MAKE THE
WORLD SEE

Milestone Systems

Milestone AI Bridge 2.0.0

Integrator manual

XProtect Corporate
XProtect Expert
XProtect Professional+
XProtect Express+
XProtect Essential+

milestone

Contents

Copyright, trademarks, and disclaimer

Trademarks

XProtect is a registered trademark of Milestone Systems A/S.

Microsoft and Windows are registered trademarks of Microsoft Corporation. App Store is a service mark of Apple Inc. Android is a trademark of Google Inc.

All other trademarks mentioned in this document are trademarks of their respective owners.

Disclaimer

This text is intended for general information purposes only, and due care has been taken in its preparation.

Any risk arising from the use of this information rests with the recipient, and nothing herein should be construed as constituting any kind of warranty.

Milestone Systems A/S reserves the right to make adjustments without prior notification.

All names of people and organizations used in the examples in this text are fictitious. Any resemblance to any actual organization or person, living or dead, is purely coincidental and unintended.

This product may make use of third-party software for which specific terms and conditions may apply. When that is the case, you can find more information in the file `3rd_party_software_terms_and_conditions.txt` located in your Milestone system installation folder.

Overview

This and related documentation and eLearning courses

This documentation describes how to deploy and set up integration between Milestone AI Bridge, the IVA applications and the processing servers which hosts the IVA applications.

The target audience of this documentation is expected to be IVA application integrators, IVA administrators and other technical personnel that will be managing the deployment, set up and maintenance of the Milestone AI Bridge.

The configuration, maintenance, and behavior of a typical XProtect VMS, Milestone AI Bridge, and IVA applications integration is described in an IVA application agnostic way.

This documentation also illustrates how and when you need collaborate with the administrators of the XProtect VMS.

Depending on your contract with organization with the XProtect VMS, you may handle some of the tasks in the XProtect VMS tasks on behalf of the administrator. How administrators of XProtect VMS systems and you as IVA app integrator decide to share the tasks of setting up the full end-to-end Milestone AI Bridge integration is individual for each organization.

Administrators of XProtect Management Client should set up and manage the integration with IVA applications[1] through Milestone AI Bridge.

Related documentation

There is a separate manual for administrators of XProtect VMS that describes how administrators of XProtect Management Client should set up and manage the integration with IVA applications[2] through Milestone AI Bridge from inside XProtect Management Client.

This administrator manual describes the configuration, maintenance, and behavior of a typical Milestone AI Bridge and IVA applications integration in an IVA application agnostic way. See the administrator manual for Milestone AI Bridge.

For specific functionality of your IVA applications, read the manuals for your IVA applications.

eLearning courses

Milestone offers eLearning courses for all XProtect products. Visit the Milestone Learning Portal at https://learn.milestonesys.com/index.htm.

---

[1] A software program that analyzes video for objects and the behavior of objects.

[2] A software program that analyzes video for objects and the behavior of objects.

With this release of Milestone AI Bridge, there are, however, no Milestone AI Bridge eLearning courses. But when there are, search for **ai bridge** to find the Milestone AI Bridge courses.

Milestone AI Bridge

Milestone AI Bridge is a MIP SDK developed using cloud-native technologies. Milestone AI Bridge acts as a bridge between installations of XProtect Video Management Software (VMS) and Intelligent Video Analytics (IVA) applications deployed as Open Container Initiative (OCI) images, and enables the exchange of data between these two types of applications.

You can strengthen your security and business understanding by integrating your XProtect VMS with IVA applications through Milestone AI Bridge.

Milestone AI Bridge forwards video streams from cameras added to the XProtect VMS to the IVA applications for video analysis. Milestone AI Bridge allows the IVA applications to send the analysis results back into your XProtect VMS as analytics data (events, metadata, and video).

Video analysis of audio data is currently not supported.

Architecture overview and system components

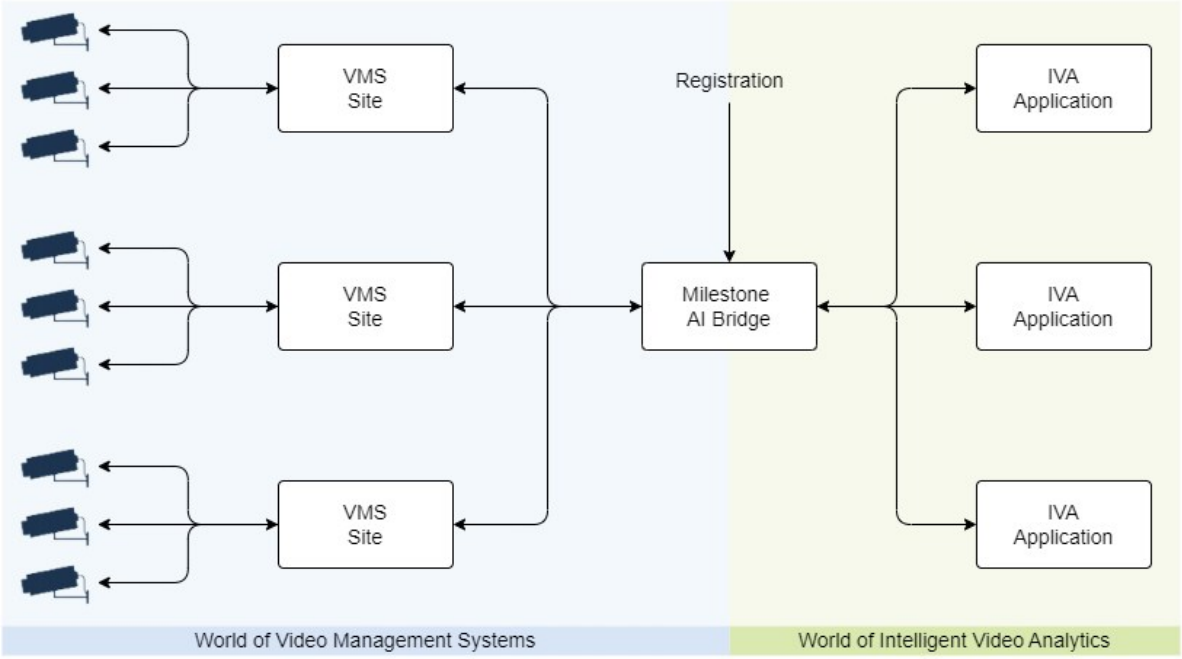
Bridging two worlds

VMS applications and IVA applications are, by nature, very different. The code bases used to develop the applications and the operating systems they run on are different. While VMS applications predominately use Microsoft Windows technology, many IVA applications are developed for the Linux operating system and are often containerized. containers[1] .

The purpose of Milestone AI Bridge is to make it easy for developers of IVA applications to integrate their solution with the XProtect VMS.

While VMS applications and IVA applications cannot communicate directly with each other, they can communicate indirectly by utilizing the Milestone Integration Platform (MIP) and the Milestone AI Bridge.

[1]A small application with a limited function like a website, a service, a database, or other. Milestone AI Bridge consists of 10 containers.



Processing server

When you and your IVA application integrator agree on the best processing server[1], you should also consider the complexity of the IVA applications, the demands for analytics, and the needs for performance, scalability, and resiliency. You can choose between two processing server hosts: the simpler Docker Compose consisting of one server or the more advanced Kubernetes consisting of a cluster of servers.

Application registration

When your IVA application integrator installs Milestone AI Bridge and your IVA applications, the IVA applications that are self-registering are automatically registered in Milestone AI Bridge.

IVA applications that are not self-registering can be modified to include the registration for the IVA application in question in the register.graphql file or can be included by running the register graphql mutation. When your IVA application integrator enters the login credentials for the Milestone AI Bridge basic user you have created into Milestone AI Bridge, Milestone AI Bridge is automatically registered in your XProtect VMS.

[1]One server or a cluster of servers that processes some kind of data. In connection with Milestone AI Bridge, the processing server hosts Milestone AI Bridge and the IVA applications. Typically, the video analysis happens on the processing server.

Data exchange

Most communication and exchange of data between the XProtect VMS and the IVA applications happens through Milestone AI Bridge by using various standard protocols. The only exception is the web page on which you draw the graphical representation of the selected analytics topic. This web page is sent directly from the IVA application to XProtect Management Client.

It is recommended that you secure all the communication between the XProtect VMS, Milestone AI Bridge, and the IVA applications[1].

From the XProtect VMS to IVA applications

The XProtect VMS shares the following camera information with the IVA applications through Milestone AI Bridge:

- Name and description
- GPS location and field of view
- Current communication status (camera online / offline)
- Available streams and stream properties (among others: codec, resolution, and framerate)
- The url to the actual stream, and the actual video stream (from a selection of protocols, including WebRTC and RTSP)

From IVA applications to the XProtect VMS

Through Milestone AI Bridge, the IVA applications send the analytics data[2] back to the XProtect VMS as events, metadata, or video.

[1]A software program that analyzes video for objects and the behavior of objects.
[2]The result of an analytics topic's analysis of a video stream. You can run multiple analytics topics on the same video.

Communication and ports in the full Milestone AI Bridge integration

Ports used by the XProtect VMS

Port number	Protocol	Owner	Purpose
80	SOAP	The Management Server service	Configuration. The purpose of port 80 and 443 is the same. • Unsecured communication: use port 80. • Secured communication: use port 443.
443	SOAP		
22331	SOAP	The Event Server service	Events.
7563	IS	The Recording Server service	Video.
	SOAP	The Recording Server service	Status API.

Ports used by Milestone AI Bridge

The port numbers vary depending on whether the processing servers use Kubernetes or Docker Compose.

Docker Compose

Port number	Protocol	Owner	Purpose
3500	GraphQL	Milestone AI Bridge Health container	Health check of processing servers.
8787	VPS	Milestone AI Bridge Proxy container	Video and metadata.

Kubernetes

Port number	Protocol	Owner	Purpose
80	SOAP and API Rest	NGINX controller	Health check of processing servers. Video and metadata. The purpose of port 80 and port 443 is the same. • Unsecured communication: use port 80. • Secured communication: use port 443.
443	SOAP and API Rest		

Ports used by the IVA applications

When an IVA application registers itself on Milestone AI Bridge, the IVA application often also provides an URL to web page displayed in XProtect Management Client. For information on which other ports are used by an IVA application to communicate with Milestone AI Bridge, consult the documentation for the IVA application.

Port number	Protocol	Owner	Purpose
Individual for each IVA application	HTTP or HTTPS	IVA application	Web page made available in XProtect Management Client where you can draw the graphical representation of the selected analytics topic on top of the video from the camera. Requires Microsoft Edge WebView2.

Communication from Milestone AI Bridge to the IVA application

Milestone AI Bridge API's are available through different ports (including 2181, 9092, 3030, 4000, 4001, 8554, 8555, 9898, 8382, and 8383.)

These ports must not be occupied by other applications, otherwise the Milestone AI Bridge will not function properly.

The Milestone AI Bridge API

The Milestone AI Bridge API exposes an API through GraphQL and can be accessed from other containers within the cluster using the endpoint http://aib-aibridge-webservice:4000/api/bridge/graphql.

All containers of the IVA application will use this endpoint when communicating through Milestone AI Bridge.

GraphQL

The Query service is the main entry point of Milestone AI Bridge and the API of this service is made with GraphQL.

GraphQL is an open-source query language for APIs as well as a query runtime engine and can utilize GraphiQL. GraphQL enables you to request exactly what you need in one request, avoiding the issues associated with retrieving too must or too litle data.

The GraphQL API is available at the URL http://<kubernetes-cluster-hostname>:4000/api/bridge/graphql where <kubernetes-cluster-hostname> is your Kubernetes cluster's external hostname. You can use the GraphiQL IDE to create and structure your GraphQL queries.

GraphiQL

GraphiQL is a browser-based user interface that can be used for editing, testing and executing GraphQL queries and mutations against a GraphQL API. GraphiQL enables you to correctly structure your GraphQL queries.

Additionally, you can use the GraphiQL to experiment and build your queries using live data from a real VMS and is a good resource for exploring and learning the API.

GraphiQL enables you to access the API's documentation directly and includes syntax highlighting, intellisense, auto-completion as well as automatic documentation.

GraphiQL is enabled when running in debug mode.

Milestone AI Bridge containers

Milestone AI Bridge consists of multiple container images with each image used for specific functions within the entire solution. The container images are a part of the deployment of Milestone AI Bridge but are also available for individual download directly from the NGC portal.

To facilitate deployment of all these containers, you can use the Milestone AI Bridge helm chart to install the containers on multiple processing servers using Kubernetes or install them on a single processing server using Docker Compose by running the docker-compose-production.yml file.

If necessary, the default settings of the .yaml file inside the Milestone AI Bridge helm chart or the .yaml file for deployment on a single server using Docker Compose can be edited, enabling you to fine-tune the deployment of Milestone AI Bridge on your processing server.

Milestone AI Bridge has an event driven architecture and much of the communication between the containers takes place through brokers and topics provided by Apache Kafka.

The container images are located at https://ngc.nvidia.com/containers (Requires NGC account).

Name	Role
AI Bridge Streaming (aibridge-streaming)	This container grants IVA applications access to video streams from the XProtect VMS using the RTSP protocol and a gRPC based protocol named Direct Streaming. Direct Streaming enables access to live and recorded video. Use the aibridge-webservice container to query which protocol to use as well as the protocol specific endpoints.
AI Bridge Kafka zookeeper (aibridge-kafka-zookeeper)	This container runs a Kafka Zookeeper instance that keeps track of all the brokers and Apache Kafka topics. The Docker image for this container is also available here: https://hub.docker.com/r/confluentinc/cp-zookeeper/.
AI Bridge Kafka broker (aibridge-kafka-broke)	This container runs a Kafka Broker instance hosting the Apache Kafka topics. The Docker image for this container is also available here: https://hub.docker.com/r/confluentinc/cp-kafka/.
AI Bridge Fuseki (aibridge-fuseki)	This container runs the Apache Jena Fuseki SPARQL server using TBD for a RDF storage database. The database replicates parts of the XProtect VMS configuration so the GraphQL interface exposed by the aibridge-webservice container can be queried in a database. The database only functions as a cache and no data needs to be persisted. The database is only populated when Milestone AI Bridge is initialized. Changes made to the XProtect VMS configuration, for example if a camera name has been changed or the Field of View has been enabled or disabled, will be reflected in the database as well.
AI Bridge Init (aibridge-init)	This container initializes the Milestone AI Bridge and registers it as a service in XProtect VMS. After the registration, the processing servers and the IVA applications are available for configuration and subscription in XProtect Management Client and the container will be stopped.
AI Bridge	This container connects to and receives data from the XProtect VMS.

