analysis capabilities into Video Insight allows SAFR to trigger camera flashes and/or pop-up alert messages on Video Insight camera windows when specified types of people are seen by cameras. This makes it much easier to quickly and accurately separate unknown people from authorized people from known threats.

115.1 Integration Overview

A typical deployment requires the following:

- A machine running VI Health MonitorPlus.
- One or more machines running the SAFR Desktop Client to process videos.
- If you're doing a SAFR on-premise deployment, you'll also need a machine running SAFR Server. SAFR Server can run on the same machine as one of the Desktop Clients, provided the host machine meets the system requirements.

Cameras are connected to the VI Health MonitorPlus. The SAFR Desktop Client can then connect to the MonitorPlus to perform analysis of the video feeds and trigger Video Insight camera view color flashes and/or pop-up alert messages. Depending on the number of cameras you need, one or more machines may be required to run the SAFR Desktop Client(s), each processing multiple video feeds. The Desktop Client is also used to perform various management activities.

115.1.1 System Requirements

Video Insight has the following system requirements:

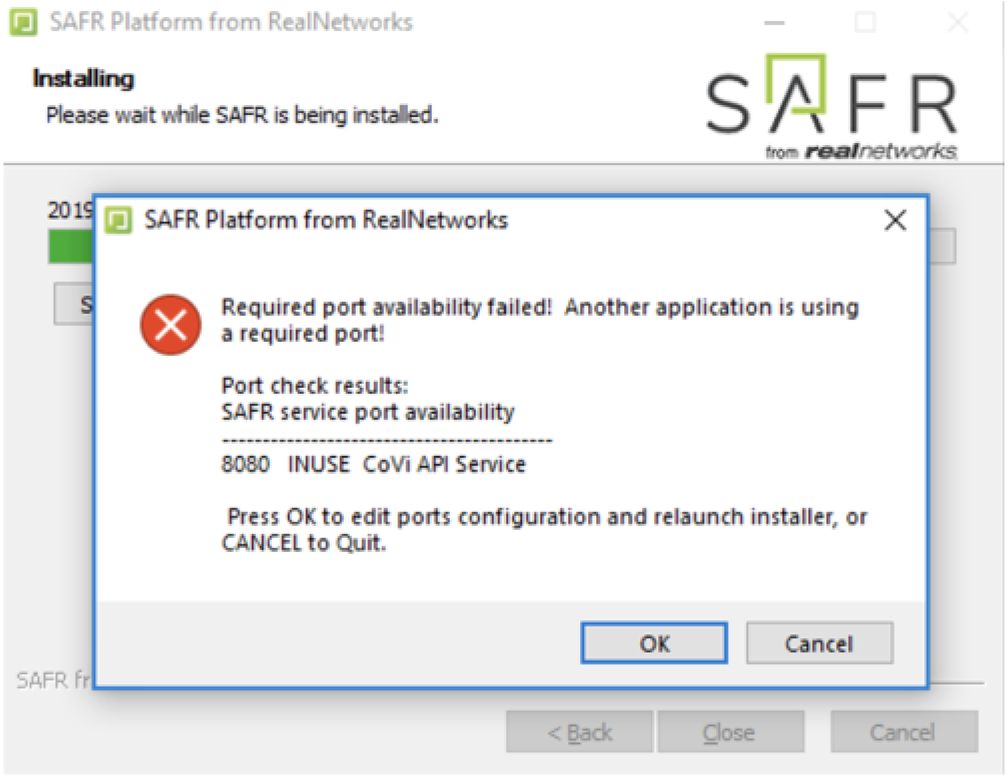
- A machine running VI Health MonitorPlus.
- Each machine running a Video Insight product must meet the following system requirements:
  - Dual 2.4 GHz Quad CPU
  - 8GB RAM
  - 1 GB/s network interface
  - Windows Server 2012 R2

SAFR has the following system requirements:

- Each machine running the SAFR Desktop Client must meet the following requirements:
  - Windows 10.
  - The Desktop Client must be version 3.2.141 or later.
  - Additional system requirements as described on the SAFR system requirements page.
- On-premise SAFR deployments require at least one machine running SAFR Platform 3.2.139 or later.
- Each machine running SAFR Server must meet the following requirements:
  - Windows 10.
  - Additional system requirements as described on the SAFR system requirements page.

115.2 Install and Configure SAFR

1. Go to the SAFR Download Portal.
2. If you're doing a cloud deployment, download and install Windows SAFR Desktop. Make sure to select the Video Insight install option.
3. If you're doing an on-premise deployment, download and install Windows SAFR Platform. Make sure to select the Video Insight install option.
   - When installing the SAFR Platform, the default SAFR port assignments sometimes conflict with other software port assignments. If a port conflict occurs, the error message shown below will pop up in the middle of your installation.


SAFR Platform from RealNetworks
Installing
Please wait while SAFR is being installed.
SAFR from realnetworks
SAFR Platform from RealNetworks
OK
Cancel
< Back
Close
Cancel

Required port availability failed! Another application is using a required port!

Port check results:
SAFR service port availability

```
-------------------------------------------
8080   INUSE  CoVi API Service
```

Press OK to edit ports configuration and relaunch installer, or CANCEL to Quit.

If this happens, do the following:

1. Click **OK** to edit port configurations.
2. **Notepad** will open, displaying the *safrports.conf* file.
3. Edit any conflicting ports to new values. (e.g. CoviHTTP=18080)
4. Save and exit **Notepad**.

The Platform installer will then restart and the new port values will be used. You can find the modified *safrports.conf* file at `C:\Program Files\RealNetworks\SAFR\`.

After the installation finishes, two icons will appear on your desktop: one labeled *SAFRActions* and another labeled *SAFR*. *SAFRActions* launches SAFR Actions, while *SAFR* launches the Desktop Client. The SAFR Server (when installed as part of a local deployment) automatically runs as a collection of background services.

Immediately following installation, the installer opens the Desktop Client and prompts you to log in with your SAFR Account credentials. Make sure to log in; it's important in acquiring the SAFR license.

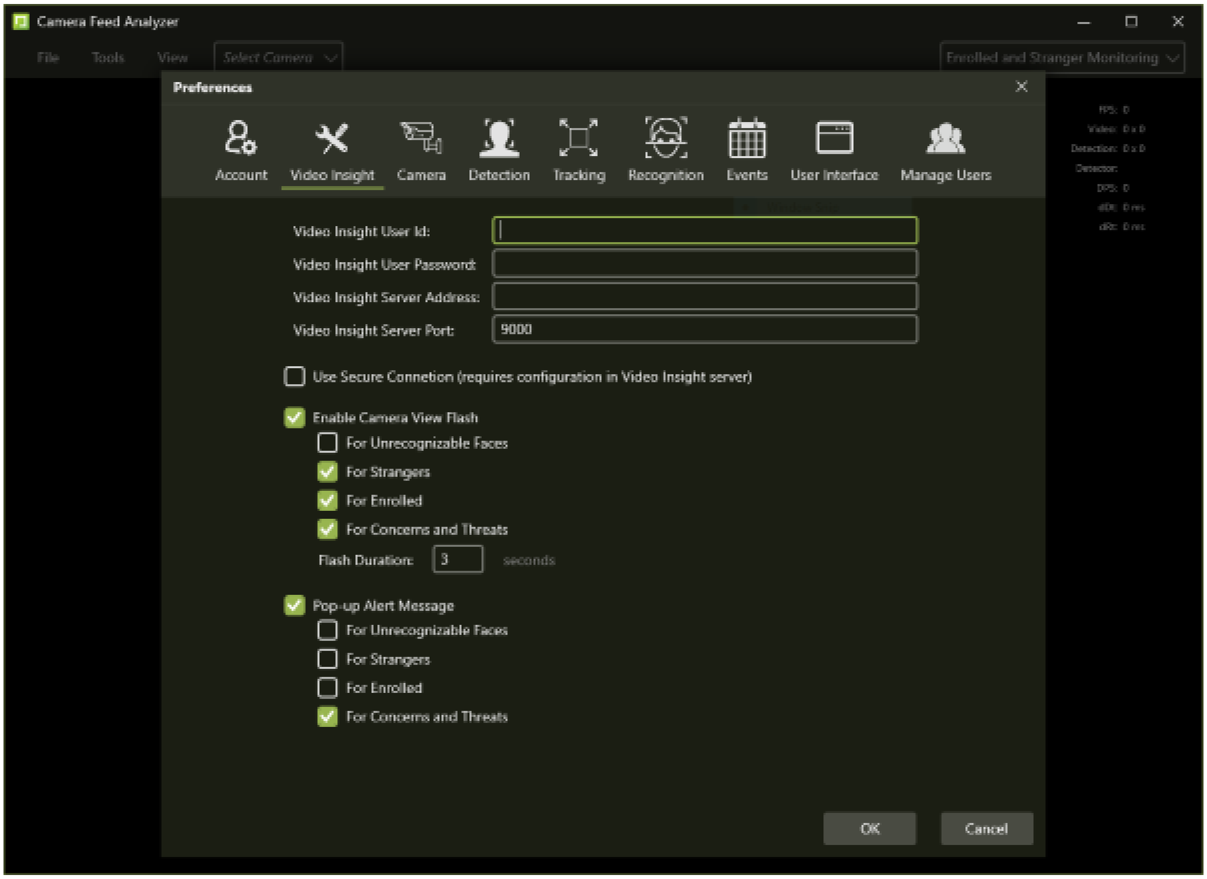
115.2.1 Connect SAFR to Video Insight

To connect SAFR to Video Insight, do the following:

