Runtime Installation Wizard

⚠ **Disclaimer!** 🚧 This release is provided as a technical preview and is not intended for production environments.

✓ Milestone Systems A/S does not collect any data or telemetry from this installation.

Introduction

This document describes the official procedure for installing the **Milestone Runtime Environment** via the Runtime Installation Wizard.
It is intended for our external partners and covers the following objectives:

1. **Deploy Milestone Private Cloud** and **App Center** onto a dedicated Kubernetes cluster.
2. **Install the** `App Center MIP Plugin`, enabling the access to the App Center from the XProtect Video Management Client.
3. **Install, start, and upgrade** all associated services in a controlled and repeatable manner.

The Runtime Installation Wizard is a desktop application available for `Windows`. It uses Ansible and Kubernetes to provision the Runtime Environment on a remote Ubuntu server.

System Requirements & Prerequisites

The Runtime Installation Wizard is a desktop GUI application that can be run from Windows.

The Milestone Runtime Environment requires to be installed into the remote Linux machine (only Ubuntu is supported at the moment)

Client Machine Requirements

- **Operating System:** Windows 11 or later, or a supported Linux distribution
- **Network:** Ability to establish SSH connections to the target server
- **User Privileges:** Local user must have permission to install and launch desktop applications

Target Server Requirements

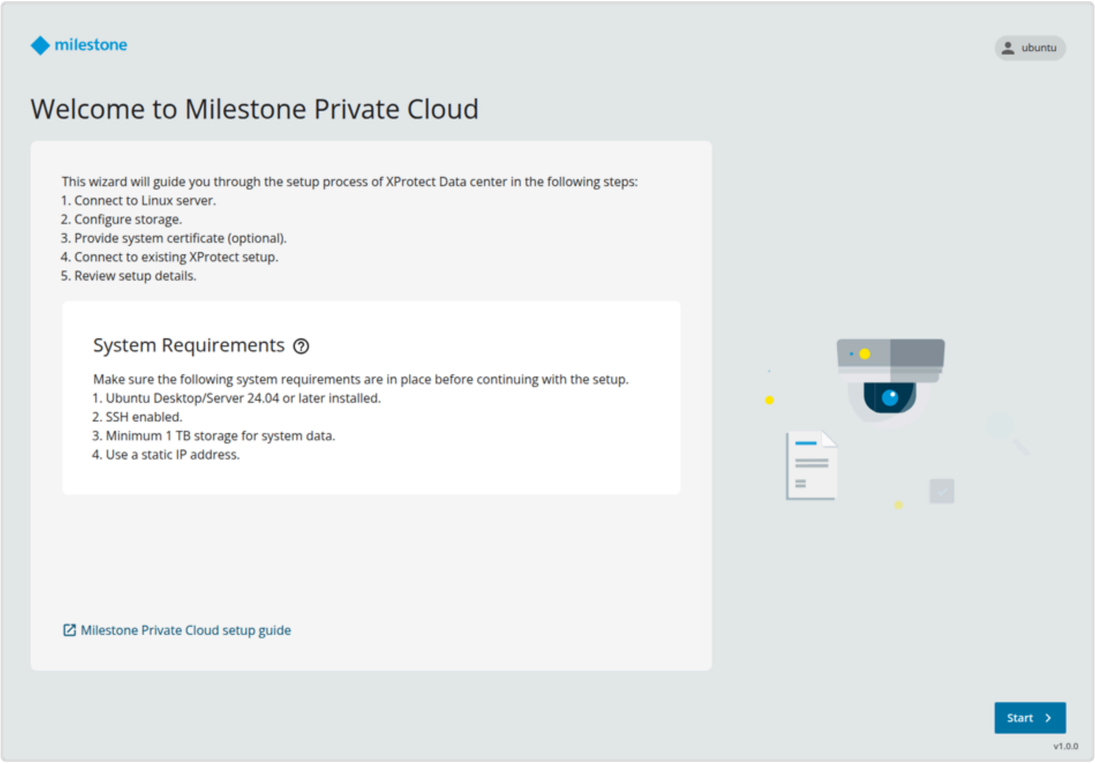
- **Operating System:** Ubuntu 24.04 LTS or later
- **Memory:** Minimum of 8 GB RAM
- **CPU:** Minimum of 4 cores
- **Storage:** Minimum of 20 GB available disk space for `Video Degradation` and `logs`
- **SSH Access:** Key-based, passwordless SSH authentication is strongly recommended
  - run the following command `sudo apt install openssh-server` to install the ssh server.
- **Sudo Rights:** The deploying user must possess sudo privileges
- Static IP Address:
  - For detailed guidance, refer to Configure Static IP Address on Ubuntu