Connector (aibridge-connector)	The type of data will determine the protocol used to retrieve the data. The container handles sharing of the configuration data, status of devices, and video streams from the XProtect VMS to Milestone AI Bridge.
AI Bridge Webservice (aibridge-webservice)	This container is the main entry point for the IVA application and Milestone AI Bridge interaction. This container exposes the GraphQL interface so camera details in the XProtect VMS can be queried by the IVA application. The container enables the IVA application to: • query the availability of protocol specific endpoints (e.g. RTSP) for getting video and other data from the XProtect VMS. • query the availability of event, metadata, and video topics that the IVA application can use to send generated data back into the XProtect VMS.
AI Bridge Health (aibridge-health)	This container exposes an API that enables the administrator of XProtect Management Client to monitor the health of the processing servers and the connectivity between the XProtect VMS and Milestone AI Bridge.
AI Bridge Broker (aibridge-broker)	This container acts as a broker to which an IVA application can send events, metadata, and video. The data is sent to specific topics, which another container (aibridge-proxy) will subscribe to in order to forward data back into the XProtect VMS. The broker supports both a REST and gRPC API for sending data to the topics.
AI Bridge Proxy (aibridge-proxy)	This container sends events, metadata, and video back into the XProtect VMS. In the XProtect VMS, events can be used by the built-in rule engine to trigger actions. You can trigger recording of videos when an event occurs. You can also trigger an alarm which an operator then can manage through the built-in Alarm Manager. Frame-based metadata can also be sent back into the XProtect VMS and stored. For example, a detected object can be highlighted on the video, not only for live video, but also for recorded video. The built-in search functionality can also use the metadata to locate relevant videos using different search criteria.

What's new

2.0

- This is the first version of this Milestone AI Bridge Integrators manual.

Licensing

Licensing

Milestone AI Bridge is a free MIP SDK and requires no separate XProtect base license. An End User License Agreement (EULA) is included with the Milestone AI Bridge product and must be read and accepted.

By downloading and using the Milestone AI Bridge functionality, resources, helm chart or containers, you are accepting the terms and conditions of the license.

You must have an already running XProtect VMS system with a base license for a XProtect VMS product version 2022 R1 or later.

If you subscribe to video analytics topics that are designed to send modified video back to your XProtect VMS, you must, however, purchase one XProtect device license per video stream to receive data or video from Milestone AI Bridge.

Your IVA applications may require one or more licenses. How you purchase these licenses depends on the IVA application manufacturer. Contact your IVA app integrator.

License activation

If you receive video streams from your IVA applications, each video stream uses a device license. As usual, changes in the use of device licenses require license activation.

Some manufacturers of IVA applications have chosen to let you activate licenses to their IVA applications through your Milestone SLC. If this is the case, you activate your IVA applications like any XProtect license.

See also the section about how to activate licenses in the administrator manual for your XProtect VMS.

If a manufacturer of one or more IVA applications has chosen to have their IVA applications activated another way, contact your IVA app integrator.

Requirements and considerations

General requirements

The Milestone AI Bridge is designed to run in a containerized environment using Ubuntu Linux. However, you can run Milestone AI Bridge on any other AMD64 (x86_64) / ARM64 -based Linux system.

Cloud vs. on-premise environments

Although the Milestone AI Bridge architecture is prepared for cloud environments, the current combination of configuration options and software capabilities mainly targets on-premise solutions that are not necessarily fully open to internet-based cloud solutions.

Processing server hardware

Milestone AI Bridge runs on linux-based docker images, as do the majority of IVA applications, and must therefore be run on machines with a Linux operating system.

You can run Milestone AI Bridge on any other AMD64 (x86_64) / ARM64 -based Linux system and Milestone AI Bridge has specifically been tested to run on the NVIDIA Jetson Xavier and Orin devices.

Any machine that fulfills the Ubuntu Server 22.04.1 Long-term support (LTS) hardware requirements can be used as a processing server, but as is usual with large, data-based, computational analysis, the better the server specifications, the faster and more reliable the results.

Milestone AI Bridge also utilizes one or more NVIDIA GPU cards for video analysis and any machines dedicated to running Intelligent Video Analysis must be correctly configured with one or more NVIDIA GPU cards.

The NVIDIA EGX platform

The NVIDIA EGX platform is a Linux system prepared for Kubernetes use, that can be ordered directly on the NVIDIA NGC portal and can be delivered with the entire NVIDIA EGX software stack pre-installed.

You can also install the NVIDIA Cloud Native Stack (Previously the NVIDIA EGX software stack) on your own AMD64(x86_64)/ARM64 Linux system, for example Canonical's Ubuntu 22.04.1 LTS server if ordering the NVIDIA EGX platform is not a possibility.

The NVIDIA Cloud Native Stack is available on Github here

When you have received the EGX machine with the NVIDIA EGX Software stack, you can start installing the required components and applications on your EGX machine as well as on the XProtect Management Client machine.

When the required components and applications are installed, you must configure the Milestone AI Bridge to integrate with your XProtect VMS installation.

Processing server software

The following must be considered when planning the installation and deployment of your processing server:

Linux distribution

While any Linux distribution can in theory be used, it is recommended to use Ubuntu Server 22.04.1 Long Term Service (LTS) or later. Milestone recommends using Ubuntu 22.04.1 LTS server and this documentation assumes the use of a Ubuntu 22.04.1 LTS server or later.

Hosting Linux on a Hyper-V virtual machine

If you want to run the Processing server on a Hyper-V virtual machine hosted by a Windows Server operating system, the **Discrete Device Assignment** feature is required. This feature is available on Windows Server 2019 or later.

Virtualization and GPU passthrough must be enabled in the BIOS of the host machine and the following BIOS settings must be enabled on the Windows server host machine:

- Intel VT for Directed I/O (VT-D)
- Trusted Execution Technology (TXT)
- ASPM (Active State Power Management) or PCI Express Native Power Management (to increase system performance)
- Single root I/O virtualization (SR-IOV)

Secure Boot may prevent NVIDIA GPU drivers from running and it is recommend you disable Secure Boot in the BIOS all host machines.

XProtect VMS

Milestone AI Bridge is compatible with XProtect 2022 R1 and later. XProtect VMS versions 2022 R1 and 2022 R2 require that the administrators of the XProtect VMS install version-specific patches.

See Milestone AI Bridge support matrix on page 27

Containers

The processing server must be able to run Linux Containers. This is typically obtained by installing a Linux server with Docker Compose or Kubernetes.

Docker Compose

You can deploy Milestone AI Bridge on any system supporting Docker Compose on Ubuntu Linux, or any other Linux distribution. If you select to deploy Milestone AI Bridge on a non-Ubuntu Linux distribution, it is advised to thoroughly test the system prior adding using the system for normal operations.

Docker Compose is usually used for a single processing server.

While Milestone AI Bridge is compatible with all versions of Docker Compose version, Milestone AI Bridge has been tested using Docker version 27.3.1.

Kubernetes

When using Kubernetes, a helm chart can be employed for faster installation and deployment of Milestone AI Bridge and IVA applications. You can use Kubernetes to deploy Milestone AI Bridge on any Linux machine that meets the processing server requirements.

Kubernetes can be used for managing multiple processing servers.

While Milestone AI Bridge is compatible with all versions of Kubernetes, Milestone AI Bridge has been tested using Kubernetes version 1.31.1.

Prepare XProtect for Milestone AI Bridge integration

Preparing your XProtect VMS

For Milestone XProtect 2022R3 or newer

If you want to use the newest version of Milestone AI Bridge and you are running Milestone XProtect 2022R3 or newer, you must install the Milestone XProtect Processing Server Admin Plugin for the XProtect Management Client.

For Milestone XProtect 2022R2 or XProtect 2022R1

If you want to use the newest version of Milestone AI Bridge and you are running XProtect 2022R2 or XProtect 2022R1, you must first update your XProtect installation with the appropriate Milestone AI Bridge patches and then install the Milestone XProtect Processing Server Admin Plugin.

For Milestone XProtect 2021R2 and older

Milestone AI Bridge does not support XProtect 2021R2 and older.

For more information, see Milestone AI Bridge support matrix on page 27

Install the Milestone XProtect Processing Server Admin Plugin

Install the Milestone XProtect Processing Server Admin Plugin to integrate your XProtect installation with the newest version of Milestone AI Bridge.

When the Milestone XProtect Processing Server Admin Plugin has been installed, the Milestone AI Bridge functionality will displayed in the **Processing Servers** node in the XProtect Management Client. You can afterwards set up and configure the registered IVA applications from within your XProtect installation.

Install and apply the Milestone XProtect Processing Server Admin Plugin

1. Log on to the NIVIDIA NGC platform, navigate to Private Registry > Resources and click Milestone AI Bridge XProtect plug-in.
2. Click Download to download the files.zip file. The files.zip file contains the newest version of the VideoOS.ProcessingServer.Plugin.Admin.Installer.exe file.
3. Extract the VideoOS.ProcessingServer.Plugin.Admin.Installer.exe file from the files.zip (or unpack the files.zip file and copy VideoOS.ProcessingServer.Plugin.Admin.Installer.exe file)
4. Place the VideoOS.ProcessingServer.Plugin.Admin.Installer.exe file on the machine that contains the XProtect Management Client.
5. Run the VideoOS.ProcessingServer.Plugin.Admin.Installer.exe file with administrator privileges on the

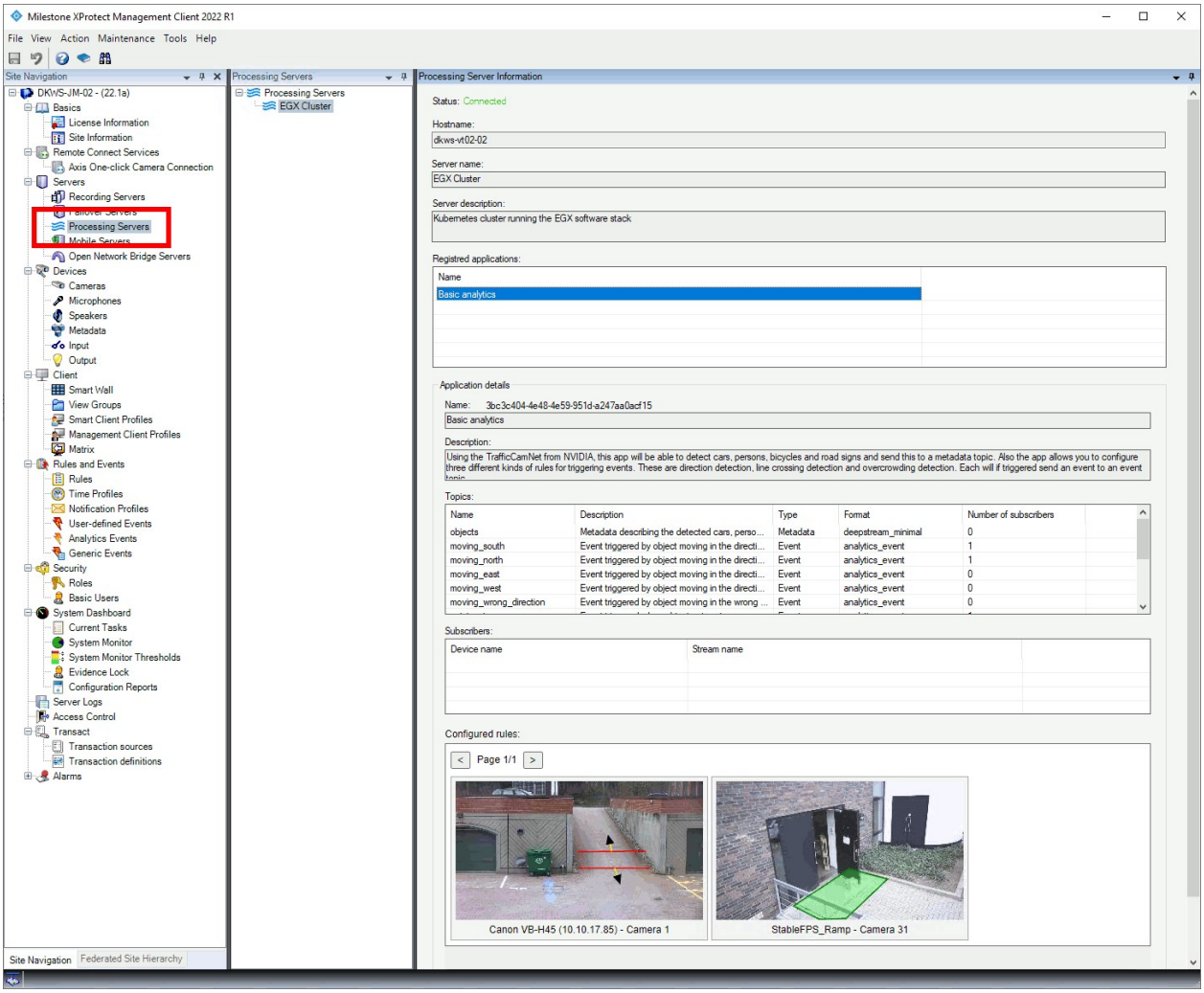
machine that contains the XProtect Management Client and follow the installation instructions.

6. Start or restart the XProtect Management Client to finalize the installation. In **Management Client** > **Site Navigation** pane > **Servers**, a new **Processing Servers** node will be displayed.

> If you want an older version of the VideoOS.ProcessingServer.Plugin.Admin.Installer.exe file, click Private Registry > Resources > Milestone AI Bridge XProtect plug-in to open the Overview tab of the XProtect plug-in page and then click the File Browser tab.
>
> Select the Milestone AI Bridge version you want and click the VideoOS.ProcessingServer.Plugin.Admin.Installer.exe to start your download.

Example of the Processing Servers node

In this example, there is one registered Milestone AI Bridge processing server called EGX Cluster, with one registered IVA application.