1. Within your SAFR Desktop Client, select **Tools->Preferences->Video Insight**.

2. Enter information for the first four settings on the SAFR Video Insight Preferences menu: *Video Insight User Id*, *Video Insight User Password*, *Video Insight Server Address*, and *Video Insight Server Port*. See below for information about the available SAFR Video Insight preference settings.

115.2.1.1 SAFR Video Insight Preferences You can configure several settings specific to Video Insight by opening the SAFR Desktop Client and clicking on **Tools -> Preferences -> Video Insight**.



Video Insight User Id:

Video Insight User Password:

Video Insight Server Address:

Video Insight Server Port: 9000

□ Use Secure Connection (requires configuration in Video Insight server)

☑ Enable Camera View Flash

- □ For Unrecognizable Faces
- ☑ For Strangers
- ☑ For Enrolled
- ☑ For Concerns and Threats
- Flash Duration: 3 seconds

☑ Pop-up Alert Message

- □ For Unrecognizable Faces
- □ For Strangers
- □ For Enrolled
- ☑ For Concerns and Threats

- **Video Insight User Id**: User Id of a Video Insight user with the necessary permissions to connect SAFR to Video Insight.
- **Video Insight User Password**: Password of a Video Insight user with the necessary permissions to connect SAFR to Video Insight.
- **Video Insight Server Address**: IP Address of the Video Insight server.
- **Video Insight Server Port**: Port number of the Video Insight server.
- **Use Secure Connetion**: Enables the use of an https connection between Video Insight and SAFR.
- **Enable Camera View Flash**: When enabled, the camera views within the Video Insight VMS system will flash when the specified person types are seen by Video Insight cameras. For more information, see the Camera View Flash section of the SAFR Panasonic Video Insight Operation Guide.
  - **For Unrecognizable Faces**: Enables camera view flashes when unrecognizable faces are seen.
  - **For Strangers**: Enables camera view flashes when people who aren't enrolled in SAFR are seen.
  - **For Enrolled**: Enables camera view flashes when people who are enrolled in SAFR and who aren't flagged as persons of concern or threats are seen.
  - **For Concerns and Threats**: Enables camera view flashes when people who are enrolled in SAFR and who are flagged as persons of concern or threats are seen.
  - **Flash Duration**: Specifies how long the triggered camera view flashes should last.
- **Pop-up Alert Message**: When enabled, the camera views within the Video Insight VMS system will display pop-up alert messages when the specified person types are seen by Video Insight cameras. For more information, see the Pop-Up Alert Messages section of the SAFR Panasonic Video Insight Operation Guide.
  - **For Unrecognizable Faces**: Enables pop-up alert messages when unrecognizable faces are seen.
  - **For Strangers**: Enables pop-up alert messages when people who aren't enrolled in SAFR are seen.
  - **For Enrolled**: Enables pop-up alert messages when people who are enrolled in SAFR and who aren't flagged as persons of concern or threats are seen.

- **For Concerns and Threats**: Enables pop-up alert messages when people who are enrolled in SAFR and who are flagged as persons of concern or threats are seen.

115.2.2 Verify the Connection to Video Insight

To verify that your SAFR system successfully connected to the Video Insight system, do the following:

1. Open the Camera Feed Analyzer in the SAFR Desktop Client.
2. Click on the camera selection drop-down menu at the top of the *Camera Feed Analyzer*.
3. You should see the cameras connected to Video Insight among the available camera. (All Video Insight cameras have the camera's IP address or hostname appended to the beginning of the camera name.)
   - If you don't see any Video Insight cameras listed, try shutting down and then restarting your SAFR Desktop Client.

116 SAFR Panasonic Video Insight Operation Guide

116.1 Camera View Flash

When enabled, faces that are seen by Video Insight cameras cause the Video Insight camera view to flash a color. Which color is flashed depends on what type of person is seen.

The camera view flash will only occur when the viewing camera is running a SAFR Video Processing Mode that supports generating events based on the person type viewed. For example, a camera running the *Enrolled and Stranger Monitoring* mode will generate a camera view flash whenever anybody is viewed, whereas cameras running the *Recognition* mode will never generate a camera view flash.

Person Type	Color
Unrecognizable face	White
Stranger (i.e. Person who isn't enrolled in SAFR)	Purple
Enrolled person without a name who isn't flagged as "Concern" or "Threat"	Blue
Enrolled person with a name who isn't flagged as "Concern" or "Threat"	Green
Enrolled person flagged as "Concern" (with or without a name)	Orange
Enrolled person flagged as "Threat" (with or without a name)	Red

116.2 Pop-Up Alert Messages

When enabled, people that are seen by Video Insight cameras cause pop-up alert messages to appear on Video Insight camera views. These messages can be generated when either a face is detected (if face detection is enabled in SAFR's Detection Preferences), a person's body is detected (if person detection is enabled in SAFR's Detection Preferences), or both.

Pop-up alert messages are only generated when the camera viewing the person is running a SAFR Video Processing Mode that supports generating events based on the person type viewed. For example, a camera running the *Enrolled and Stranger Monitoring* mode can generate any of the pop-up alert messages, whereas cameras running the *Recognition* mode can't generate any messages.

Below are all the pop-up alert messages that can be generated.

Event Message	Id Class	Named	Person Type	Condition	People Attributes
Unrecognizable person detected.	N/A	N/A	N/A	Face detected but insufficient information for recognition	idClass="unidentified"
Stranger detected	Stranger	N/A	N/A	Face detected but not found in registered people	idClass="stranger"
Enrolled person detected without name. 98.2% match.	Normal	No	None	Enrolled person without name or person type assigned	idClass="noconcern" && personType="" && name=""

Event Message	Id Class	Named	Person Type	Condition	People Attributes
Enrolled person detected with name <name>. 98.2% match.	Normal	Yes	None	Enrolled person with name but no person type	idClass="noconcern" && personType="" && name=<name>
Enrolled person detected of type <personType>. 98.2% match.	Normal	No	Defined	Enrolled person with person type but no name	idClass="noconcern" && personType=<personType> && name=""
Enrolled person detected of type <personType> with name <name>. 98.2% match.	Normal	Yes	Defined	Enrolled person with person type and name	idClass="noconcern" && personType=<personType> && name=<name>
Smile activation by enrolled person of type <personType> without a name. 98.2% match.	Normal	No	Defined	Smile activation by an enrolled person with person type but no name	personType=<personType> && name=""
Smile activation by enrolled person of type <personType> with name <name>. 98.2% match.	Normal	Yes	Defined	Smile activation by an enrolled person with person type and name	personType=<personType> && name=<name>
Concern person detected without a name. 98.2% match.	Concern	No	None	Same as above for Concern	idClass="concern" && personType="" && name=""
Concern person detected with name <name>. 98.2% match.	Concern	Yes	None	Same as above for Concern	idClass="concern" && personType="" && name=<name>