XProtect Installation Access Requirements

Before starting the installation, please create a basic administrator account in the XProtect VMS.

Installing Milestone Runtime Environment (GUI)

- Download the latest version of the Installation Wizard from here.
- Unzip the content of the file above and double-click the executable file to start the installation process.



Welcome to Milestone Private Cloud

This wizard will guide you through the setup process of XProtect Data center in the following steps:
1. Connect to Linux server.
2. Configure storage.
3. Provide system certificate (optional).
4. Connect to existing XProtect setup.
5. Review setup details.

System Requirements

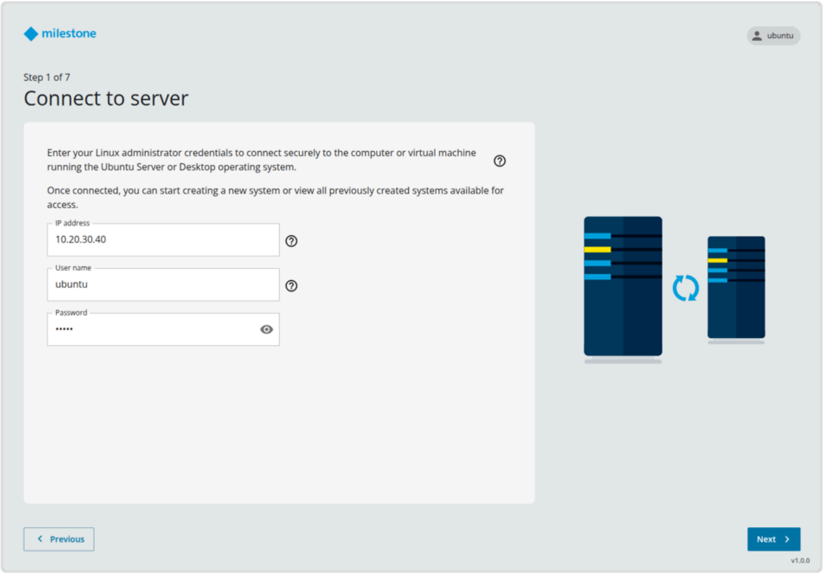
Make sure the following system requirements are in place before continuing with the setup.
1. Ubuntu Desktop/Server 24.04 or later installed.
2. SSH enabled.
3. Minimum 1 TB storage for system data.
4. Use a static IP address.

Once the Welcome screen appears, you will find the application version in the bottom-right corner.

- Click Start button on the right-bottom corner.
- A Connect to server form will appear.

  The Connect to server form requires you to enter the remote server ip address, user name and password.

Connect to Server step



Step 1 of 7

Connect to server

Enter your Linux administrator credentials to connect securely to the computer or virtual machine running the Ubuntu Server or Desktop operating system.

Once connected, you can start creating a new system or view all previously created systems available for access.