The EGX Cluster processing server is running the Basic analytics IVA application and the Basic analytics application has registered a number of analytics topics. An analytics topic is a named feed that the analytics application can send data to:

- objects
- moving_south
- moving_north
- moving_east
- moving_west
- moving_wrong_direction
- etc

The moving_ topics are event topics which can receive event data from the Basic analytics application if an object is detected to be moving in a certain direction (South, North, East, West or in the wrong direction).

The objects topic is a metadata topic.

The rules for triggering events are visually displayed in the Configured rules group at the bottom of the page. Here two rules for triggering are displayed:

- Camera 1: arriving_to_area or leaving_from_area
- Camera 31: zone_has_unexpected_activity

Update your Milestone XProtect installation

If you are running XProtect 2022R1 or 2022R2, you must update your XProtect installation before you can install the Milestone AI Bridge functionality.

To update your XProtect installation, you must replace the **VideoOS.Administration.AddIn.dll** and **VideoOS.Administration.Client.dll** files with the corresponding files from the patch file. The files are found in the XProtect Management Client folder, located in the installation folder of XProtect on the XProtect Management Client machine.

The default XProtect installation path is C:\Program Files\Milestone but the installation path of your XProtect product may be different.

Since the existing **VideoOS.Administration.AddIn.dll** and **VideoOS.Administration.Client.dll** files will be replaced with newer versions, it is recommended to make a back-up copy of these files in case you have to restore them.

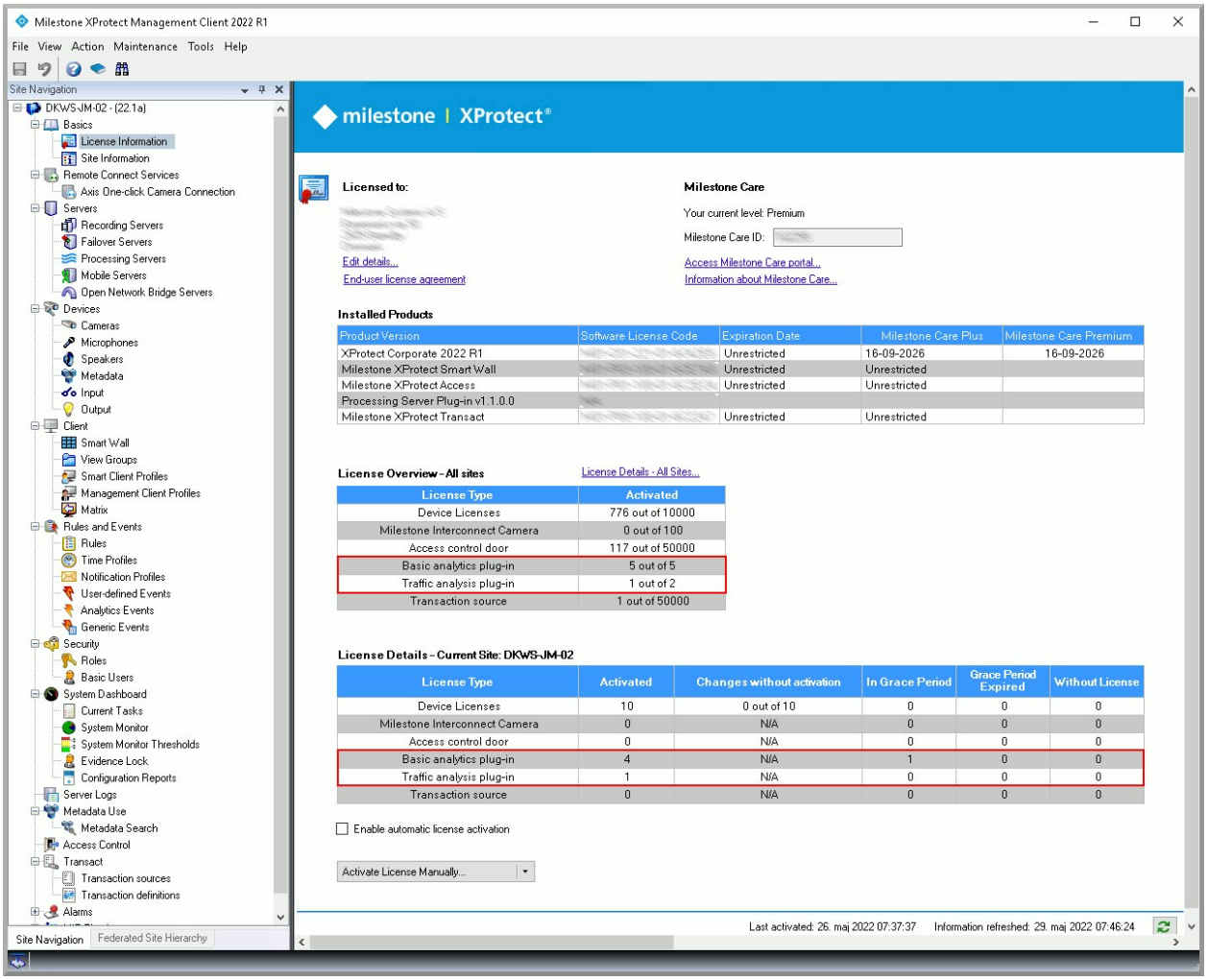
IVA license activation

The Milestone AI Bridge enables you to license your own IVA integration through the Milestone License Server.

This allows you to issue and manage licenses to customers through MyMilestone and the MIP License Management Tool. However, for the license activation to work, you must first apply the Milestone AI Bridge patch to all XProtect versions older than XProtect 2022 R2

From XProtect 2022R3 and later, the updates contained in the patch are part of the product by default.

When the patch is successfully applied, your own Video Analytics Apps will be displayed on the **License page** in the XProtect Management Client.



If you don't use the XProtect Management Client License Activation feature, you will not need to apply the Milestone AI Bridge patch.

Download the patch files

Log on to the NIVIDIA NGC platform, navigate to **Private Registry** > **Resources** and click Milestone AI Bridge XProtect patch to open the Milestone AI Bridge XProtect patch page.

1. On the Milestone AI Bridge XProtect patch page, click the **File Browser** tab.
2. On the **File Browser** tab, select the XProtect version you want to update and click the aibridge_xprotect_patch.zip file to download the file.
3. Extract the contents of the aibridge_xprotect_patch.zip file and close your XProtect Management Client.
4. Copy the **VideoOS.Administration.AddIn.dll** and **VideoOS.Administration.Client.dll** files, located in the **mcactivation** folder of the aibridge_xprotect_patch.zip file to the XProtect Management Client folder on the Management Client machine.

   This will replace the existing **VideoOS.Administration.AddIn.dll** and **VideoOS.Administration.Client.dll** files.
5. Re-start your XProtect Management Client

When the patch has been successfully applied, your own Video Analytics Apps will be displayed on the **License** page in the XProtect Management Client.

Install the Milestone XProtect Processing Server Admin Plugin

Once you have updated your XProtect Management Client, you must install the Milestone XProtect Processing Server Admin Plugin.

See Install the Milestone XProtect Processing Server Admin Plugin on page 23

Milestone AI Bridge support matrix

Milestone AI Bridge 1.5 and newer utilizes OAuth integration and is only supported by Milestone XProtect 2022R1 or newer.

Milestone AI Bridge 1.5 and newer - Supported XProtect versions

- XProtect 2022R3 and newer: Install the Milestone XProtect Processing Server Admin Plugin for the XProtect Management Client.
- XProtect 2022R2: Update your XProtect installation with the Milestone AI Bridge patch for XProtect 2022R2 and then install the Milestone XProtect Processing Server Admin Plugin for the XProtect Management Client.
- XProtect 2022R1: Update your XProtect installation with the Milestone AI Bridge patch for XProtect 2022R1 and then install the Milestone XProtect Processing Server Admin Plugin for the XProtect Management Client.
- XProtect 2021R2 and older: Unsupported.

Install the processing server

You can deploy a fully featured Milestone AI Bridge application on any system running Ubuntu 22.04.1 LTS or another Linux distro that fulfills the Ubuntu Server 22.04.1 LTS hardware requirements.

Regardless of the machine, you can use either Kubernetes or Docker Compose to install and deploy the Milestone AI Bridge application.

Kubernetes can best be used to install and deploy the Milestone AI Bridge application on one or multiple processing servers and Docker Compose can best be used to install and deploy the Milestone AI Bridge application on a stand-alone machine.

You must also install and deploy any preferred Intelligent Video Analytics (IVA) applications, as it is these IVA applications that will do the data analysis and send the results back to your XProtect VMS.

This documentation

The following describes how to deploy Milestone AI Bridge and its dependents using Kubernetes or Docker Compose. The documentation is based on Ubuntu 22.04.1 LTS but you can use the description to extrapolate the procedure for deploying Milestone AI Bridge on any other Linux-based machine using Docker Compose or Kubernetes.

Generating the server SSL certificates

Creating the directories to store the certificates

This section is only relevant if you have secured your XProtect VMS using certificates.

In order for Milestone AI Bridge certificates to be found, you must create the following folder structure inside your **certs** directory:

```
.
├── tls-ca
└── tls-server
```

You can use the following command to create the former directories:

```
mkdir -p certs/{tls-ca,tls-server}
```

Generate a server SSL certificate based on an existing Certificate Authority

If you want to generate a server SSL certificate based on an existing Certificate Authority (CA) file, you must first determine if your XProtect VMS is running a secured state.

You can then generate the server ssl certificate and private key for the sample IVA application.

If your XProtect VMS is not running in a secured state

If your XProtect VMS is not running in a secured state, your Milestone AI Bridge should not run in a secured state.

If you want create a Certificate Authority (CA) for your XProtect VMS installation anyway, see this guide: XProtect VMS certificates guide.

If your XProtect VMS is running in a secured state

If your XProtect VMS is already running in a secured state and the communication is encrypted based on a public Certificate Authority (CA), you must do the following, storing the resulting files on the machine running Milestone AI Bridge:

- Extract the vms-authority certificate file in PEM format
- Extract the server ssl certificate and private key from the PFX file

Extract the vms-authority.crt file in PEM format

1. On the machine your XProtect VMS is running on, open the **certmgr** tool and in the left pane, in the **Trusted Root Certification Authorities** > **Certificates** folder, select the VMS CA certificate.
2. Click **Action** > **All Tasks** > **Export** to start the **Certificate Export** wizard.
3. In the **Certificate Export** wizard, select the .CER export file format and select a location and for the exported certificate and make sure the certificate is named **vms-authority-public.cer**.
4. Copy the vms-authority-public.cer certificate file to the **tls-server** folder of your Linux host machine. You can use the SCP command line utility or the WINSCP file manager to do this.
5. On the machine running Milestone AI Bridge, open a terminal and use the following command to extract the vms-authority certificate file in PEM format:

   ```
   openssl x509 -inform der -in vms-authority-public.cer -outform pem -out
   vms-authority.crt
   ```

6. Copy the extracted vms-authority.crt file in PEM format to the **tls-ca** folder.

Extract the server ssl certificate and private key from the PFX file

To create server ssl certificates for the machine running Milestone AI Bridge, you must first run a script that creates a server certificate for the machine your XProtect VM is running on.

For more information, see Create server SSL certificate script from the XProtect VMS certificates guide guide.

After running the script a .PFX file will be generated, for example server.pfx file.

1. Copy the server.pfx certificate file to the **tls-server** folder of the machine running Milestone AI Bridge.

   (missing or bad snippet)

2. On the machine running Milestone AI Bridge, open a terminal and use the following commands to extract the server ssl certificate and private key from the PFX file:

   ```
   openssl pkcs12 -in server.pfx -nocerts -out server.key -nodes

   openssl pkcs12 -in server.pfx -clcerts -nokeys -out server.crt -nodes
   ```

Make sure the extracted server.key and server.crt files both are located in the **tls-server** folder.

Deploying using Kubernetes

Deploying Milestone AI Bridge (Kubernetes)

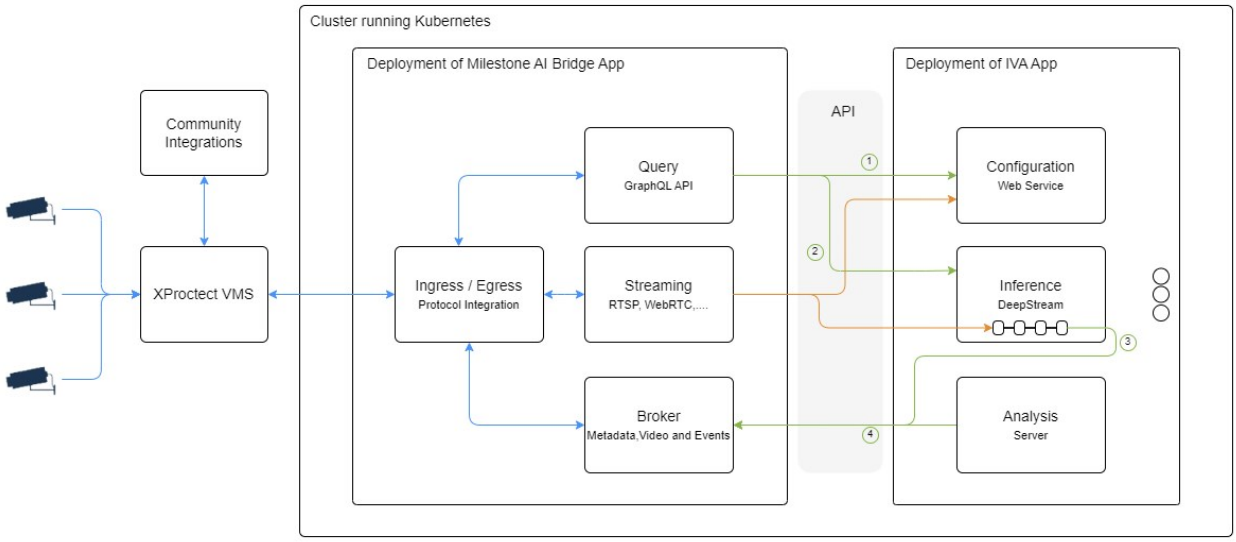
Milestone AI Bridge can be deployed, maintained and operated using Kubernetes or Docker Compose to manage the Milestone AI Bridge containers. This section of the Milestone AI Bridge documentation describes how you can use Kubernetes and the Milestone AI Bridge Helm charts to do this.