Event Message	Id Class	Named	Person Type	Condition	People Attributes
Concern person detected of type <personType>. 98.2% match.	Concern	No	Defined	Same as above for Concern	idClass="concern" && personType=<personType> && name=""
Concern person detected of type <personType> with name <name>. 98.2% match.	Concern	Yes	Defined	Same as above for Concern	idClass="concern" && personType=<personType> && name=<name>
Threat person detected without a name. 98.2% match.	Threat	No	None	Same as above for Threat	idClass="threat" && personType="" && name=""
Threat person detected with name <name>. 98.2% match.	Threat	Yes	None	Same as above for Threat	idClass="threat" && personType="" && name=<name>
Threat person detected of type <personType>. 98.2% match.	Threat	No	Defined	Same as above for Threat	idClass="threat" && personType=<personType> && name=""
Threat person detected of type <personType> with name <name>. 98.2% match.	Threat	Yes	Defined	Same as above for Threat	idClass="threat" && personType=<personType> && name=<name>

117 July 2021 Release Notes

117.1 Web Console

- Enhanced upload and snap face

117.2 Windows

117.2.1 Lite Desktop Client

- Added support for *.webm and *.mkv video formats
  - Also added support for the VP8 video codec
- Enhanced the person record GUI
- Improved event image quality selection and updates
  - When transitioning from match < 100% to match >= 100%, updates events immediately
  - Recognized images with a higher % are always better than ones with lower % match
  - Fully visible faces (i.e. not clipped by edge of the screen) are considered of higher quality than clipped faces.
- Added keyboard short-cuts to search results GUI
  - Cursor keys navigate the candidate grid.
  - Enter key displays Face image/Scene blow-up dialog of the selected candidate.
  - ESC closes the Face image/Scene blow-up dialog
- Enhanced person record changes audit tracking
  - Site and source where the last edits were made are now tracked.

117.2.2 Windows Desktop Client

- Enhanced RGB Liveness accuracy
  - Texture model increased its true positive rate
- Optimized small input size face detection for 16:9 video
  - 16:9 aspect ratio 320x180 input size face detection model was added (in addition to 320x240)
- Enabled video viewing for Monitor role users in Operator Console
- Enhanced face detector accuracy on non-standard aspect ratios
  - Aspect-fit pre-scaling is now applied this eliminating aspect ration distortions and enhancing face localization and thus recognition accuracy.
- Added support for accelerator.gpu-id feed property in VIRGO
  - This property allows for explicit assignment of feeds to GPU on multi-gpu systems

117.2.3 Windows Platform

- Admin API to obtain all directories in use by tenant
- Added ARES command line option (-s) for saving credentials
  - Credentials are saved in SAFRActions.config in encrypted form.
- Added 3D liveness model support
  - Compatible with Windows support for Intel's RealSense cameras
- Added non-FIFO queue handling to Queue Dashboard
- Enhanced server fall-back face detector accuracy.
  - 480x480 retinaface detector is used as fall-back detector in HTFS
  - This enhances enrollment reliability from images via web-console and APIs.
  - It also enhances biometric event indexing reliability.
  - It increase use of GPU memory usage by ~250GB (overall - not per feed).
  - It can disabled in face service config file via face_detection_fallback=false to reduce GPU memory usage.
- Reduced MongoDB default RAM (cache) footprint:
  - MongoDB cache configuration for different system RAM sizes:
    - [0 .. 16GB> RAM: set cache to 1GB

        - [16GB .. 48GB> RAM: set cache to 1/3 RAM - 1GB
        - [48GB .. > RAM: set cache to 1/2 RAM - 1GB (MongoDB default config)
- Enhanced display of wait times in Queue Dashboard
- Added face selection query parameters
- Enhanced event server indexes/query speeds
    - CV Event Server Indices
- Enhanced access logs with client origin IP addresses
- Enhanced system auto-recovery robustness on MongoDB out of memory conditions
    - Enhanced robustness of COVI start and re-start when MongoDB is offline
- Upgraded Apache for security patches
- Enhanced reliability of video file processing by VIRGO
    - Starting video frames are now ensured not to be skipped due to slow initialization.

117.3 Linux Platform

- Added RBG Liveness support
- Added RBG Liveness Action support
- Added Secure Access with Smile and RBG Liveness support
- Added support for *.webm and *.mkv video formats
    - Also added support for the VP8 video codec
- Reduced MongoDB binary size:
    - 740MB -> 56 MB
- All of the Windows Platform changes with the following exceptions:
    - Windows Desktop Client changes aren't included.
    - 3D liveness model support isn't included.

117.4 Jetson Platform

- All of the Linux Platform changes

117.5 Android Mobile Client

- Android SAFR 3.5 enhancements
- Enabled imageTime event attribute

117.6 iOS Mobile Client

- Updated the default mask threshold to 0.5. MaskCheck remains at 0.7.
- Added a new global URLSessionDelegate that can be used to add IPAddress certificate exceptions for all the sessions in the app.
- If the similarity score goes over 100% and the current similarity score for the event is less than 100% then update the image. This can only occur once. This was ported from Windows.
- The age, gender, occlusion, and mask icons are not supposed to show up in "Detect Mask" mode. In this mode SAFR is meant to work very similar to MaskCheck. When in "Detect Mask" mode the detect mask action is also enabled.
- The source rotation of videos is now handled. The metadata is now checked to get the display matrix and then get the rotation angle. The source rotation is now saved off.

117.7 SDK

- Enabled the imageTime event attribute

117.8 Embedded SDK

- Added aspect-fit pre-scaling step for high sensitivity face detector

  - Increased detection accuracy on non-standard image sizes (aspect-ratios).
- Added support for 640x360 and 360x640 aspect ratios.
- Increased accuracy of the high sensitivity quantized models.

118 May 2021 Release Notes

118.1 Web Console

- Enhanced Search By Image
- Improved VIRGO feed error handling

118.2 Windows

118.2.1 Lite Desktop Client

- Copy event to clipboard
- Biometric Event Indexing Config
- Retrospective Event Search GUI
- Event time-stamp sync to Server

118.2.2 Windows Desktop Client

- All the Lite Desktop Client changes
- Secure Access with Smile and RGB Liveness video processing mode
- Improved VIRGO feed error handling

118.2.3 Windows Platform

- All the Windows Desktop Client changes
- HTFS detection fallback
- Enhanced face/no-face classifier
- IP Address change detection and auto-reconfig
- Expanded backup and restore
- MongoDB memory requirements

118.3 Linux

118.3.1 Linux Platform

- All of SAFR Windows Platform changes except for Desktop Client UI changes

118.4 Jetson

118.4.1 Jetson Platform

- All of SAFR Windows Platform changes except for Desktop Client UI changes

118.5 SAFR SDK

- All platforms:
  - RGB liveness support
  - Bug fixes

118.6 Embedded SDK

- All platforms:
  - RGB Liveness support
  - Embedded SDK License Enforcement v3
  - Person store search optimizations:
  - Memory leak bug fixes

118.7 Android Mobile Client

- RGB Liveness support
- Bug fixes

118.8 iOS Mobile Client

- Bug fixes

119 March 2021 Release Notes

119.1 Web Console

- SMS Watchlist Alarms
- Occupancy Reports improvements:
    - Occupancy dashboard changed to Occupancy Areas Editor
    - Shortened occupancy dashboard URL
    - Added occupancy at reset time to be entered in Occupancy Areas Editor
    - Optimized responsiveness and of Sites and Sources pull-downs
- Web console event archive improvement:
    - Web Console live event updates will be turned off when sorting by Age, Gender, or Duration.

119.2 Windows

119.2.1 Lite Desktop Client

- Restricting users to directories
- Event filter presets
- Japanese language support

119.2.2 Windows Desktop Client

- All the Windows Desktop Lite Client changes
- RGB Liveness:
    - RGB liveness integration enhancements
    - RGB Liveness Action
    - Capability profile:
        - Developed with equal demographic representation.
        - Masks are supported
- Robust faster-than-real-time file processing (~4x).

119.2.3 Windows SAFR Platform

- All the SAFR Windows Desktop Client changes.

119.3 Linux

119.3.1 SAFR Linux Platform

- Event filter presets

119.4 Jetson

- Enabled person detection on Nano (tiny model).
- Enabled support for Age, Gender, Sentiment, Mask on Nano:

119.5 Android Mobile Client

- Support for SMS Watchlist Alarms deep links.
- Support for https://<IP address> connection to server.
- Bug fixes

119.6 SAFR SDK

- Android:
    - Support for https://<IP address> connection to server.

- Bug fixes

120 February 2021 Release Notes

120.1 Web Console

- Web Console People and Events tabs image mouse-over enhancements

120.2 Windows

120.2.1 Desktop Lite Client

- Search by Image
- Search By Image Initiation Points
- Select Event Deletion
- alarm.mail.recipients configuration for VIRGA Health Monitoring.

120.2.2 Windows Desktop Client

- All the Windows Desktop Lite Client changes
- Windows D3D11 to CUDA buffer detection optimizations
- Added SAFR Actions support
- SAFR Edge installer has been removed.

120.2.3 Windows SAFR Platform

- Cross platform event archiving
- Cross platform directory sync fixes.
- Enabled alarm.mail.recipients for VIRGA Health Monitoring
- Select Event Deletion
- Search By Image Initiation Points
- Enhanced Event Query Mongo Performance:
    - Last 24hrs query:
        - Improved from 4.3 seconds to 0.022 seconds.
    - Median data query:
        - Improved from 3.63 seconds to 0.82 seconds.

120.3 macOS

120.3.1 macOS Desktop Client

- Bug fixes

120.4 Linux

120.4.1 Linux SAFR Platform

- Cross platform event archiving
- Cross platform directory sync fixes.
- Enabled alarm.mail.recipients for VIRGA Health Monitoring.
- Select Event Deletion
- Search By Image Initiation Points
- Enhanced Event Query Mongo Performance:
    - Last 24hrs query:
        - Improved from 4.3 seconds to 0.022 seconds.
    - Median data query:
        - Improved from 3.63 seconds to 0.82 seconds.

120.5 Android Mobile Client

- MaskCheck by COV-IRT:
  - Added “Enable MaskCheck mistake reporting” to MaskCheck Settings
  - 40% higher frame rate
  - Removed extraneous event reporting
  - Bug fixes
- SAFR Recognition:
  - Enhanced event reporting reliability
  - 40% higher frame rate
  - Removed ffmpeg dependencies.
  - Bug fixes

120.6 iOS Mobile Client

- MackCheck by COV-IRT:
  - Added “Enable MaskCheck mistake reporting” to MaskCheck Settings.
- Removed ffmpeg dependencies.

120.7 SAFR SDK

- Enhanced event reporting reliability
- 40% higher frame rate
- Bug fixes

120.8 Embedded SDK

- Added mask detection support on the following platforms:
  - Android (armeabi-v7a and arm64-v8a architectures)
  - Windows (standard and lite)
  - Linux x86 (standard and lite)
  - Linux ARM
  - Jetson

121 December 2020 Release Notes

121.1 Web Console

- Occupancy Reports and dashboards:
  - Occupancy Areas Editor
  - Occupancy Alarms Dashboard
  - Occupancy Report Specification
- Search by image

121.2 Windows

121.2.1 Lite Desktop Client

- Added support for .mxf file format processing and MPEG2 decoding

121.2.2 Windows Desktop Client

- All the Windows Lite Desktop Client changes
- Video Insight VMS Integration
- Geutebrueck VMS Integration
- Fixed Retinaface detector integration on cropped videos

121.2.3 Windows SAFR Edge

- All the Windows Desktop Client changes

121.2.4 Windows SAFR Platform

- All the Windows Desktop Client changes
- COVI:
  - Optimized operation when detect-mask=false and there is no possibility for signature to be inserted or updated.
  - Face recognition signature for masked face matching is not computed under the following settings:
    - Mask detection is disabled and
    - A non-learning mode is used and
    - Update identity every 0 days and
    - Update identity with better image is disabled
- Events:
  - Paged Event Queries
- VIRGA:
  - Improved handling of task queueing operation:
    - POST /task API

121.3 Linux (Ubuntu, CentOS, Amazon Linux 2, and Jetson)

121.3.1 Linux SAFR Platform

- All the Windows SAFR Platform changes

122 November 2020 Release Notes

122.1 Web Console

- Added display of MaskCheck and TempCheck events in event archive
- Event tab expanded filter state is not preserved within a session
- When Occlusion and Masks are detected at the same time, only the mask icon (i.e. not the occlusion icon) is shown in the video overlays in the video feed viewer, analyzer, and event archive
- Added Person count header to the People page

122.2 Windows

122.2.1 Lite Desktop Client

- Added display of MaskCheck and TempCheck events in event archive
- Added View > Face Landmarks to video analyzer
- Fixed the display of overlays and options for docked live video view overlay
- When Occlusion and Masks are detected at the same time, only the mask icon (i.e. not the occlusion icon) is shown in the video overlay in the video feed viewer, analyzer, and event archive

122.2.2 Windows Desktop Client

- All the Lite Desktop Client changes

122.2.3 Windows SAFR Edge

- All the Windows Desktop Client changes

122.2.4 Windows SAFR Platform

- All the Windows Desktop Client changes
- COVI:
  - Image quality comparator fix
- VIRGA:
  - Enabled VIRGA to handle same X-CLIENT-ID in different tenants
    - VIRGA treats same X-CLIENT-ID in different tenants as separate VIRGOS which can coexist and operate simultaneously without any issues
    - This allows the same SAFR Inside camera to change tenants back and forth without configuration being lost for it within each tenant.

122.3 Linux (Ubuntu, CentOS, Amazon Linux 2, Jetson)

- VIRGO:
  - bug fixes
- All the Windows SAFR Platform changes

122.4 Android Mobile Client

- Bug fixes

122.5 SAFR SDK

- Android:
  - Bug fixes

122.6 SAFR eSDK

- Android:
  - Bug fixes

123 October 2020 Release Notes

123.1 Web Console

- Mask Detection Dashboard
- Enhanced Mask detection integration - new Video Feed parameters
- Support for static feed VIRGO configuration
- VIRGA Health Monitoring

123.2 Windows

123.2.1 Lite Desktop Client

- VIRGA Health Monitoring - Configuration and Alarm Status display
- Support for static feed VIRGO configuration
- Expanding Import from Event Archive to unrecognizable faces
- File > Logout menu item
- Integration of facial recognition optimized for masks

123.2.2 Windows Desktop Client

- All the Lite Desktop Client changes
- Retinaface face detector integration

123.2.3 Windows SAFR Edge

- All the Windows Desktop Client changes

123.2.4 Windows SAFR Platform

- All the Windows Desktop Client changes
- Support for persistent personId
- Enhanced mask detection integration
- Expanded support for GET/eventCounts to mask and occlusion attributes

123.3 Linux Ubuntu Platform

- Retinaface face detector integration
- Integration of facial recognition optimized for masks
- Added mask and occlusion detection
- Enhanced person tracking
- Enhanced mask detection
- Enhanced GET /eventCounts API to support mask and occlusion queries
- Enhanced persistence of personId
- Bug fixes

123.4 macOS

123.4.1 macOS Desktop Client

- Retinaface face detector integration
- Added mask and occlusion detection support
- Bug fixes

123.5 Android Mobile Client

- Retinaface face detector integration
- Mask and occlusion detection support
- Bug fixes

123.6 iOS Mobile Client

- Events Tab
- Mask Detection mode
- Retinaface face detector integration
- Bug fixes

123.7 SAFR SDK

- Android:
    - Retinaface face detector integration
    - Added mask and occlusion detection
    - Bug fixes
- iOS:
    - Retinaface face detector integration
    - Added mask and occlusion detection
    - Bug fixes

124 August 2020 Release Notes

124.1 Web Console

- Cross platform directory sync: two way sync

124.2 Windows

124.2.1 Lite Desktop Client

- Bug fixes

124.2.2 Windows Desktop Client

- Memory leak fixes related to high volume event reporting
- Genetec overlay misalignment fixes due to aspect ratio discrepancies
- Bug fixes

124.2.3 Windows SAFR Edge

- All the Windows Desktop Client changes

124.2.4 Windows SAFR Platform

- All the Windows Desktop Client changes
- MaskCheck events support for SAFR mask events

124.3 Linux

124.3.1 Linux SAFR Platform

- MaskCheck events support for SAFR mask events

124.4 Cloud

- SAFR Central Event Notifications
- Public Benefit Data Sharing
- MaskCheck events support for SAFR mask events

124.5 iOS

- MaskCheck by SAFR

124.6 SAFR SDK

- Bug fixes

125 July 2020 Release Notes

125.1 Web Console

- Fixed age display
- Added similarity score display on events
- Changed sites/sources for events and reports
- Upgraded Queue Dashboard
- Fixed Person types
- Upgraded Occupancy Dashboard
- Fixed browser-specfic issues

125.2 Windows

125.2.1 Lite Desktop Client

- Person record GUI update
- Mask Detection event/UI integration improvements
- Bug fixes

125.2.2 Windows Desktop Client

- All the Lite Desktop Client changes
- VIRGO uptime stability improvements
- Bug fixes