Linux and Windows

The majority of Intelligent Video Analytics (IVA) applications are designed as Linux programs and run on various Linux distributions. For this reason, the processing server will invariably utilize a Linux operating system, typically Ubuntu Linux, while the XProtect Management Client requires the Microsoft Windows operating system.

The installation and configuration instructions will therefore be different depending on the operating system.

Example of a Milestone AI Bridge installation and an IVA application



This is one example of a simple deployment of the Milestone AI Bridge in connection with the XProtect VMS and an Intelligent Video Analysis (IVA) application.

There are two deployments: the Milestone AI Bridge itself and the IVA application.

The API's that the Milestone AI Bridge expose to the IVA application are all internal to the cluster network and cannot be accessed from the outside (unless deployed in debug mode). However, all communication going into the cluster and leaving the cluster must be secured.

Milestone recommends all traffic between the Milestone AI Bridge and the XProtect VMS is encrypted using TLS encryption.

For more information, see Securing the Milestone AI Bridge connection (Kubernetes) on page 40

Install prerequisites

The Kubernetes application and the Helm application must both be installed on your machine.

For more information, see How to install Kubernetes client (external link) and How to install Helm charts (external link).

Configure the XProtect Management Client machine

You must configure the XProtect Management Client to communicate with the processing server through the Milestone AI Bridge.

> If you have not yet installed the Milestone XProtect Processing Server Admin Plugin on your XProtect Management Client machine, you should do so now. See Install the Milestone XProtect Processing Server Admin Plugin on page 23 for more information.

To configure your XProtect Management Client for communication with the processing server you must also create an XProtect basic user and assign the new basic user the administrator role.

See Create a basic user for Milestone AI Bridge.

Install Milestone AI Bridge

After you have created a basic user with the Administrator role in your XProtect Management Client, you can install the Milestone AI Bridge application.

To install Milestone AI Bridge, you should:

1. Make your NGC API key available to the system.
2. Fetch the Helm chart of the Milestone AI Bridge
3. Unpack the Helm chart
4. Fetch any dependencies of the Helm chart
5. Deploy the Milestone AI Bridge application

Make your NGC API key available to your system

The API Key is used to authenticate your access to the NGC container registry and enables you to access locked container images from the NGC container registry. A valid NGC API key is therefore required to install Milestone AI Bridge.

If you do not yet have an API key, you can access the NGC portal to generate a new key.

Remember to store the API key locally, as the NGC portal does not store API keys.

If you already have an API key but cannot find it and then generate a new API key, the old API key will automatically be invalidated.

To make your NGC API key available to your system

On your machine, open a terminal and run the following command to create a namespace called aibridge.

You can use your own namespace instead of aibridge but you must then replace all instances of the aibridge namespace with your own namespace in the examples.

```
kubectl create namespace aibridge
```

In the same terminal, run the following command:

```
kubectl create secret docker-registry imagepullsecret \
--docker-server=https://nvcr.io \
--docker-username='$oauthtoken' \
--docker-password=<your-api-key> \
--docker-email=<your-ngc-email> \
-n aibridge
```

where <your-api-key> is your API key from the NGC portal and <your-ngc-email> is the email you are using to access the NGC portal. If you have created your own namespace, replace the aibridge namespace with your namespace after the -n parameter.

Fetch and install the Helm chart of the Milestone AI Bridge

The Helm chart is used to facilitate the installation of Milestone AI Bridge including potential dependencies as well as manage the Kubernetes YAML files used to configure your Milestone AI Bridge installation.

To fetch the Helm chart

On your machine, open a terminal and run the following commands:

```
helm fetch https://helm.ngc.nvidia.com/isv-milestone/partners/charts/aibridge-
2.0.0.tgz \
--username='$oauthtoken' \
--password=<your-api-key>
```

where aibridge-2.0.0.tgz is the tar file of the Helm chart for the version of Milestone AI Bridge version you want to install (n this case 2.0.0) and <your-api-key> is your NGC API key from the NGC portal.

When the command is executed successfully, the aibridge-2.0.0.tgz tar file will be located in your local folder.

Unpack the Helm chart

You can now unpack Helm chart in the aibridge-2.0.0.tgz file located in your local folder.

On your machine, open a terminal and run the following command:

```
tar -zxvf aibridge-2.0.0.tgz
```

The Helm chart will be unpacked and a new folder named **aibridge** created. The **aibridge** folder will be used by the commands in the following steps.

Fetch any dependencies of the Helm chart

The Helm chart contains a collection of files that are used as resources for Milestone AI Bridge application as well as the deployment files for the Milestone AI Bridge application itself.

On your machine, open a terminal, navigate to the **aibridge** folder and run following command to fetch any dependencies of the Helm chart:

```
helm dependency build .
```

When the command is run, the following output will be displayed in the terminal:

```
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "ingress-nginx" chart repository
...Successfully got an update from the "nvidia" chart repository
Update Complete. ⎈Happy Helming!⎈
Saving 1 charts
Downloading ingress-nginx from repo https://kubernetes.github.io/ingress-nginx
Deleting outdated charts
```

Deploy the Milestone AI Bridge application

Deploy Milestone AI Bridge using values.yaml settings

Once all dependencies of the Helm chart have been fetched, you must determine how the Milestone AI Bridge is to connect to your XProtect VMS and if your XProtect VMSis running in a secured (using https) or unsecured (using http) state .

If you are connecting to a VMS running in a secured state

If you are connecting Milestone AI Bridge to a VMS running in a secured state, you must first configure Kubernetes to connect securely by doing the following:

1. Create a Kubernetes configmap object
2. Create the Kubernetes secrets
3. Edit the values.yaml file
4. Deploy the Milestone AI Bridge application

Create a Kubernetes configmap object

On your machine, open a terminal and run the following command to create a Kubernetes configmap object:

```
kubectl create configmap vms-authority \
--from-file=path/to/vms-authority.crt \
-n aibridge
```

If you deploy the Milestone AI Bridge inside a namespace, you must create the Kubernetes Secret inside the same namespace. If you have created your own namespace, replace the aibridge namespace with your namespace after the -n parameter.

The file vms-authority.crt must contain the VMS CA certificate in PEM format.

All certificates must use the PEM format and must be named with the .crt file extension. For more information, see Ubuntu manual - certificates

Create the server-tls and vms-credentials Kubernetes secrets

Once the ConfigMap object has been created, run the following command to create the server-tls Kubernetes secret:

```
kubectl create secret tls server-tls \
--cert=path/to/server.crt \
--key=path/to/server.key \
-n aibridge
```

If you deploy the Milestone AI Bridge inside a namespace, you must create the Kubernetes Secret inside the same namespace. If you have created your own namespace, replace the aibridge namespace with your namespace after the -n parameter.

When the server-tls Kubernetes secret has been created, run the following command to create the vms-credentials Kubernetes secret:

```
kubectl create secret generic vms-credentials \
--from-literal='username=<username>'
--from-literal='password=<password>'
-n aibridge
```

where <username> is the new XProtect basic user and <password> is the password of the new XProtect basic user.

Edit the values.yaml file

Before deploying the the Milestone AI Bridge application, you must edit the settings in the values.yaml file to conform to your organization's characteristics, including the following settings:

- vms > url
- general > externalIP
- general > externalHostname
- ingress-nginx > controller > service > externalIPs

Other settings in the values. yaml file that may be relevant to your organization's requirements may also have to be adjusted.

For more information, see Configuring Milestone AI Bridge (Kubernetes) on page 43

Deploy the Milestone AI Bridge application

Run the following command to deploy the Milestone AI Bridge application:

```
helm install aib . -n aibridge
```

The parameters defined in the values.yaml files will automatically be used.

If you are connecting to a VMS running in an unsecured state

If you are connecting Milestone AI Bridge to a VMS running in a unsecured state, you must do the following:

1. Create a Kubernetes secret
2. Edit the values.yaml file
3. Deploy the Milestone AI Bridge application

Create a Kubernetes Secret

Create a Kubernetes Secret to help authenticate the XProtect basic user.
On your machine, open the terminal and run the following command:

```
kubectl create secret generic vms-credentials \
--from-literal='username=<username>'
--from-literal='password=<password>'
-n aibridge
```

where <username> is the new XProtect basic user and <password> is the password of the new XProtect basic user.

If you deploy the Milestone AI Bridge inside a namespace, you must create the Kubernetes Secret inside the same namespace. If you have created your own namespace, replace the aibridge namespace with your namespace after the -n parameter.

Edit the values.yaml file

Before deploying the the Milestone AI Bridge application, you must edit the settings in the values.yaml file to conform to your organization's characteristics, including the following settings:

- vms > url
- general > externalIP
- general > externalHostname
- ingress-nginx > controller > service > externalIPs

> Other settings in the values. yaml file that may be relevant to your organization's requirements may also have to be adjusted.

For more information, see Configuring Milestone AI Bridge (Kubernetes) on page 43

Deploy the Milestone AI Bridge application

Run the following command to deploy the Milestone AI Bridge application:

```
helm install aib . -n aibridge
```

The parameters defined in the values.yaml files will automatically be used.

Deploy Milestone AI Bridge using custom settings

If you want to deploy Milestone AI Bridge and set your own parameters during the deployment, navigate to the **aibridge** folder and run the following command:

```
helm install aib . -n aibridge \
--set vms.url=<url-of-xprotect-management-server> \
--set general.externalIP=<kubernetes-cluster-ip-address> \
--set general.externalHostname=<kubernetes-cluster-hostname> \
--set ingress-nginx.controller.service.externalIPs={<external-ip-address-of-cluster>}
```

where

- aib is the release name of the deployment. You can specify any name for deployment you like.
- <url-of-xprotect-management-server> is the URL of your XProtect management server.
- <kubernetes-cluster-hostname> is the hostname of your Kubernetes cluster.
- <ip-address-of-cluster> is the IP address of your Kubernetes cluster.

Defining custom settings is more cumbersome, can be more error-prone, and is not generally recommended.

Disable an NGINX controller

Milestone AI Bridge employs an ingress controller and by default the Helm chart is set up to automatically deploy an NGINX ingress controller during Milestone AI Bridge deployment.

You can disable the automatic deployment of an NGINX controller by adding the following option to the deployment terminal command:

```
--set ingress-nginx.enabled=false
```

If you disable the NGINX ingress controller, you must deploy the controller to the Kubernetes cluster manually.

Manually deploy an NGINX ingress controller to the Kubernetes cluster

To manually deploy the NGINX ingress controller to the Kubernetes cluster, run the following commands in the terminal:

```
helm repo add ingress-nginx https://kubernetes.github.io/ingress-nginx

helm repo update

kubectl create namespace nginx

helm install --namespace nginx --generate-name --set
controller.service.externalIPs={<ip-address-of-cluster>} ingress-nginx/ingress-
nginx --version 4.1.4
```

where <ip-address-of-cluster> is the IP address of your Kubernetes cluster or the IP address of the load balancer if Milestone AI Bridge is deployed on a multi-node Kubernetes cluster.

The ingress template files for Milestone AI Bridge are compatible with 'ingress-nginx' helm package version 4.1.4 but not the newer versions.

Securing the Milestone AI Bridge connection (Kubernetes)

You can employ TLS encryption to help secure the connections between your XProtect installation and Milestone AI Bridge but before you can use TLS encryption, you will first have to enable TLS encryption for all communication in XProtect.

The Milestone Server Configurator is used to enable TLS encryption and to select the server certificates.

For more information, see the https://doc.milestonesys.com/2024r1/en-US/portal/htm/chapter-page-certificates-guide.htm.

Server certificates are issued by a Certificate Authority (CA). This can be an externally trusted certificate authority, or you can act as your own certificate authority by using a self-signed CA certificate.

In the following the certificate authority is referred to as the VMS CA and the actual CA certificate in question is referred to as the VMS CA certificate.

Streaming container security considerations

For improved compliance with defined user permissions in the XProtect VMS, user oauth tokens assigned to video sent from the XProtect VMS to the IVA application must be assigned to webRTC feeds forwarded by the IVA application back into the XProtect VMS.

User oauth tokens assigned to video sent from the XProtect VMS to the IVA application can also be assigned to snapshot feeds. If you do not assign oauth tokens to snapshot feeds, the Milestone XProtect basic user defined when installing Milestone AI Bridge will be used as a token instead.

In a production environment

For production environments, IVA application developers should always set the **enforce-oauth** parameter in the **AI Bridge Streaming** (aibridge-streaming) container to **true** in the docker-compose or helm chart settings.

If the **enforce-oauth** parameter is set to **false** in a production environment, the oauth token of the Milestone XProtect basic user defined when installing the Milestone AI Bridge is used as a token. This means that snapshots or webRTC feeds from the IVA application may be available for Milestone XProtect users that otherwise do not have permission to this data.

In a test environment

For test purposes, IVA application developers can set the **enforce-oauth** parameter to **false** to facilitate testing results unless security testing is being performed.

The **enforce-oauth** parameter is located in the aibridge-streaming.yaml file.

Create a Kubernetes ConfigMap object

If your XProtect installation is not running in a secured state (running over https), the following steps are optional.

To register the VMS CA certificate as trusted by the Milestone AI Bridge, you must create a Kubernetes ConfigMap object by opening a terminal and running the following command:

```
kubectl create configmap vms-authority \
--from-file=path/to/vms-authority.crt \
-n aibridge
```

If you deploy the Milestone AI Bridge inside a namespace, you must create the Kubernetes Secret inside the same namespace. If you have created your own namespace, replace the aibridge namespace with your namespace after the -n parameter.

The file vms-authority.crt must contain the VMS CA certificate in PEM format.

All certificates must use the PEM format and must be named with the .crt file extension. For more information, see Ubuntu manual - certificates

Assign server certificate to Milestone AI Bridge

If your XProtect installation is not running in a secured state (running over https), the following steps are optional.

Milestone AI Bridge itself also acts as a server towards your XProtect installation and thus must have a server certificate issued for it by the VMS CA.