125.2.3 Windows SAFR Edge

- All the Windows Desktop Client changes

125.2.4 Windows SAFR Platform

- All the Windows Desktop Client changes
- COVI improvements:
    - Enhanced Identity Sync performance and stability
    - Fixed configuration APIs
    - Updated volume statistics
    - Fixed account cleanup
    - Fixed VIRGA script/password cleanup
    - Fixed propagated DOB
    - Updated getClients API permissions
    - Added timeout in face detection
    - Updated logging
- VIRGA improvements:
    - Changed Unauthorized Direction feature rename and default values
    - Updated DB
- Bug fixes

125.3 Linux

125.3.1 Linux SAFR Platform

- COVI improvements:
    - Enhanced Identity Sync performance and stability
    - Fixed configuration APIs
    - Updated volume statistics
    - Fixed account cleanup

    - Fixed VIRGA script/password cleanup
    - Fixed propagated DOB
    - Updated getClients API permissions
    - Added timeout in face detection
    - Updated logging
- VIRGA improvements:
    - Changed Unauthorized Direction feature rename and default values
    - Updated DB
- Bug fixes

125.4 SAFR SDK

- Runtime support DLLs added
- Bug fixes

125.5 Occupancy Report Feature Notes

Occupancy report only displays Count and Age data if person detection or person+face detection is enabled. If only face detection is enabled, then only mask data is reported.

125.6 Follow-up Update

A small follow-up update was released later in July.

- Enhanced the Direction of Travel feature
- Added additional tracking options
- Increased the performance of person detection
- Bug fixes

126 June 2020 Release Notes

126.1 Windows

126.1.1 Lite Desktop Client

- Added an easy way to provide additional images for a person record
- Added new options for conflict resolution while importing images:
  - Skip Import
  - Confirm Match and Add to Existing Person Record
  - Confirm Match and Replace as New Image in Person Record
  - Decline Match and Create New Person Record
- Bug fixes

126.1.2 Windows Desktop Client

- All the Lite Desktop Client changes
- Bug fixes

126.1.3 Windows SAFR Edge

- All the Windows Desktop Client changes

126.1.4 Windows SAFR Platform

- All the Windows Desktop Client changes
- Bug fixes

127 May 2020 Release Notes

127.1 Windows

127.1.1 Lite Desktop Client

- Added Intel RealSense camera support
- Added 3D Liveness Detection (Beta)
- Added Mask Detection integration
- Added easy way to add feeds for background processing
- Bug fixes

127.1.2 Windows Desktop Client

- All the Lite Desktop Client changes
- Added Vehicle Detection (Beta)
- Bug fixes

127.1.3 Windows SAFR Edge

- All the Windows Desktop Client changes

127.1.4 Windows SAFR Platform

- All the Windows Desktop Client changes
- Bug fixes

127.2 SAFR SDK

- Windows:
    - Added Intel RealSense camera support
    - Added 3D Liveness Detection (Beta)
    - Added Mask Detection integration
    - Added Vehicle Detection (Beta)
    - Bug fixes

128 April 2020 Release Notes

128.1 Web Console

- Security fixes
- Detection List and image quality metrics display in remote video feed viewer

128.2 Windows

128.2.1 Lite Desktop Client

- VIRGA Processor Naming and Identification
- Added Video File Analyzer and Camera Feed Analyzer to Operator Console Tools menu
- Detection List and image quality metrics display in remote video feed viewer

128.2.2 Windows Desktop Client

- All the Lite Desktop Client changes
- Contrast Enhancement: Global vs Local contrast enhancement
- Windows VIRGO auto naming based on PC Name

128.2.3 Windows SAFR Edge

- All the Windows Desktop Client changes

128.2.4 Windows SAFR Platform

- All the Windows Desktop Client changes
- Internal database update to keep date of birth reference instead of age
  - This results in people aging in the database with the passage of time.
  - At start, the database silently converts records to new format.

128.3 Linux

128.3.1 SAFR Linux Ubuntu and CentOS Platform

- All the Windows SAFR Platform changes

128.3.2 Jetson

- All the Windows SAFR Platform changes
- Higher efficiency (faster) face recognition leveraging FP16

128.4 macOS

128.4.1 macOS Desktop Client

- All the Lite Desktop Client changes

128.4.2 macOS SAFR Edge

- All the macOS Desktop Client changes

128.4.3 macOS SAFR Platform

- All the Windows SAFR Platform changes

128.5 iOS Mobile Client

- Bug fixes

128.6 Android Mobile Client

- Enable sorting of People alphabetically by last name or by registration date
- Order by last name is now supported by the GET /rootpeople API call
- People can now be searched by name

128.7 SAFR SDK

- Windows:
    - Cropping API Updates
    - Contrast Enhancement: Global vs Local contrast enhancement
    - Bug fixes
- Android:
    - No updates
- Linux:
    - Cropping API Updates
    - Contrast Enhancement: Global vs Local contrast enhancement
    - Bug fixes
- Jetson:
    - Cropping API Updates
    - Contrast Enhancement: Global vs Local contrast enhancement
    - Bug fixes

128.8 Embedded SDK

- Platforms being released:
    - Windows:
        - eSDK-full (includes NVIDIA GPU support, includes auto-selection between non-AVX and AVX2 support)
            - Bug fixes
        - eSDK-lite (no NVIDIA GPU support, includes auto-selection between non-AVX and AVX2 support)
            - Bug fixes
    - Linux x86 Ubuntu 16.04:
        - eSDK-full (includes NVIDIA GPU support, includes auto-selection between non-AVX and AVX2 support)
            - Bug fixes
        - eSDK-lite (no NVIDIA GPU support, includes auto-selection between non-AVX and AVX2 support)
            - Bug fixes
    - Linux ARM Ubuntu 18.04:
        - eSDK-lite (no GPU support)
            - Bug fixes
    - Jetson - Linux ARM Ubuntu 18.04:
        - eSDK-Jetson (NVIDIA GPU support)
            - Bug fixes
    - Android ARM - Android 5.0 or later:
        - eSDK-lite (no GPU support)
            - Bug fixes
        - eSDK-lite 64 bit (no GPU support)
            - Bug fixes

129 March 2020 Release Notes

129.1 Web Console

- Increased Video Viewer Frame Rate video feed viewer
- Video feed viewer overall support
- Event Archive support for Unauthorized Direction of Travel Detection action events.
- Email and SMS Server Configuration in Status Tab
- Support for Unauthorized Direction of Travel Detection feed configuration attributes
- Support for Cropping Parameters feed configuration attributes
- Support for Contrast Enhancement Integration feed configuration attributes
- Support for person detection input size configuration

129.2 Windows

129.2.1 Lite Desktop Client

- SAFR 2.0 UX
    - Live monitoring in single-window app
    - Inline camera settings UX
    - Handling password change for licensor userId
    - Option to disable Operator Console as primary window.
- Increased Video Viewer Frame Rate - support up to 30fps (new platform needed)
    - 30fps, 480p video for local deployments (configurable in VIRGA Tenant Config)
    - 5fps, 480p video for cloud deployments (configurable in VIRGA Tenant Config)
- Video Feed Viewer overlays
    - Right click on feed video to open context menu with overlay options.
- Video contrast enhancement (~20% improvement):
    - Contrast Enhancement Integration
- Unauthorized Direction of Travel Detection configuration and display in Event Archive

129.2.2 Windows Desktop Client

- All the Lite Desktop Client changes
- Avigilon Integration
- More efficient face detection on NVIDIA GPUs
- Person detection input size configuration: NORMAL (default),SMALL, and LARGE.
    - SMALL: 26% faster than NORMAL
    - LARGE: 66% slower than NORMAL
- VIRGO for Windows updated:
    - VIRGO support for multiple remote video feed viewers
    - VIRGO support for video overlays (shown on remote video feed viewers).
    - VIRGO support for video feed Cropping Parameters
    - VIRGO support for Unauthorized Direction of Travel Detection configuration
    - VIRGO support for person detection input size configuration
    - VIRGO support for Contrast Enhancement Integration configuration.
    - VIRGO stability fixes
- Updated higher accuracy person detection model
    - Max Accuracy and Balanced modes improvement: 3%
    - Max Speed mode improvement: 7.1%
    - Balanced vs. Max Speed accuracy advantage: 36.2%

129.2.3 Windows SAFR Edge

- All the Windows Desktop Client changes
- SMS Notifications support in SAFR Actions

    - Support for SMS Server Config in SAFR Actions (AWS SNS)
    - Support for configuring SMS alerts triggered by events in SAFR Actions
- Support for Unauthorized Direction of Travel Detection action events

129.2.4 Windows SAFR Platform

- Age Model update with accuracy age recognition model
    - 15% improvement on Asian faces
    - 9.4% general improvement
- Increased Video Viewer Frame Rate
- Security Patches
- All the Windows SAFR Edge changes

129.3 Linux

129.3.1 SAFR Linux Ubuntu and CentOS Platform

- All the Windows SAFR Platform changes

129.4 Jetson

- Person detection added
- Higher efficient (faster) face detection
- All the Windows SAFR Platform changes
- All the Windows SAFR Platform changes

129.5 macOS

129.5.1 macOS Desktop Client

- Increased Video Viewer Frame Rate
    - support up to 30fps (new platform needed)
    - 30fps, 480p video for local SAFR Platform (configurable in VIRGA Tenant Config)
    - 5fps, 480p video for Cloud SAFR Platform (configurable in VIRGA Tenant Config)
- Video Feed Viewer overlays
    - Right click on feed video to open context menu with overlay options.
- Unauthorized Direction of Travel Detection configuration and display in Event Archive

129.5.2 macOS SAFR Edge

- All the macOS Desktop Client changes

129.5.3 macOS SAFR Platform

- All the SAFR Window Platform changes

129.6 Android Mobile Client

- Addition of Android Events:
    - New side-menu navigation
    - Recent Matches view
    - Watchlist person view
        - Profile
        - Timeline
    - Deep-links from SMS or email
- Bug Fixes

129.7 iOS Mobile Client

- Bug fixes

129.8 SAFR SDK

- Windows:
  - Contrast Enhancement Integration
  - More efficient face detection on NVIDIA GPUs
  - Updated higher accuracy person detection model
- Android:
  - Bug fixes
- Linux:
  - Contrast Enhancement Integration
  - Updated higher accuracy person detection model
- Jetson:
  - Contrast Enhancement Integration
  - More efficient face detection on NVIDIA GPUs
  - Updated higher accuracy person detection model

129.9 Embedded SDK

- Platforms being released:
  - Windows:
    - eSDK-full (includes NVIDIA GPU support, includes auto-selection between non-AVX and AVX2 support)
      - Person detection
      - Bug fixes
    - eSDK-lite (no NVIDIA GPU support, includes auto-selection between non-AVX and AVX2 support)
      - Bug fixes
  - Linux x86 Ubuntu 16.04:
    - eSDK-full (includes NVIDIA GPU support, includes auto-selection between non-AVX and AVX2 support)
      - Person detection
      - Bug fixes
    - eSDK-lite (no NVIDIA GPU support, includes auto-selection between non-AVX and AVX2 support)
      - Bug fixes
  - Linux ARM Ubuntu 18.04:
    - eSDK-lite (no GPU support)
      - Bug fixes
  - Jetson - Linux ARM Ubuntu 18.04:
    - eSDK-Jetson (NVIDIA GPU support)
      - Person detection
      - More efficient face detection
      - Bug fixes
  - Android ARM - Android 5.0 or later:
    - eSDK-lite (no GPU support)
      - Bug fixes
    - eSDK-lite 64 bit (no GPU support)
      - Bug fixes

130 January 2020 Release Notes

130.1 Web Console

- New report: Queue Dashboard.
- Traversal Dashboard improvements.
- Traffic Dashboard optimizations.
- Attendance Dashboard enhancement.

130.2 Windows

130.2.1 Lite Desktop Client

- Sign-in UX changes to support operator workflows.
- Option to require sign-in on every start.
- User Administration.
- Option to disable Windows auto-update when in SAFR Kiosk Mode.
- Video Feed Viewer hides stats by default. Right click to display stats.
- Video Feed Viewer supports 10fps, 720p video (new platform needed).
- Genetec FR Plugin Improved SSL error handling and GUI option to turn off SSL.

130.2.2 Windows Desktop Client

- All the Lite Desktop Client changes.
- VIRGO for Windows stability fixes.

130.2.3 Windows SAFR Edge

- All the Windows Desktop Client changes.

130.2.4 Windows SAFR Platform

- All the Windows Desktop Client changes.
- Installer options to install without SAFR Desktop and to customized path.
- Returned installer option to force CPU Face Recognition service.
- Initiated model initialization during installation to reduce initialization time upon launch.

130.3 Linux

130.3.1 Linux Ubuntu and CentOS SAFR Platform

- VIRGO enhancements.

130.4 Jetson

- SAFR Jetson Ubuntu 18.04 Platform has been implemented.

130.5 macOS

130.5.1 macOS Desktop Client

- Bug fixes.

130.5.2 macOS SAFR Edge

- Bug fixes.

130.5.3 macOS SAFR Platform

- Bug fixes.

130.6 Android Mobile Client

- Added support for arm64-v8 architecture.
- Bug Fixes.

130.7 iOS Mobile Client

- Bug fixes.

130.8 SAFR SDK

- Windows:
  - Added Image analyzer support for person (object) and badge detections.
- Android:
  - Added support for arm64-v8 architecture.
  - Bug fixes.
- Linux:
  - Added Image analyzer support for person (object) and badge detections.
- Jetson:
  - Initial release

130.9 Embedded SDK

- Platforms being released:
  - Windows:
    - eSDK-full (includes NVIDIA GPU support, includes auto-selection between non-AVX and AVX2 support)
    - eSDK-lite (no NVIDIA GPU support, includes auto-selection between non-AVX and AVX2 support)
    - Changes:
      - Bug fixes.

131 December 2019 Release Notes

131.1 Web Console

- New Traversal Dashboard report

131.2 Windows

131.2.1 Lite Desktop Client

- Enhanced Event Archive GUI
- Person activity view
- CBP Face Acquisition System
- Advanced configuration of Center Pose Quality for strangers/learning: pitch, roll, and yaw
- Identity retention configuration
- Full Screen, Locked Screen, Auto-restart, Auto-logon, and Kiosk mode for Windows

131.2.2 Windows Desktop Client

- All the Lite Desktop Client changes
- Enhanced Person Detection Accuracy - especially in crowded scenes
- Ximea Camera Integration

131.2.3 Windows SAFR Edge

- All the Windows Desktop Client changes

131.2.4 Windows SAFR Platform

- All the Windows Desktop Client changes
- All the System Console changes
- Concurrent face matching (3.5X lower matching latency on 8 core processor)
- Higher face-recognition throughput on non-GPU machines
- SAFR offline licensing

131.3 Linux

131.3.1 Linux Ubuntu VIRGO

- Person-face consolidated tracking enhancements
- Advanced configuration of Center Pose Quality for strangers/learning: pitch, roll, and yaw
- Enhanced Person Detection Accuracy - especially in crowded scenes

131.3.2 Linux Ubuntu and CentOS SAFR Platform

- All the Linux VIRGO changes
- All the System Console changes
- Concurrent face matching (3.5X lower matching latency on 8 core processor)
- Higher face-recognition throughput on non-GPU machines
- Identity retention configuration
- SAFR offline licensing

131.4 macOS

131.4.1 macOS Desktop Client

- Person-face consolidated tracking enhancements

- Event retention configuration GUI revision
- Identity retention configuration GUI
- Advanced configuration of Center Pose Quality for strangers/learning: pitch, roll, and yaw

131.4.2 macOS SAFR Edge

- All the macOS Desktop Client changes

131.4.3 macOS SAFR Platform

- All the macOS Desktop Client changes
- SAFR offline licensing

131.5 Cloud

- Concurrent face matching (3.5X lower matching latency on 8 core processor)
- Identity retention configuration

131.6 Android Mobile Client

- Hardware Video Decode
- Active Camera Connect
- Bug Fixes

131.7 iOS Mobile Client

- Dark mode bug fixes

131.8 SAFR SDK

- Windows:
  - Enhanced Person Detection Accuracy - especially in crowded scenes
- Linux:
  - Bug fixes
- macOS:
  - Bug fixes
- Android:
  - Bug fixes
- iOS:
  - Bug fixes