This server certificate and its associated private key must be added as a Kubernetes Secret object by opening a terminal and running the following command:

```
kubectl create secret tls server-tls \
--cert=path/to/server.crt \
--key=path/to/server.key \
-n aibridge
```

where <path> is the path to the server.crt and server.key files respectively.

If you have created your own namespace, replace the aibridge namespace with your namespace after the -n parameter.

Here, server.crt and server.key are the issued server certificate and its associated private key respectively, both in PEM format and with the .crt file name extension.

You deploy AI Milestone AI Bridge in a namespace, then the secret object must also be created in the same namespace.

You can now use TLS encryption for all connections between your XProtect installation and the Milestone AI Bridge by using the HTTPS scheme in the URL of the XProtect management server, see the example below.

Example of terminal command

You must be in the **aibridge** folder.

```
helm install aib . -n aibridge \

--set vms.url=https://my-management-server \

--set general.externalIP= <kubernetes-cluster-ip-address> \

--set general.externalHostname= <kubernetes-cluster-hostname>\

--set ingress-nginx.controller.service.externalIPs={<kubernetes-cluster-ip-address> }
```

Configuring Milestone AI Bridge (Kubernetes)

After you have installed Milestone AI Bridge and its required resources, you must configure the Milestone AI Bridge to integrate both with your XProtect VMS and with your IVA application.

Default configuration settings

The default settings of the Milestone AI Bridge are specified in the values.yaml file inside the Helm chart.

The values.yaml file

The values.yaml file in the Helm chart contains default settings of the Milestone AI Bridge.

These settings can be overridden on the command line when installing the Helm chart by using the -- set option or you can edit the settings in the values.yaml file in the Helm chart directly.

The contents of a sample values.yaml file

```
vms:

 url: "http://my-management-server"
```

```
# Define these variables if your vms is not in the network domain
# ip: "<my-management-server-ip>"
# hostname: "<my-management-server-hostname>"
bridge:
  id: "12355b21-5a25-4a1d-b6d2-f6e02c9b95b4"
  name: "EGX Cluster"
  description: "Kubernetes cluster running the EGX software stack"
webpage: ""
gateway:
  id: "1b80eaa0-203d-4dc0-ae3b-9bf4b85ec992"
  version: "1.0.0"
replicas:
  health: 1
  connector: 1
  streaming: 1
  broker: 1
  proxy: 1
  webservice: 1
general:
  tag: v1.5
  debug: false
  externalIP: "10.10.16.34"
  externalHostname: "Kubernetes-cluster-ip-address" # In a multi-node cluster,
the externalHostname must be given a hostname that does not exist in the system.
This fake hostname must then be resolved to the IP address of the load balancer
in the DNS or in the host machine's configuration file.
  masterKey: "encryption key example"
externalRootPath: "/processing-server" # Path used to segmentate endpoints
exposed outside of the AI Bridge running cluster.
gpuEnabled: false
kafka:
```

```
  logRetentionMs: 300000 # AI Bridge's Kafka topics retention time in ms
ingress-nginx:
  enabled: true
  controller:
    service:
      externalIPs:
      - "<kubernetes-cluster-ip-address>" # In a multi-node cluster, it must be
the ip address of the load balancer.
      annotations: # Define this variable only in a multi-node cluster setup
with metallb loadbalancer.
        metallb.universe.tf/loadBalancerIPs: "<kubernetes-cluster-ip-address>" #
Defines the ip address of the load balancer.
```

The vms section

Parameter	Description
url	Displays the URL of the XProtect management server.
ip	The IP address of your VMS machine.
hostname	The hostname of your VMS machine.

If you use a separate VMS to test your Milestone AI Bridge solutions, your test VMS can placed in the network domain or outside the network domain.

If your test VMS is placed in the network domain, Milestone AI Bridge supports domain networks where the hostnames of the involved machines can be resolved by pointing to the DNS.

If your test VMS is placed outside the network domain, you can enable Milestone AI Bridge to resolve your VMS hostname by adapting the VMS network configurations in the values.yaml file for Kubernetes installations or the .env file for Docker-Compose installations.

If you are using Kubernetes, you can set the **ip** and **hostname** variables in the values.yaml file with the IP address and hostnme of your test VMS.

The ip and hostname variables are contained in the values.yaml but are not active. Remove the # comment marker to activate them.

The bridge section

Parameter	Description
id	The unique identifier of the Milestone AI Bridge. The id value identifies Milestone AI Bridge when connecting to the XProtect VMS. Unless you want to run multiple AI bridges, you should not change this value. If you register multiple AI bridges in the same VMS, each VMS must be assigned a different ID.
name	Displays the name of the Milestone AI Bridge as it appears in the XProtect Management Client
description	Displays the description of the Milestone AI Bridge as it appears in the XProtect Management Client.

The replicas section

The replicas section contains parameters that enable you to scale the number of pods running for each micro service in the cluster. By default just one pod of each service is run.

If a bottleneck occurs as the workload of the Milestone AI Bridge is increased, you can scale the Milestone AI Bridge to overcome this bottleneck by adjusting these numbers.

This is mostly relevant if you are running a cluster with more than one node.

The general section

Parameter	Description
externalHostname	The external facing hostname of the cluster running the Milestone AI Bridge. In a multi-node cluster, the externalHostname must be given a hostname that does not exist in the system. This fake hostname must then be resolved to the IP address of the load balancer in the DNS or in the host machine's configuration file.
debug	Enables or disables running the Milestone AI Bridge in debug mode.

Parameter	Description
	The default value is false. Set the parameter to true to run Milestone AI Bridge in debug mode. When running in debug mode, your IVA application will run outside the cluster, for example on a developer machine which makes testing and additional debugging easier. In debug mode, all API's of the Milestone AI Bridge will be exposed to the external network directly through the IP address specified in the externalIP parameter. The API's will be available through different ports, including 2181, 9092, 3030, 4000, 4001, 8554, 8555, 9898, 8382 and 8383. These port must not be occupied by other applications or the Milestone AI Bridge will not function as expected. ⚠ For a production environment, the debug parameter should always be set to false.
externalIP	The IP address of the Milestone AI Bridge when running in debug mode.
masterKey	Used to encrypt the XProtect VMS basic user credentials. For security reasons, you should encrypt the credentials of the Milestone XProtect basic user that is used by the Milestone AI Bridge to log in to the XProtect VMS. If you enter a value for the masterKey parameter directly in the values.yaml file, the credentials will be encrypted at rest. You can define any value to the masterKey parameter any value as there no set requirements for the number or types of characters. Additionally, you can define a new masterKey parameter value if you forget the current one.
gpuEnabled	Used to ensure the aibridge-streaming pod is deployed in a node that contains an Nvidia GPU. If set to true, at least one node must have an Nvidia GPU installed for the aibridge-streaming pod to be deployed. The default value is false.

Parameter	Description
	The Nvidia GPU Operator must be installed on the control plane node. For more information about the GPU Operator, see Installing the NVIDIA GPU Operator (External link).

The ingress-nginx section

The Milestone AI Bridge employs an ingress controller and by default the Helm chart is set up to automatically deploy an NGINX ingress controller during Milestone AI Bridge deployment.

Parameter	Description
ingress-nginx.enabled	Enable or disable the ingress controller for the Milestone AI Bridge. You must configure the controller with the external IP address of the cluster. The controller will only accept incoming network requests sent to this address. If you already have an ingress controller running, you can disable the dependency by setting ingress-nginx.enabled parameter to false.
externalIPs	The external IP address of your Kubernetes cluster or the IP address of the load balancer if Milestone AI Bridge is deployed on a multi-node Kubernetes cluster.

The MetalLB Loadbalancer

When configuring a multi-node Kubernetes cluster, you can use MetalLB as a load-balancer provider if you have installed it during the initial set up and configuration of your multi-node Kubernetes cluster. MetalLB is one of many other potential load-balancer providers that you can use and has only been used by Milestone as a proof-of-concept.

You are of course free to select other load-balancer providers. If you select other load-balancers, you must adapt your helm charts to configure your selected load balancer provider.

The MetalLB configuration is located in the **values.yaml** file under **ingress-nginx.controller.service.annotations**.

The **metallb.universe.tf/loadBalancerIPs** variable must be defined as the specific IP address of the MetalLB load balancer.

The MetalLB documentation defines this IP address as optional, but for Milestone AI Bridge, it is mandatory as the XProtect VMS connects to the Milestone AI Bridge using the known values defined in the **general.externalHostname** and **general.externalIP** variables.

The same IP address defined for the **metallb.universe.tf/loadBalancerIPs** must also be defined for the **general.externalIP** variable.

The hostname that you want to use for Milestone AI Bridge services, must be set using a hostname that does not exist in the system, effectively creating a fake hostname.

DNS or local machines

If you are using a Domain Name System (DNS), you must configure the DNS to resolve the fake hostname to the value of the **metallb.universe.tf/loadBalancerIPs** (ie. the IP address of the load balancer)

If you are not using a DNS, you must configure the XProtect VMS windows machine hosting the management server (and all other machines that host the Management Client) to resolve the fake hostname to the value of the **metallb.universe.tf/loadBalancerIPs** (ie. the IP address of the load balancer).

For more information, see MetalLB Concepts and MetalLB Usage. (external links)

Verifying Milestone AI Bridge is running (Kubernetes)

After deploying the Milestone AI Bridge, you can verify all pods are running as expected by opening a terminal and running the following command

```
kubectl get pods -n aibridge
```

If you have created your own namespace, replace the aibridge namespace with your namespace after the -n parameter.

The output of the command will resemble the example below.

Your installation may contain additional pods running in your cluster, but the ones displayed below should be listed.

POD	READY	STATUS	RESTARTS	AGE
aib-aibridge-broker-ccc86479-676mc	1/1	Running	0	13m

POD	READY	STATUS	RESTARTS	AGE
aib-aibridge-connector-8c5b9dbf7-jdjcd	1/1	Running	0	13m
aib-aibridge-fuseki-dbb789678-s5nv5	1/1	Running	0	13m
aib-aibridge-health-58bf7fbcb7-4c8xj	1/1	Running	0	13m
aib-aibridge-kafka-broker-77cd764b4f-fh4ms	1/1	Running	0	13m
aib-aibridge-kafka-zookeeper-876bfdc66-jmt4s	1/1	Running	0	13m
aib-aibridge-proxy-7bd9d9d59-9hpdj	1/1	Running	0	13m
aib-aibridge-streaming-8d75885d9-qqf9c	1/1	Running	0	13m
aib-aibridge-webservice-564d7dbc68-cwfrv	1/1	Running	0	13m

If you see aibridge-init-xxxxx pod running, the Milestone AI Bridge is still initializing.

If the aibridge-init-xxxxx pod does not complete within a couple of minutes, you can check the log file of the pod by opening a terminal and running the following command:

```
kubectl logs aib-aibridge-init-<xxxxx> -n aibridge
```

where <xxxxx> in the pod name will be different for every deployment.

Running in debug mode (Kubernetes)

Milestone recommends you run the integration in debug mode before deploying to a live production environment to test the integrations and connections. You can also run the integration in debug mode to troubleshoot issues and double-check any changes made before deploying them to a live environment.

When running in debug mode, your IVA application will run outside the cluster, for example on a developer machine which makes testing and additional debugging easier.

In debug mode, all API's of the Milestone AI Bridge will be exposed to the external network directly through the IP address specified in the externalIP parameter. The API's will be available through different ports, including 2181, 9092, 3030, 4000, 4001, 8554, 8555, 9898, 8382 and 8383. These port must not be occupied by other applications or the Milestone AI Bridge will not function as expected.

For a production environment, the debug parameter should always be set to false.

To deploy Milestone AI Bridge in debug mode, add or edit the following two options to the deployment terminal command:

```
--set general.debug=true \
--set general.externalIP=<external-ip-address-of-cluster>
```

where <external-ip-address-of-cluster> is the external IP address of your Kubernetes cluster.

Deploying using Docker-Compose

Installing Milestone AI Bridge (Docker Compose)

Milestone AI Bridge can be deployed, maintained and operated using Kubernetes or Docker Compose to manage the Milestone AI Bridge containers. This section of the Milestone AI Bridge documentation describes how you can use Docker Compose to do this.

Linux and Windows

The majority of Intelligent Video Analytics (IVA) applications are designed as Linux programs and run on various Linux distributions. For this reason, the processing server will invariably utilize a Linux operating system, typically Ubuntu Linux, while the XProtect Management Client requires the Microsoft Windows operating system.

The installation and configuration instructions will therefore be different depending on the operating system.

Log in to the NGC portal

You must log in to the NGC portal using your NGC API key if you want to be able to access the Milestone AI Bridge components in the NGC container registry.

To log in to the NGC container registry

On your machine, open a terminal and run the following command:

```
docker login nvcr.io
```

When prompted for your user name, run the following command:

```
$oauthtoken
```

The $oauthtoken username is a special username that indicates that you will authenticate with an API key and not a user name and password.

When prompted for your password, enter your NGC API key as shown in the following example

```
Username: $oauthtoken
Password: <your-API-key>
```

where <your-api-key> is your API key from the NGC portal.

The Docker Compose resource file

Once you have logged in to the NGC portal, you can access the Docker Compose resource files in Private registry > Resources. Click Milestone AI Bridge Compose Deployment and on the File Browser page, select the relevant Milestone AI Bridge version to download.

The zipped resource file is named **aibridge_compose_deployment.zip**.

The zipped resource file contains all the resources required to deploy Milestone AI Bridge using Docker Compose and consists of the following resource files:

Folder	Sub-folder	Files
certs	tls-ca	• vms-authority.crt
certs	tls server	• server.crt • server.key
config		register.graphql
		• .env • docker-compose.yml • docker-compose-production.yml

(missing or bad snippet)

Install Docker and Docker Compose

On your machine, open a terminal and run the following command to install Docker:

```
sudo apt install -y curl; \
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -; \
repo="https://download.docker.com/linux/ubuntu"; \
sudo add-apt-repository "deb [arch=amd64] ${repo} $(lsb_release -cs) stable"; \
sudo apt update; \
```

```
sudo apt install -y docker-ce; \
sudo gpasswd -a $USER docker
```

After you have installed Docker, you must install Docker Compose by running the following commands in the terminal:

```
base="https://github.com/docker/compose/releases/download/v2.16.0"; \
file="docker-compose-$(uname -s)-$(uname -m)"; \
sudo curl -L ${base}/${file} -o /usr/local/bin/docker-compose; \
sudo chmod +x /usr/local/bin/docker-compose
```

Install Lazydocker (optional)

You can also install the Lazydocker tool. Lazydocker is a tool with a terminal UI for both docker and docker-compose that can help you keep track of all the running containers.