131.9 Embedded SDK

- Addition of new models: Age, Gender, Sentiment, Occlusion, Composite Signatures, Pose Profile, and Face/No-Face
- Platforms being released:
  - Windows:
    - eSDK-full (includes NVIDIA GPU support, includes auto-selection between non-AVX and AVX2 support)
    - eSDK-lite (no NVIDIA GPU support, includes auto-selection between non-AVX and AVX2 support)
  - Linux x86 Ubuntu 16.04:
    - eSDK-full (includes NVIDIA GPU support, includes auto-selection between non-AVX and AVX2 support)

    - eSDK-lite (no NVIDIA GPU support, includes auto-selection between non-AVX and AVX2 support)
- Linux ARM Ubuntu 18.04:
    - eSDK-lite (no GPU support)
- Jetson - Linux ARM Ubuntu 18.04:
    - eSDK-Jetson (NVIDIA GPU support)
    - Include model compilation/optimization tool
- Android ARM - Android 5.0 or later:
    - eSDK-lite (no GPU support)

132 November 2019 Release Notes

132.1 Web Console

- Support for Anonymous vs. Known identity event retention configuration
- Support for Face-Person enhanced tracking
- Video feed occlusion detection config support
- Video feed config support to limit stranger reporting only to occluded strangers

132.2 ARES

- hasRootEventId filter was added

132.3 Windows

132.3.1 SAFR Windows Desktop Lite

- Support for Occlusion Detection configuration in Video Feeds
- Support for Anonymous vs. Known identity event retention configuration
- Preferences to limit stranger reporting only to occluded strangers
- Improved person import GUI
- Enabled person import directly from Event Archive
- Support for Mobotix Camera Events
- Support for Event time offset for offline videos
- Image quality metrics in Person Details dialog

132.3.2 Desktop Client

- All the SAFR Desktop Lite changes
- Enhanced person-face tracking and reporting
  - Consolidated person/face event reporting
  - Consolidated person/face event display
- Support for false face detection filtering
- Support for Genetec FR Plugin Integration
- Updated Virgo for Windows

132.3.3 SAFR Windows Edge

- Updated SAFR Desktop
- Updated ARES

132.3.4 SAFR Windows Platform

- Higher accuracy Face Recognition Model (v5 signatures)
- Faster (2x) DB Matching
- Redundant CVOS support
- Support for Anonymous vs. Known Identity event retention
- Support for Face No-Face Classification
- Updated System Console
- Updated ARES
- Filtering of secondary faces on import via REST API

132.4 Linux

132.4.1 SAFR Linux Ubuntu VIRGO

- Occlusion Detection

- Enhanced person-face tracking and reporting
  - Consolidated person/face event reporting
  - Consolidated person/face event display
- Face No-Face Classification Integration

132.4.2 SAFR Linux Ubuntu and CentOS Platform

- Higher accuracy Face Recognition Model (v5 signatures)
- Faster (2x) DB Matching
- Redundant CVOS support
- Port conflict resolution at install time
- Support for Anonymous vs. Known Identity event retention
- Support for Face No-Face Classification
- Updated VIRGO
- Updated System Console
- Updated ARES
- Filtering of secondary faces on import via REST API

132.5 macOS

132.5.1 macOS Desktop Client

- Occlusion Detection
- Enhanced person-face tracking and reporting
  - Consolidated person/face event reporting
  - Consolidated person/face event display
- Face No-Face Classification Integration
- Support for Anonymous vs. Known identity event retention configuration
- Model Upgrade GUI

132.5.2 SAFR macOS Edge

- Updated SAFR Desktop
- Updated ARES

132.5.3 SAFR macOS Platform:

- Higher accuracy Face Recognition Model (v5 signatures)
- Faster (10x) DB Matching
- Support for Anonymous vs. Known Identity event retention
- Support for Face No-Face Classification
- Updated System Console
- Updated ARES
- Filtering of secondary faces on import via REST API

132.6 Cloud

132.6.1 SAFR Platform

- Higher accuracy Face Recognition Model (v5 signatures)
- Faster (2x) DB Matching on Windows
- Redundant CVOS support via NFS
- Support for Anonymous vs. Known Identity event retention
- Support for Face No-Face Classification
- Updated System Console
- Filtering of secondary faces on import via REST API

132.6.2 Download Portal

- Updated Android System Requirements
- Removed Create new account link

132.7 Android SAFR App and SDK:

- Bug fixes

132.8 Embedded SDK (Windows and Android)

- Composite signature support
- Faster multi-core face signature matching

133 September 2019 Release Notes

133.1 SAFR Windows

133.1.1 1. Central Video Feed Management

Video feeds on Windows can now be configured and managed centrally for the entire cluster of SAFR Windows (and Linux) Platform machines. This means that a large deployment can be configured from a single machine using the Desktop Client (preferred) or the System Console. SAFR no longer requires video feed windows to remain open, nor do Windows users need to remain logged. SAFR will now also automatically resume processing on system reboot. This makes SAFR on Windows a fully resilient service that can handle power outages and be easily managed even when distributed on many machines.

To enable this, SAFR Windows Platform now comes with Virgo for Windows which performs video feed processing in the background. Windows Virgo supports Genetec, Milestone and Digifort VMS feeds as well as ONVIF, direct RTSP URL, and USB camera feeds. You can configure Windows Virgo via the Windows Desktop Client or the System Console. The Windows Desktop Client is recommended as a configuration tool in all cases and is required if configuring VMS feeds. When adding a feed simply select an auto-detected camera and choose operation mode.

133.1.2 2. Redundant DB Configuration

As was already available on Linux, SAFR Windows Platform can now be configured for redundant DB operation. This means that all DB information (this includes face signatures, person meta-data and events but does not yet include images) will be stored in two or more separate machines and loss of one DB machine will automatically fail-over to another. Redundant DB operation also enables horizontal scalability of the face-matching operation which is distributed across all participating DB machines thus increasing size of deployment achievable (hardware estimator provides number of DB machines needed).

Keep in mind that you must have an odd number of DB machines for automatic failover to function and that the maximum number of redundant DB machines is 50.

133.1.3 3. Watchlist Synchronization across SAFR Platforms and Accounts

SAFR can now be configured to synchronize watchlists from one SAFR Platform or Account to any number of other SAFR Platforms or Accounts. This means that SAFR Platform can now be deployed in a distributed manner with many independent SAFR Platforms at different locations and yet be kept updated with a watchlist maintained centrally (e.g. in Cloud).

You can configure SAFR Platform to synchronize one directory per account (tenant) from the System Console Status tab. Max latency for synchronization is 10 minutes and max throughput is ~20 records per second per sync connection. It might thus take up to 10 minutes to perform initial sync of 10K records.

133.1.4 4. 5X Faster DB Matching Speed

DB matching speed and efficiency have been improved 5x. This means that matches are 5x faster and require 5x less processing power. This translates to significant TCO savings for deployments requiring large watchlists.

On single CPU core, 1 million faces can now be matched in 350-400ms.

133.1.5 5. SAFR Actions for Occlusion

SAFR Actions and Action Relay Event Service (ARES) now supports occlusion event attributes. This means you can configure actions to trigger specifically on occluded faces. For more information, search on “occlusion” in *Action Relay Event Service - ARES manual.*

133.1.6 6. Person (Body) Detection Balanced Mode

Person detection balanced mode delivers 50% more throughput than max accuracy mode with only slight degradation in accuracy. This is now the default mode for person detection and is recommended for all cases when high accuracy of person body detection is needed (e.g. tracking in visually complex environments with several persons present).

In comparison, max speed person detection mode delivers 300% more throughput than balanced mode but with significant reduction in accuracy. However, this mode is commonly adequate for low complexity tracking such as casino tables or teleconferencing rooms.

133.2 SAFR Linux

133.2.1 1. Multi-GPU Scalability

SAFR Linux Platform now offers enhanced scalability across multiple NVIDIA GPUs. SAFR Linux VIRGO has been optimized to be even less reliant on CPU and to maximize use of NVIDIA GPUs. This means that a single large machine can support 6 NVIDIA T4 processors which amounts to a SAFR recognition payload of 90 1080p@15fps feeds or 75 4K@15fps feeds (inclusive of recognition).

This capability is also available in standalone VIRGO Ubuntu download from Developers page.

133.2.2 2. Person Body to Face Recognition Linkage

Person body detection and tracking is now enhanced with face recognition and thus takes on identity established through face recognition. As person body detection is more accurate than face (due to size and being detectable in nearly any orientation) this means that identity tracking with combined person body and face detection is more accurate than face alone. When more accurate account of identity presence before the camera is needed, person events can now be used which are augmented with associated face attributes.

This function is automatically enabled when both person (body) detection and face recognition are enabled.