On your machine, open a terminal and run the following command to install Lazydocker:

```
base="https://github.com/jesseduffield/lazydocker/releases/download/v0.23.0"; \
file="lazydocker_0.23.0_Linux_x86_64.tar.gz"; \
wget ${base}/${file}; \
sudo tar -zxvf ${file} -o -C /usr/local/bin lazydocker; \
rm ${file}
```

LazyDocker can only be accessed from the processing server.

Configure your DNS infrastructure

Your DNS infrastructure must be configured correctly for Milestone AI Bridge communication.

When configuring your DNS, the following things must be kept in mind:

- The machine running the Milestone AI Bridge (typically the Ubuntu machine) must be able to access all XProtect machines using machine IP addresses or hostnames.
- All machines running XProtect must be able to access the Milestone AI Bridge machine using the Milestone AI Bridge machine hostname or IP address.

Configure the XProtect Management Client machine

You must configure the XProtect Management Client to communicate with the processing server through the Milestone AI Bridge.

> If you have not yet installed the Milestone XProtect Processing Server Admin Plugin on your XProtect Management Client machine, you should do so now. See Install the Milestone XProtect Processing Server Admin Plugin on page 23 for more information.

To configure your XProtect Management Client for communication with the processing server you must also create an XProtect basic user and assign the new basic user the administrator role.

See Create a basic user for Milestone AI Bridge.

Deploying Milestone AI Bridge (Docker Compose)

To deploy Milestone AI Bridge using Docker Compose, you must retrieve Milestone AI Bridge images from the NGC container registry and then deploy the Milestone AI Bridge.

When you have deployed Milestone AI Bridge, you can check the deployment status to see how the deployment has progressed.

As an option, you can also use Lazydocker to monitor the status of the deployment.

Retrieve Milestone AI Bridge containers

To retrieve the Milestone AI Bridge container images from the NGC container registry, navigate to the folder containing the docker-compose.yml file and open a terminal on the host running Docker Compose and run the following command: `docker-compose pull`

If the command executes successfully, results similar to the example below will be displayed in the terminal. The output for pulling the individual layers is not displayed.

```
[+] Running 10/10
⠿ aibridge-streaming Pulled
⠿ aibridge-kafka-broker Pulled
⠿ aibridge-init Pulled
⠿ aibridge-health Pulled
⠿ aibridge-fuseki Pulled
⠿ aibridge-connector Pulled
⠿ aibridge-broker Pulled
```

```
⠿ aibridge-proxy Pulled
⠿ aibridge-kafka-zookeeper Pulled
⠿ aibridge-webservice Pulled
```

Deploy the Milestone AI Bridge

When you have pulled the resources, you can deploy the Milestone AI Bridge by opening a terminal on the host running Docker Compose and running the following command (replacing the values in the brackets <...> with your actual values).

```
EXTERNAL_IP=<ip-address-of-aibridge> \
EXTERNAL_HOSTNAME=<hostname-of-aibridge> \
VMS_URL=<url-of-xprotect-management-server> \
VMS_USER=<user-name-of-basic-user-in-xprotect> \
VMS_PASS=<password-of-basic-user-in-xprotect> \
MASTER_KEY=<Master key used to encrypt VMS sensitive info> \
docker-compose up -d
```

Here, the default values of the EXTERNAL_IP, EXTERNAL_HOSTNAME, VMS_URL, VMS_USER and VMS_PASS variables in the .env file are overridden by the values in the command lines.

When the command executes successfully, results similar to the output below will be displayed in the terminal.

```
[+] Running 11/11
⠿ Network ngc_default Created
⠿ Container ngc-aibridge-fuseki-1 Started
⠿ Container ngc-aibridge-kafka-zookeeper-1 Started
⠿ Container ngc-aibridge-init-1 Started
⠿ Container ngc-aibridge-health-1 Started
⠿ Container ngc-aibridge-kafka-broker-1 Started
⠿ Container ngc-aibridge-connector-1 Started
⠿ Container ngc-aibridge-proxy-1 Started
⠿ Container ngc-aibridge-streaming-1 Started
⠿ Container ngc-aibridge-webservice-1 Started
```

```
⠿ Container ngc-aibridge-broker-1 Started
```

Debug mode

The **docker compose up -d** command uses the docker-compose.yml file to run Milestone AI Bridge in debug mode. When running in debug mode, all the services are exposed on the host running Docker Compose directly.

If you want to run Milestone AI Bridge in production mode, see Deploying in a production environment (Docker Compose) on page 58.

Check deployment status

After you have set the values of the variables, you can check the status of the deployment by opening a terminal on the host running Docker Compose and running the following command: `docker-compose ps.`

This command will list each of the containers and their status.

You can monitor the init container to see that it stops, although the init container should stop within a minute or two with an exit value of 0, as displayed below. If the init container does not stop, investigate the init log files for a potential root cause.

NAME	COMMAND	SERVICE	STATUS
ngc-aibridge-broker-1	"./broker-brokersa..."	aibridge-broker	running
ngc-aibridge-connector-1	"./connector-broker..."	aibridge-connector	running
ngc-aibridge-fuseki-1	"./entrypoint.sh--u..."	aibridge-fuseki	running
ngc-aibridge-health-1	"./health-port-numb..."	aibridge-health	running
ngc-aibridge-init-1	"./init-ontology-fi..."	aibridge-init	exited(0)
ngc-aibridge-kafka-broker-1	"start-kafka.sh"	aibridge-kafka-broker	running
ngc-aibridge-kafka-zookeeper-1	"/bin/sh-c'/usr/sb..."	aibridge-kafka-zookeeper	running

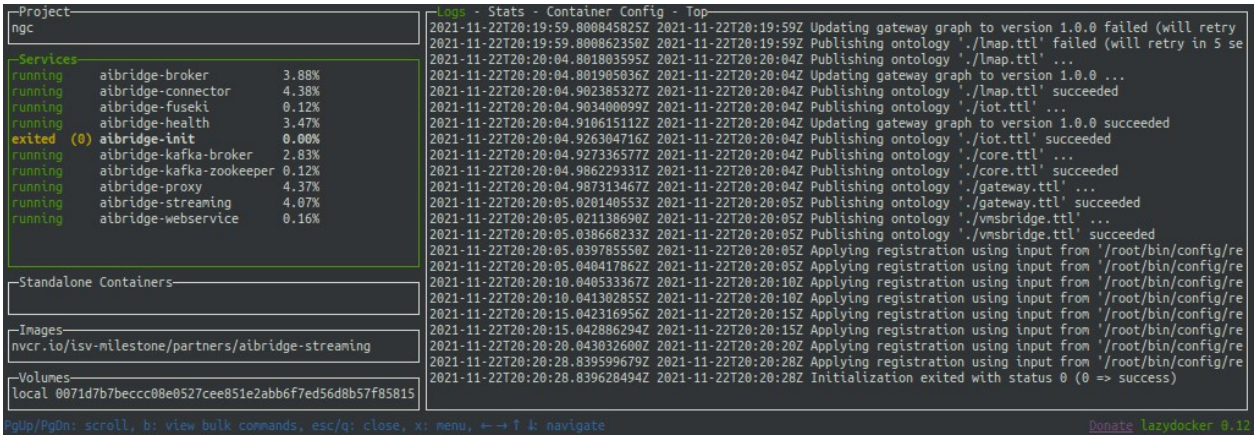
ngc-aibridge-proxy-1	"./proxy-sparql-que..."	aibridge-proxy	running
ngc-aibridge-streaming-1	"./streaming-rtsp-p..."	aibridge-streaming	running
ngc-aibridge-webservice-1	"nodeapp--document..."	aibridge-webservice	running

Using LazyDocker

You can also run lazydocker from the folder containing the docker-compose.yml file to open a user interface that will help you to monitor the status of the deployment. Lazydocker is an open-source terminal interface for managing Docker environments and enables you to inspect Docker objects without using terminal commands.

In the Lazydocker user interface, you can see the log output of each container, among other things.

The image below is an example of a successfully completed init container's log output.



Deploying in a production environment (Docker Compose)

The docker-compose.yml file facilitates debugging and experimenting with the Milestone AI Bridge, because all the services are exposed on the host running Docker Compose directly. In a production environment, you would not want to expose all these services and you can use the docker-compose-production.yml file instead.

To run Milestone AI Bridge in production mode, when deploying Milestone AI Bridgereplace this command in the terminal on the host running Docker Compose:

```
docker-compose up -d
```

with this one:

```
docker compose -f docker-compose-production.yml up -d
```

The docker-compose-production.yml file is a modified compose file that only exposes services that are absolutely required by creating an 'internal network'. In the 'internal network', containers can connect to containers by using their own container name.

To access the internal network services from your application, you can:

- Extend the compose file to also deploy your own containers
- Connect to the network defined by the Milestone AI Bridge deployment in your own docker-compose file

The default name of the internal network is ngc_default and can be accessed from another compose deployment by adding the following section to your own docker-compose file.

```
networks:
 default:
  external: true
  name: ngc_default
```

For more information, see the Docker Compose documentation https://docs.docker.com/compose/ (external link)

Configuring Milestone AI Bridge (Docker Compose)

After deploying Milestone AI Bridge , you can configure the Milestone AI Bridge by editing the .env file. The .env file contains environment parameters which are used inside the docker-compose.yml file.

The default settings of the Milestone AI Bridge are also specified in the .env file.

The .env file defines the following environment parameters:

```
# The version of the AI Bridge to run
VERSION="v2.0.0"


# How the AI Bridge will identify itself in XProtect
BRIDGE_ID="12355b21-5a25-4a1d-b6d2-f6e02c9b95b4"
BRIDGE_NAME="AI Bridge"
BRIDGE_DESCRIPTION="AI Bridge connecting IVA Apps with XProtect"
BRIDGE_WEBPAGE=""


# XProtect endpoint and credentials
```

```
VMS_URL="http://my-management-server"

VMS_USER="my-management-server-admin-username"

VMS_PASS="my-management-server-admin-password"

# Define these variables if your vms is not in the network domain (Check the
docker-compose.yaml to uncomment relevant configuration)

# VMS_IP="<my-management-server-ip>"

# VMS_HOSTNAME="<my-management-server-hostname>"


#Macro to encrypt XProtect VMS.credentials

MASTER_KEY = "<MASTER_KEY>"


# Encrypt communication with XProtect using TLS (uncomment both lines to enable)

#TLS_ENABLED="true"

#TLS_SCHEME="https"


# External IP address and hostname through which the AI Bridge services can be
reached

EXTERNAL_IP="127.0.0.1"

EXTERNAL_HOSTNAME="localhost"
```

Description

The VERSION parameters

The VERSION parameter defines the version of the Milestone AI Bridge to pull and deploy from the NGC container registry.

The BRIDGE parameters

The BRIDGE_ID parameter is a unique id (UUID) that identifies Milestone AI Bridge when connecting to the XProtect VMS.. Do not change this parameter value unless you want to run more than one Milestone AI Bridge.

The name (VMS_NAME) and description (VMS_DESCRIPTION) parameters define the display strings that you will see in XProtect Management Client for this specific Milestone AI Bridge.

The VMS parameters

With the parameter prefixed with VMS_, you can define how to connect to the XProtect VMS by providing the URL of the management server (VMS_URL) and the user name (VMS_USER) and password (VMS_PASS) of a basic user configured in XProtect.

See Create a basic user for Milestone AI Bridge for more details about how to create a basic XProtect user and how to assign access rights required for the Milestone AI Bridge to work.

Testing on a dedicated test VMS

If you use a separate VMS to test your Milestone AI Bridge solutions, your test VMS can placed in the network domain or outside the network domain.

If your test VMS is placed in the network domain, Milestone AI Bridge supports domain networks where the hostnames of the involved machines can be resolved by pointing to the DNS.

If your test VMS is placed outside the network domain, you can enable Milestone AI Bridge to resolve your VMS hostname by adapting the VMS network configurations in the values.yaml file for Kubernetes installations or the .env file for Docker-Compose installations.

If you are using Docker-Compose, you can set the VMS_IP and VMS_HOSTNAME environment variables in the .env file with the IP address and hostname of your test VMS.

The MASTER_KEY parameter

The MASTER_KEY is used to encrypt the XProtect VMS.credentials. For security reasons, it is strongly advised to encrypt the credentials of the XProtect basic user that you use in Milestone AI Bridge for logging into your XProtect VMS..

From the command line, set a value to the MASTER_KEY parameter in the .env file, as described above to encrypt the credentials at rest.

For security reasons, do not save the credentials in the .env file. You should only use the command line to pass the credentials. You can assign any value to the MASTER_KEY parameter. There are no requirements for the number or types of characters for the MASTER_KEY.

For more information, see Deploy the Milestone AI Bridge on page 56

If you forget the MASTER_KEY value

If you forget the current MASTER_KEY value, you can set a new value to the MASTER_KEY but you will need to re-register your Milestone AI Bridge and all IVA applications.

To set a new MASTER_KEY value

1. Stop all containers by running the command in the terminal: docker-compose down
2. In the .env file, enter a new value for the MASTER_KEY macro.
3. Start all containers by running the command in the terminal: `docker-compose up -d`

The TLS parameters

If your XProtect installation is not running in a secured state (running over https), the following steps are optional.

The two parameters with TLS_ are used when TLS encryption is needed.

For more information, see Securing the Milestone AI Bridge connection (Docker Compose) on page 62

The EXTERNAL_ parameters

You can specify the IP address and the DNS hostname of the machine running the Milestone AI Bridge in the EXTERNAL_IP and EXTERNAL_HOSTNAME macros respectively.

Set default parameter values

You must set the default values of the following parameters before deploying the Milestone AI Bridge.

- VMS_URL
- VMS_USER
- VMS_PASS
- EXTERNAL_IP
- EXTERNAL_HOSTNAME

You can either manually update the parameters in the .env file directly or override the settings on the command line, as described in Deploying Milestone AI Bridge (Docker Compose) on page 55.

Securing the Milestone AI Bridge connection (Docker Compose)

You can employ TLS encryption to help secure the connections between your XProtect installation and Milestone AI Bridge but before you can use TLS encryption, you will first have to enable TLS encryption for all communication in XProtect.

The Milestone Server Configurator is used to enable TLS encryption and to select the server certificates.

For more information, see the https://doc.milestonesys.com/2024r1/en-US/portal/htm/chapter-page-certificates-guide.htm.

Server certificates are issued by a Certificate Authority (CA). This can be an externally trusted certificate authority, or you can act as your own certificate authority by using a self-signed CA certificate.

In the following the certificate authority is referred to as the VMS CA and the actual CA certificate in question is referred to as the VMS CA certificate.

The zipped resource file for installing Milestone AI Bridge using Docker Compose contains a **certs** folder which contains dummy certification files. These files (vms-authority.crt, server.crt and server.key) must be replaced with your real certification files.

For more information on the resource file, see The Docker Compose resource file on page 53

The vms-authority.crt certification file in the tls-ca folder must be replaced with the VMS CA certificate to allow the Milestone AI Bridge to validate its connection to a trusted XProtect server.

The Milestone AI Bridge itself acts as a server towards XProtect and therefore must also have a server certificate issued for it by the VMS CA. This server certificate and its associated private key must be stored in the two server.crt and server.key files in the tls-server folder.

> ⚠ All certificates must use the PEM format and must be named with the .crt file extension.
> For more information, see Ubuntu manual - certificates

Once you have replaced the dummy certificate files with your own real certificates, you can enable TLS encryption for all connections between XProtect and the Milestone AI Bridge.

To enable TLS encryption

You can edit the .env file by using the HTTPS scheme in the URL of the XProtect management server and remove the comment character (#) from the two macros prefixed with TLS_.

```
# XProtect endpoint and credentials

VMS_URL="https://my-management-server"

...

# Secure services called by XProtect with TLS (uncomment both lines to disable)

TLS_ENABLED="true"

TLS_SCHEME="https"
```

For more information about the system communication and data flow in XProtect scheme, see System communication and data flow

Streaming container security considerations

For improved compliance with defined user permissions in the XProtect VMS, user oauth tokens assigned to video sent from the XProtect VMS to the IVA application must be assigned to webRTC feeds forwarded by the IVA application back into the XProtect VMS.

User oauth tokens assigned to video sent from the XProtect VMS to the IVA application can also be assigned to snapshot feeds. If you do not assign oauth tokens to snapshot feeds, the Milestone XProtect basic user defined when installing Milestone AI Bridge will be used as a token instead.

In a production environment

For production environments, IVA application developers should always set the **enforce-oauth** parameter in the **AI Bridge Streaming** (aibridge-streaming) container to **true** in the docker-compose or helm chart settings.

If the **enforce-oauth** parameter is set to **false** in a production environment, the oauth token of the Milestone XProtect basic user defined when installing the Milestone AI Bridge is used as a token. This means that snapshots or webRTC feeds from the IVA application may be available for Milestone XProtect users that otherwise do not have permission to this data.

In a test environment

For test purposes, IVA application developers can set the **enforce-oauth** parameter to **false** to facilitate testing results unless security testing is being performed.

The **enforce-oauth** parameter is located in the docker-compose.yml file.

The register.graphql file

The register.graphql file is a GraphQL mutation and is used to initialize Milestone AI Bridge, register the VMS and any IVA applications as well as contain the IVA application topic configurations.

You add and configure analytics topics to the IVA application by modifying the register.graphql file.

The file format of the register.graphql file is equivalent to what the register mutation of the GraphQL interface uses as input.

The register.graphql file is located inside the Milestone AI Bridge Helm Chart if you have deployed Milestone AI Bridge using Helm charts or inside the **config** folder, if you have deployed Milestone AI Bridge using Docker Compose.

Register Milestone AI Bridge and the VMS

The register.graphql file as it is initially installed only contains the following settings required to connect to the VMS:

```
{
  url: "${VMS_URL}"
  username: "${VMS_USER}"
  password: "${VMS_PASS}"
}
```

where VMS_URL is the address to your VMS, typically your XProtect management server, VMS_USER is the user name of the XProtect basic user dedicated for Milestone AI Bridge use and VMS_PASS is the password of the XProtect basic user.

Thus, by default, Milestone AI Bridge does not include any built-in IVA applications when installed and therefore the register.graphql file does not contain any IVA topic settings after you have installed Milestone AI Bridge. You must edit the register.graphql file to add and configure the desired IVA application analytics topics.

The zone and scope properties

You can use the zones and scope properties to optimize the amount of camera information to be read and replicated in Milestone AI Bridge by selecting which recording servers or camera device groups Milestone AI Bridge can access.

The zone property defines which recording servers Milestone AI Bridge can access, while the scope property defines which XProtect camera group Milestone AI Bridge can access. Both properties can be combined, in which case only the union can be accessed.

If no zone or scope is specified, all enabled cameras can be accessed.

Zone values (recording servers) are defined by the recording server ID, while the XProtect camera group name is used for the scope value.

For information about how to find the recording server ID, see Replace recording server

In the example below, Milestone AI Bridge can access all enabled cameras in the Building A South camera group as well as all enabled cameras on the recording server (ID: A42BB068-F6F3-4D00-8774-A6B4E05DE3E9).

Zone and scope example

```
{
  url: "${VMS_URL}"
  username: "${VMS_USER}"
  password: "${VMS_PASS}"
  scope: "Building A south"
  zone: ["A42BB068-F6F3-4D00-8774-A6B4E05DE3E9"]
}
```

The zone property

The zone property can contain an array of recording servers, each recording server ID separated by a comma.

```
  zone: ["A42BB068-F6F3-4D00-8774-A6B4E05DE3E9", "B42BB068-F6F3-4D00-8774-A6B4E05DE3E0"]
```

This enables you to specify multiple recording servers as zones.

The scope property

The scope property can only contain a single value, the camera group, but you can specify a path to a specific camera group by inserting a slash (/) as the separator, for example

```
  scope: "All Groups/All Sites/Building A south"
```

In this example, the **All Groups** camera group contains the **All Sites** camera group, which in turn contains the **Building A south** camera group.

If a slash is part of a camera group name, for example **All/Groups**, you can place the group name in quotes (single ' or double quotes " can be used), like this:

```
  scope: "'All/Groups'/All Sites/Building A south"
```

Camera group names are not required to be unique. Milestone AI Bridge can access all camera groups with the same name as the specified scope value.

Incorrect scope or zone property

If you specify an incorrect scope or zone value, no cameras can be accessed by Milestone AI Bridge.

Updating the zone and scope properties

The property values are set when you register Milestone AI Bridge. If you update the zone and scope properties later, you must unregister Milestone AI Bridge and then register it again for the changes to take effect.

IVA application topic configurations

IVA application topic configurations are saved in the register.graphql file which is read during initialization, while the init container is still running.

By default, Milestone AI Bridge does not include any built-in IVA applications when installed and therefore the register.graphql file does not contain any IVA topic settings after you have installed Milestone AI Bridge.

The register.graphql file as it is initially installed only contains the following:

```
{
  url: "${VMS_URL}"
  username: "${VMS_USER}"
  password: "${VMS_PASS}"
}
```

You can add and configure analytics topics to the IVA application by modifying the register.graphql file which is found inside the Milestone AI Bridge Helm Chart if you have deployed Milestone AI Bridge using Helm charts or inside the **config** folder, if you have deployed Milestone AI Bridge using Docker Compose.

Use your own configuration file to initialize Milestone AI Bridge

You can save your configuration file as a local file with the extension .graphql and then use it to deploy and initialize Milestone AI Bridge by adding following option to the deployment terminal command:

```
  --set-file register=<name-of-your-register.graphql>
```

This will override the content of the register.graphql file with your .graphql file instead.

Configure Milestone AI Bridge analytics topics

An analytics topic is a named feed that the IVA application can send data to and each IVA application contains one or more analytics topics.

Analytics topics are used to analyze video sequences for recognizable patterns, for example car license plates, movement, appearance, etc. and to send data back from the IVA application through the Milestone AI Bridge to your XProtect VMS if configured to do so.

An IVA application can define a number of topics when it is registered in Milestone AI Bridge. By default, no topics are defined and you can add the topics you want by modifying the register.graphql file.

By subscribing to an analytics topic, your XProtect VMS can receive data sent from the IVA application and use it in your XProtect VMS for whatever purpose is appropriate for the data received, for example displaying a video sequence or triggering an event.

There are three types of topics an IVA application can register:

- Event topics: The topic can send data back to your XProtect VMS. The data can then be used to trigger an event or an alarm, for example if an object (vehicle) is detected driving in the wrong direction, perhaps down a one-way street.
- Metadata topics: The topic can send metadata back to your XProtect VMS, for example drawing a bounding box around detected objects.
- Video topics: The topic can send processed video back to your XProtect VMS. The processed video can then be displayed like any other video stream in the VMS, for example a video stream can be sent back to the VMS where key identified areas of the original video have been blurred.

IVA applications

IVA applications can be created as self-registering or IVA applications that are not self-registering.

Self-registering IVA applications

Self-registering IVA applications have connection details embedded in the initialization of the IVA application itself and do not need to be manually provided, for example in the register.graphql file.

Self-registering IVA applications will register the application and all its topics and their configuration settings whenever the application itself is started.

Milestone recommends the use of self-registering IVA applications to improve the process when updating the application.

IVA applications that are not self-registering

IVA applications that are not self-registering will require the processing server to be started or restarted to register the application and all its topics and their configuration settings. The register.graphql file must be modified to include the registration for the IVA application in question. When the processing server is started or restarted, the register.graphql file will be read and all IVA applications correctly defined in the register.graphql file will registered.

This is especially relevant when changes are made to the configuration settings of the topics in already registered IVA applications.

Editing IVA application topic settings

The only way you can change the configuration settings of topics in an IVA application that is not self-registering is to edit the register.graphql file. Since the register.graphql file is only read when the processing server starts up, the processing server must be restarted to deploy the new topic configuration.

A self-registering IVA application will register any changes made to its topic configuration when the IVA application itself is started. This way, you will not need to restart the processing server if all you have done is edit a few topic configurations in the IVA applications.

Self-registering IVA application characteristics

A self-registering IVA application must query the endpoint register with the IVA application configuration data as well as with the unique identifier of the video management system the IVA application wants to register on.

You can obtain the unique identifier of your video management system by requesting the identifier in a GraphQL query.

GraphQL query example of a video management system ID request

```
query {
  about {
    videoManagementSystems {
      id
    }
  }
}
```

The requested video management system ID can then be used to register the IVA application later in the GraphQL query as depicted in the example below.

Example of application of video management system ID in a GraphQL query

```
mutation {
  register(
    input: {
      id: "<The vms id obtained from the about query>"
      apps: {
        id: "<An app id assigned by the app developer>"
        url: "<An url to the app webservice for example>"
        name: "<App name>"
        description: "<App description>"
        version: "<App version>"
        manufacturer: {
          name: "<Manufacturer name>"
      }
        eventTopics:[{
          url: "<Path to topic handler>"
          name: "<Topic name>"
          description: "<Topic description>"
          eventFormat: ANALYTICS_EVENT # there is one format
        }],
        metadataTopics:[{
          url: "<Path to topic handler>"
          name: "<Topic name>"
          description: "<Topic description>"
          metadataFormat: ONVIF_ANALYTICS # ONVIF_ANALYTICS, ONVIF_ANALYTICS_
FRAME, DEEPSTREAM
          }],
          videoTopics:[{
            url: "<Path to topic handler>"
            name: "<Topic name>"
```

```
            description: "<Topic description>"
            videoCodec: H265 # MJPEG, H264 or H265
        }]
      }
    }
  ) {
      id
  }
}
```

IVA application registration

When the IVA application is successfully registered, a status code 200 response with the unique identifier of the specified video management system will be displayed. Additionally, the topics of the IVA application are displayed in the Process server tab in XProtect Management Client.

If you do not receive the Status Code 200 response or if the IVA application topics are not displayed correctly, the IVA application may not be registered correctly, or the IVA topics themselves may not be configured correctly.

Editing self-registering IVA application settings

If you need to edit the configuration settings of a self-registering IVA application, including any topic configuration settings, you should edit the relevant sections of the GraphQL query for the IVA application instead of the register.graphql file.

Other ways of editing IVA application settings

Some IVA applications utilize their own configuration file which can be edited directly and some IVA applications contain internal configuration settings that are edited from within the IVA application.

For these IVA applications, you must update the IVA and/or topic configurations in the relevant places instead of the GraphQL query for the IVA application or the register.graphql file.

Adding and configuring Analytics topics

Milestone AI Bridge and its default IVA applications do not contain any pre-configured analytics topics, as topic configuration will depend on the IVA application itself and the analytics topics the application contains as well as the needs and requirements of your organization.

You can add and configure analytics topics to the IVA application by modifying the register.graphql file.

Topics and XProtect Management Client

The configured topics will be displayed in the XProtect Management Client on the **Processing Server** node and can be subscribed to through the XProtect Management Client.

If the IVA application contains a web interface that is exposed in the XProtect Management Client, you can perform additional configuration of the topic through the web interface from XProtect Management Client.

The Milestone AI Bridge reference manual

The Milestone AI Bridge reference manual provides reference material about the Milestone AI Bridge GraphQL API andits elements. It also contains the GraphiQL and GraphQL Voyager services, which are two GraphQL services that you can use to test your GraphQL queries and get an overview of the GraphQL API.

GraphiQL

GraphiQL is a browser-based user interface that can be used for editing, testing and executing GraphQL queries and mutations against a GraphQL API. GraphiQL enables you to correctly structure your GraphQL queries.

Additionally, you can use the GraphiQL to experiment and build your queries using live data from a real VMS and is a good resource for exploring and learning the API.

GraphiQL enables you to access the API's documentation directly and includes syntax highlighting, intellisense, auto-completion as well as automatic documentation.

GraphiQL is enabled when running in debug mode.

If Milestone AI Bridge is deployed on localhost, you can access GraphiQL at http://localhost:4000/api/bridge/graphql.