133.2.3 3. The Following New SAFR Windows features are also now available on Linux

- Watchlist synchronization across SAFR Platforms and Accounts
- 5X faster DB Matching speed
- Person (body) detection balanced mode

133.3 macOS Desktop Client

133.3.1 1. Pose Based Liveness Detection

This features previously introduced on Linux is now also available on macOS. It enables liveness detection based on consistent change in face orientation (pose) as an alternative to smile action. It can be used for walk-up and walk-through secure access scenarios that require liveness confirmation when paired with well positioned cameras.

133.3.2 2. Person Body to Face Recognition Linkage (Same as Linux)

133.4 SAFR Android

133.4.1 Faster SAFR Native Face Detector

SAFR native face detector is now multi-threaded on Android and offers higher frame-rate and accuracy than Google Vision face detector (available when Google Play is present on the device). The Android Mobile Client now delivers excellent face detection performance at ~15fps while utilizing 35% CPU and Google Pixel phone.

133.4.2 Frame Skipping Logic to Maintain Low Latency of Detection and Recognition

When video frame rate is higher than detection rate device can deliver, video frames will be appropriately skipped for analysis in order to not cause backlog of processing that would increase latency in detection and recognition.

133.5 SAFR Embedded SDK (Windows and Android)

1. Person record export/import API
2. Face landmark coordinates (eyes, nose, mouth)
3. Face signature export/import API

133.6 SAFR SDK

Windows:

- Bug Fixes

Android:

- Multi-threaded face detector with higher face detection throughput.
- Frame skipping logic to maintain low latency of detection and recognition.

134 August 2019 Release Notes

134.1 SAFR Windows

- Occlusion Detection:

  SAFR now has the ability to detect faces that are occluded. Occlusion constitutes any obstruction of the key facial features such as from a scarf, hand, glasses, hair draping over the face, etc... This capability is currently integrated to accomplish two features:

  1. To filter out any occluded faces while learning them in the wild and thus prevent storing ambiguous face references in the SAFR person database.

     For example, such a feature is used when learning and memorizing players sitting at the casino table to prevent learning them with an occlusion feature such as a wineglass in front of their face which may later create recognition inaccuracies.

  2. To update occurrence event records with better face images without the occlusion and thus increase the value of the image stored with the event for presentation and investigation purposes.

  You will find the occlusion recognition switch in the **Recognition** tab under SAFR Preferences as well as max tolerable occlusion level adjustment for newly learned faces.

- Core Face Recognition Optimizations for NVIDIA GPUs:

  These optimizations enable up to 463 recognitions per second on NVIDIA GTX 1080Ti graphics cards. This is 14x more recognition throughput in comparison to the maximum achievable on 4 Core 3.4GHz Intel Xeon Skylake-SP processor. The improvement is even more pronounced when all face attributes are computed together (identity, age, gender, sentiment). In such case optimization delivers 320 combined recognitions per second which is 40x more throughput in comparison to maximum achievable on 4 Core 3.4GHz Intel Xeon Skylake-SP processor. These optimizations also reduce recognition latency by 50% and thus enable even faster and more reliable recognition. All this results in cost reductions for on-premise core recognition subsystem deployments from $2,477 to $518 per 100 recognitions per second and from $10,667 to $797 for 100 all-attributes recognitions per second.

  Note that these optimizations introduced a necessary one-time GPU calibration step which is performed when the system is started for the first time with GPU(s) present. It takes about 3 minutes per recognition model (15 minutes total) and per GPU for the system to be properly calibrated. Until this is completed, you will see System Initializing message in video view and recognition will not be be operational.

- Person Body Detection NVIDIA GPU Optimizations

  Person detection speed was improved by 30% and throughout by 50%. This means person detection is faster and more fluid than before. Maximum person detection throughput for our max accuracy model is 115 frames per second on NVIDIA GTX 1080Ti and 329 frames per second on NVIDIA Quadro RTX 6000. Maximum person detection throughput for our max speed model is 625 frames per second on NVIDIA GTX 1080Ti and 1052 frames per second on NVIDIA Quadro RTX 6000.

- Customizable options were added to our popular traffic dashboard (available from the Reports tab in the System Console). These options enable traffic dashboard to be customized in color, logo, language, and time-range. The traffic dashboard can now also be linked directly from another web site and all customization options are available as URL query parameters. This feature enables easy integration of the dashboard into customers' portals who may wish to display the dashboard in colors and logos of their brand.

- A new attendance dashboard was added to the Report tab in the System Console. For a specified time range and location, it displays all recognized individuals in attendance along with the time interval they were observed present. This dashboard can be used as a replacement of punch-card system that tracks employee attendance when properly combined with entry and exit camera monitoring ingress and egress at the work site.

- Installer has been equipped with more customizable options to allow SAFR Logs to be removed from deployment and heap auto-configure behavior to automatically scale memory allocation for SAFR based on system memory available. These options enable SAFR Platform to be deployed on very small PCs (8GB RAM, 32GB Disk, $550) that can independently monitor 2 1080p video feeds. For example, such a small configuration could be used for a small SAFR Platform deployed at a casino table. The heap auto-config also enables SAFR to scale up on larger system and thus reliably handle higher recognition throughput and event traffic.

- To further protect privacy, SAFR now also limits retention of system logs associated with events to the same time frame as configured for events retention in the SAFR database. This means that no trace of individual whereabouts is kept beyond the configured retention time. Recognition logs have also been reduced in their default logging level so as not to include any personally identifiable information (PII).

134.2 SAFR Linux

- The Linux release inherited the following improvements introduced above for Windows:
    - Customizable options for Traffic Dashboard.
    - New Attendance Dashboard.
    - Log retention and log content changes to protect privacy.
- Database fail-over is now enabled on Linux. This means when SAFR is deployed on multiple machines with Database redundancy enabled, failure of the primary machine (containing primary Database) will not degrade secondary nodes that are running redundant Database from full functionality.

134.3 SAFR SDK

- RTSP support has been added to iOS and Android SAFR SDK. This means that SAFR SDK can now process video feeds delivered via rtsp protocol widely supported by IP cameras and can be thus used to process video feeds from a detached camera. For example, iOS or Android device can be used to process video feed from body camera connected to the device via WiFi.
- iOS SAFR SDK is available in our Partner Cloud and Production environment.
- Android SAFR SDK is available in our Partner Cloud environment and will be further validated and pushed to production next week.
- Windows SAFR SDK has person body detection added to its capabilities which enables developers to implement alerts based on body detection and traffic counting. Also new in Windows SAFR SDK is availability of pitch, roll and yaw face attributes which describe orientation of the face around all three axis.

134.4 Mobile Clients

- iOS and Android Mobile Clients have been equipped with same RTSP support described above for SAFR SDK. To connect an RTSP feed, press-and-hold camera selection button in bottom right corner. You will be able to register several RTSP feeds that will be stored and made available for selection.
- iOS SAFR application is awaiting review by Apple and will be available next week in the app-store.
- Android SAFR application will also be available next week on SAFR Partner and Production Cloud portal.

134.5 SAFR Cloud

- Occlusion detection is now available in SAFR Cloud and can be utilized by developers via SAFR REST APIs or be used through the Desktop Client for Windows.
- Customizable Traffic Dashboard and Attendance Dashboard described above are also available in SAFR Cloud.

134.6 SAFR Stability

- 67 defects were fixed for this release.

134.7 Follow-up Update

A small follow-up update was released later in August.

1. The Mobile Client for Android was released with the following new capabilities:
   - RTSP video feeds are now supported. This means that Mobile Clients can now process video feeds delivered via RTSP protocol widely supported by IP cameras and can be thus used to process video feeds from a detached camera. For example, Android devices can be used to process video feeds from body cameras connected to the device via WiFi. To connect an RTSP feed, long-press camera selection button in bottom right corner. You will be able to register several RTSP feeds that will be stored and made available for selection.
   - Google Play Services are no longer required on Android device. SAFR now includes own SAFR face detector. You can switch between Google and SAFR detectors for integrated camera use. SAFR face detector provides higher detection accuracy but is slightly slower when processing feeds from devices integrated camera due image conversion overhead which we will look to eliminate in the future. RTSP feeds are always processed via SAFR face detector which offers higher detection accuracy and speed over Google supplied face detector.
2. SAFR Cloud, SAFR Windows Platform, SAFR Windows SDK, and SAFR Android SDK were released with a few more bug fixes.