GraphQL Voyager

GraphQL Voyager enables you to visually explore the GraphQL API as an interactive graph, helping you get an overview of how everything connects.

You can access GraphQL Voyager at http://localhost:4000/voyager. The Milestone AI Bridge reference manual content is also available directly in GraphQL Voyager.

Accessing the reference manual

The reference manual can be accessed as a web page hosted by Milestone AI Bridge but only if you have deployed Milestone AI Bridge to run in debug mode.

If Milestone AI Bridge is running in production mode, you cannot access the Milestone AI Bridge reference manual unless you download a copy and access the manual locally.

The reference manual is accessed through port 4000 (for example http://localhost:4000).

If you are using Kubernetes

You can access the Milestone AI Bridge Reference Manual from outside the Kubernetes cluster by using the following URL: `http://<kubernetes-cluster-hostname>:4000` where <kubernetes-cluster-hostname> is the hostname of your Kubernetes cluster.

The GraphiQL query interface is accessed by using the following URL: `http://<kubernetes-cluster-hostname>:4000/api/bridge/graphql`.

GraphQL Voyager is accessed by using the following URL: `http://<kubernetes-cluster-hostname>:4000/voyager`.

If you are using Docker-Compose

You can access the Milestone AI Bridge Reference Manual on the host running Docker Compose by using the following URL: http://<hostname-of-aibridge>:4000 where <hostname-of-aibridge> is the hostname of your machine.

The GraphiQL query interface is accessed by using the following URL: `http://<hostname-of-aibridge>:4000/api/bridge/graphql`.

GraphQL Voyager is accessed by using the following URL: `http://<hostname-of-aibridge>:4000/voyager`.

Download a local copy

The Milestone AI Bridge reference manual can also be downloaded from the NGC portal and unpacked locally. You can access the content by opening the Index.htm file in a browser.

The the GraphiQL and GraphQL Voyager services will not be available when opening a downloaded copy of the Milestone AI Bridge reference manual.

Troubleshooting

Log files

Configuring and accessing your system log files will depend on whether you use Docker-Compose or Kubernetes to manage your containers.

Docker-Compose

For Docker-compose installations, separate log files are created for each Milestone AI Bridge container during operations and are mounted in volumes on the host machine in the /var/log/aib/[container-name] folder. For example, the log file for the Milestone AI Bridge Webservice container (webservice.log) is in the `/var/log/aib/aibridge`-webservice folder.

Add the desired log parameters to the docker-compose-production.yml or docker-compose.yml on each container, for example, the Milestone AI Bridge Webservice container.

You must have administrator privileges for the Docker file system to access the log files directly.

Log file retention

You can use the log-max-backups, log-max-size, and log-max-age log parameters described below to create an impromptu retention policy for your log files.

Whenever a log file exceeds the value defined for the log-max-size parameter, a new log file is created with the same name and the old log file is compressed into a .zip file and renamed with the year, month, and day suffix (YYYY-MM-DD).

For example, the **webservice.log** file is renamed to **webservice_YYYY_MM_DD.zip**, where YYYY is the current year, MM is the current month, and DD is the day the compressed log file is created. A new webservice.log file is also created to contain new incoming log messages.

A compressed log file is automatically deleted when its age exceeds the limit defined in log-max-age log parameter or whenever the number of compressed log files exceeds the limit defined in the log-max-backups parameter.

If you want to archive your compressed log files for longer than the values in the log parameters have defined, you must move them to another location before the log-max-age or log-max-backups parameter values are exceeded and the compressed log files are deleted.

Log parameters

Five optional log parameters are available and used to specify enabling/disabling, size, permitted number, retention period, and logging level of all log files. The log file parameters are defined in the docker-compose files.

If you do not specify any log file parameter values, the default values noted in the Docker-Compose log parameters on page 76 table below are used.

Docker-Compose syntax example

```
aibridge-webservice
# preceding yaml file content
 command: -- id "${BRIDGE_ID}"
  --name "${BRIDGE_NAME}"
  --description "${BRIDGE_DESCRIPTION}"
    --log-file-enabled=True
  --log-max-size 150
  --log-max-backups 10
  --log-level info
# yaml file content continues
```

Docker-Compose log parameters

Parameter	Description
log-file-enabled	Boolean, expressed as True or False Enables or disables the option to mount log files into volumes • If set to True, log files are mounted in volumes. • If set to False, log files are not created and logging is disabled. The default value is False.
log-max-size	Numeric, expressed in megabytes. The maximum file size for a single log file. If a log file exceeds its maximum size, a new log file is created to contain incoming log messages, and the old log file is compressed into a .zip file and renamed with a _YYYY_MM_DD suffix where YYYY is the current year, MM is the current month and DD is the day the compressed log file is created. The default value is 100 megabytes. You can disable this parameter by setting the value to 0.

Parameter	Description
log-max-backups	Numeric The maximum number of retained compressed log files. If the maximum number of retained compressed log files is exceeded, the oldest compressed log file is deleted. The default value is 15 log files. You can disable this parameter by setting the value to 0.
log-max-age	Numeric, expressed in days The maximum number of days that compressed log files are to be retained, based on the compressed log file's date suffix (the "_YYYY-MM_DD" part of the file name). When a new compressed log file is created, existing compressed log files older than the retention period is deleted. The default value is 15 days. You can disable this parameter by setting the value to 0.  You can not apply this parameter to the Milestone AI Bridge Webservice container.
log-level	String Defines which errors that are logged ("error", "warn", "info", "debug"). Values: • error: Logs error messages only • warn: Logs error and warning messages • info: Logs error, warning, and information messages • debug: Logs all messages (error, warning, information, and debug) The default value is "info". You can not disable this parameter, but the number of entries in the log file can be reduced by using the "error" value which leads to the lowest number of entries in the log.  This parameter is only applicable on the Milestone AI Bridge Webservice container.

Kubernetes

If you are using Kubernetes, you do not need to specify log parameters as log files are automatically generated on each pod using Kubernetes own logging system.

In general, the logs of each pod can be found in the location /var/logs/pods.

For more information, see Logging Architecture (external link)

Video length errors

The **rtsp-write-buffer-count** parameter in the Milestone AI Bridge streaming container helps control and queue internal RTSP buffers with video from the VMS system.

If an incoming video package is larger than the **rtsp-write-buffer-count** parameter value, the video stream package will not be forwarded by the Milestone AI Bridge and the following entry in the Milestone AI Bridge streaming log file will be written:

"RTSP buffer (Camera ID: [XXX-AAAA] - Stream ID: [YYY-BBB]): The VMS system has sent a video package with the length of '[Integer-Value-Here]'. However, Milestone AI Bridge can only handle packages whose maximum length is '[IntegerValue-Here]'. Increase the length of the 'rtsp-write-buffer-count' parameter in the Milestone AI Bridge Streaming container to handle longer video packages."

Log file location

The streaming log file is located on the host machine running Milestone AI Bridge or inside the streaming container in the /var/log/aib/aibridge-streaming folder.

The parameter value

The **rtsp-write-buffer-count** parameter value defines the maximum number of RTP packets the streaming buffer can contain. If too many RTP packets are received, a buffer overrun will occur and the video will not be processed.

Too many RTP packets can occur if large video frames are sent, as large video frames will generate many RTP packets in order to transfer the required data.

Keyframes (the first video frame) are usually substantially larger than the subsequent video frames and are usually sufficient to cause a buffer overrun by themselves. Compression levels and detail levels both affect the size of the keyframe, the lower the compression level or the higher the level of detail, the larger the keyframe will be.

RTP packet buffer overruns may occur for seemingly random cameras with no apparent connection to video resolution, size, or other characteristics.

The parameter value is an integer with a default value of 1024.

You can change the default value and specify other package lengths if necessary. However, all parameter values must be specified as a value to the power of 2 (1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, and so on.)

If you specify erroneous parameter values, you will not be able to save your changes until you specify a valid parameter value.

Editing the parameter

If you are experiencing loss of video packages in the communication between your VMS and the Milestone AI Bridge, you can try to change the **rtsp-write-buffer-count** parameter value, adjusting it until Milestone AI Bridge forwards all video packages correctly.

The parameter is defined in the docker-compose or helm chart files, depending on which container management application you use: Docker Compose or Kubernetes.

Docker-Compose

Add the desired log parameters to the docker-compose-production.yml or docker-compose.yml on the Milestone AI Bridge Streaming container.

Syntax example, docker compose

```
aibridge-streaming:
# preceding yaml file content
 command: -rtsp-port 8554
  -rtsp-rtp-udp-port 8000
  -rtsp-rtcp-udp-port 8001
  -rtsp-write-buffer-count 1024
  -webrtc-external-ip ${EXTERNAL_IP}
# yaml file content continues
```

Kubernetes

In helm charts, add the parameter to the Milestone AI Bridge Webservice template file.

Syntax example, Kubernetes

```
 args: [
...
  "-rtsp-port,              8554",
  "-rtsp-rtp-udp-port       8000",
  "-rtsp-rtcp-udp-port      8001",
```

```
  "-rtsp-write-buffer-count 1024",]
...
```

Update the URL of your VPS hardware

When you subscribe to a video or metadata topic, new hardware will be created using the VPS driver.

If you change the protocol used by your XProtect VMS or your Milestone AI Bridge from https (secured) to http (unsecured) or from http (unsecured) to https (secured), you must also adjust the URLs defined in your hardware settings to run the selected protocol as well.

Additionally, changing the Milestone AI Bridge configuration parameter **EXTERNAL_ROOT_PATH** will also impact the URLs for hardware using the VPS driver and you must update these URLs to the new URL of Milestone AI Bridge as well.

To update the URL for your device

1. Open your Management Client and in the **Servers** node, click **Recording Servers**.
2. In the **Recording Servers** pane, select the device you want to update the URL for.
3. In the **Properties** pane > **Driver parameters** field, enter the new URL for the device.

Updating and upgrading

Updating the Recording Server configuration

Milestone AI Bridge continually monitors and synchronizes changes detected in the Recording Server configuration. Implementing major changes to the configuration impacts the performance of Milestone AI Bridge because each change will trigger a new synchronization.

If you plan to implement major changes to the Recording Server configuration, for example, adding, moving, or removing many cameras or adding or removing additional recording servers, start by shutting down Milestone AI Bridge. When you have implemented your changes, re-start Milestone AI Bridge and any IVA applications.

How you stop and start your Milestone AI Bridge installation will depend on whether you use Docker-Compoase container or Kubernetes pods.

Stopping and starting Docker-Compose containers

You can stop all running Docker-Compose containers without removing them and start all existing Docker-Compose containers by using the terminal and command line interfaces.

To stop all Docker-Compose containers

On the processing server, open a terminal and run the command: `docker-compose down`

To start all Docker-Compose containers

On the processing server, open a terminal and run the command: `docker-compose up`

Stopping and starting Kubernetes clusters and pods

Unlike stopping and starting Docker-Compose containers, Kubernetes clusters and pods cannot be paused or stopped. Instead, you must remove the pods and reinstall them.

Updating and upgrading your Milestone AI Bridge

It may become desirable or necessary to update or upgrade various components of Milestone AI Bridge as enhanced functionality, new features, improved security and privacy, and bug fixes become available.

Updating is the application of new or improved features to an existing released version of your XProtect VMS and Milestone AI Bridge components while upgrading is the process of installing and configuring a new released version of your XProtect VMS and Milestone AI Bridge components. Installing an older released version is often referred to as downgrading.

Updating the processing server operating system

You can update the Linux operating system of your processing server as necessary, applying hotfixes, patches and updates using whichever update method you find most efficient, for example through the command line or using the Software Updater GUI tool or other update tools available.

Upgrading the processing server operating system

It is not advised to change the version of the Linux operating system of your processing server as other versions of Ubuntu or other Linux distributions may not be compatible with running a processing server.

If you are required to change the version of Linux operating system, make sure the new version is compatible with running a processing server.

> Changing the operating system version will install a new version of the operating system on the machine and irrevocably delete any data stored on the hard drive of the local machine.
> Milestone does not supply any internal tool for managing data backup on the processing server and you must identify and save any data you wish to keep.

Updating the Milestone XProtect Processing Server Admin Plugin

To update your Milestone XProtect Processing Server Admin Plugin, download and install the newest plugin from the NGC portal, following any instructions that are displayed.

Milestone XProtect Processing Server Admin Plugin is a MIP plugin, and conforms MIP plugin backup policies.

Updating the Milestone AI Bridge patch

If you are running an XProtect VMS version that requires the Milestone AI Bridge patch, you can update the patch by downloading and applying the new patch that corresponds to your XProtect VMS version from the NGC Portal.

Upgrading your XProtect VMS

If you are upgrading or downgrading your XProtect VMS, refer to the processing server support matrix to determine any other requirements the XProtect VMS version must meet in order to be able to access the processing server and utilize the installed IVA applications.

In some cases, you will have to download and install a new Milestone AI Bridge patch for your version of XProtect VMS, in others, you may only have to re-apply the Milestone XProtect Processing Server Admin Plugin.

For more information, see Milestone AI Bridge support matrix on page 27

Updating Milestone AI Bridge components

Milestone AI Bridge consists of multiple container images with each image used for specific functions within the entire solution. All container images are available for download directly from the NGC portal.

Updating and upgrading your Milestone AI Bridge is identical to installing a new version of the Milestone AI Bridge.

Update/upgrade using Kubernetes and the Helm chart

If you are using Kubernetes and the Helm chart, you must fetch the new version of the Helm chart and thereafter fetch and deploy any dependencies of the Helm chart This will deploy the newest Milestone AI Bridge components, including container images.

Update/upgrade using Docker Compose

If you are using Docker Compose, you must download the latest version of the zipped resource file that contains the docker-compose.yml file from the NGC portal and using the docker-compose.yml file, retrieve the container images defined in the VERSION parameter in the .env file.

helpfeedback@milestone.dk

About Milestone

Milestone Systems is a leading provider of open platform video management software; technology that helps the world see how to ensure safety, protect assets and increase business efficiency. Milestone Systems enables an open platform community that drives collaboration and innovation in the development and use of network video technology, with reliable and scalable solutions that are proven in more than 150,000 sites worldwide. Founded in 1998, Milestone Systems is a stand-alone company in the Canon Group. For more information, visit https://www.milestonesys.com/.