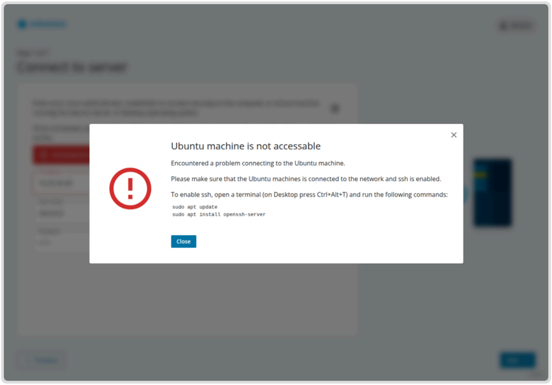
IP address: 10.20.30.40

User name: ubuntu

Password: •••••

When you reach the Connect to Server step, provide the following information:

- **IP address** – Enter the remote server's address. You may use either an IP address or a domain name. Only static IP addresses are supported; dynamic addresses are not permitted. See the prerequisites for more details.
- **User name** - Enter your Ubuntu user name. This account must have sudo privileges.
- **Password** - Enter the password for that user account.



Ubuntu machine is not accessable

Encountered a problem connecting to the Ubuntu machine.

Please make sure that the Ubuntu machines is connected to the network and ssh is enabled.

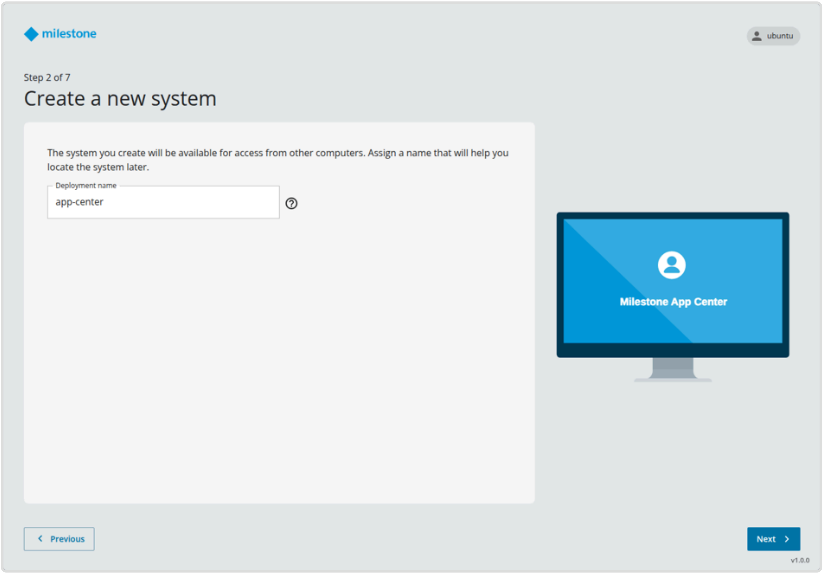
To enable ssh, open a terminal (on Desktop press Ctrl+Alt+T) and run the following commands:

```
sudo apt update
sudo apt install openssh-server
```

If you make a mistake, an error popup will appear. Click the error bar to view detailed information about the issue.

Once all fields are completed, click `Next` to proceed to the next step.

Create a new system - deployment name



Step 2 of 7

Create a new system

The system you create will be available for access from other computers. Assign a name that will help you locate the system later.

Deployment name: app-center

There is only one field - deployment name

In this step, you will assign a name to your new cluster (klynge). There is only one field:

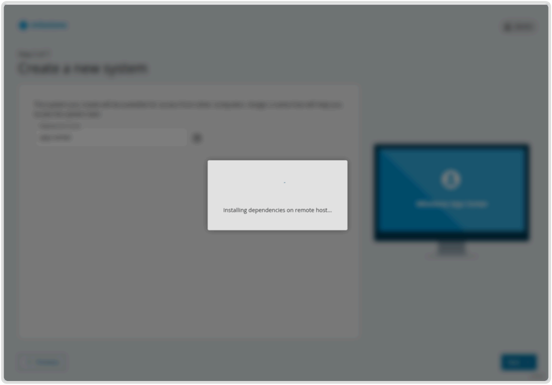
- **Deployment Name** – Enter a single, letter-only/dash name. For example: `app-center`.

Press `Next` button to continue.

Upon doing so, the Wizard will:

1. Establish a secure connection to your remote server.
2. Install the necessary prerequisites.
3. Collect storage information.

These actions prepare your environment for the subsequent step.



Installing dependencies on remote host...

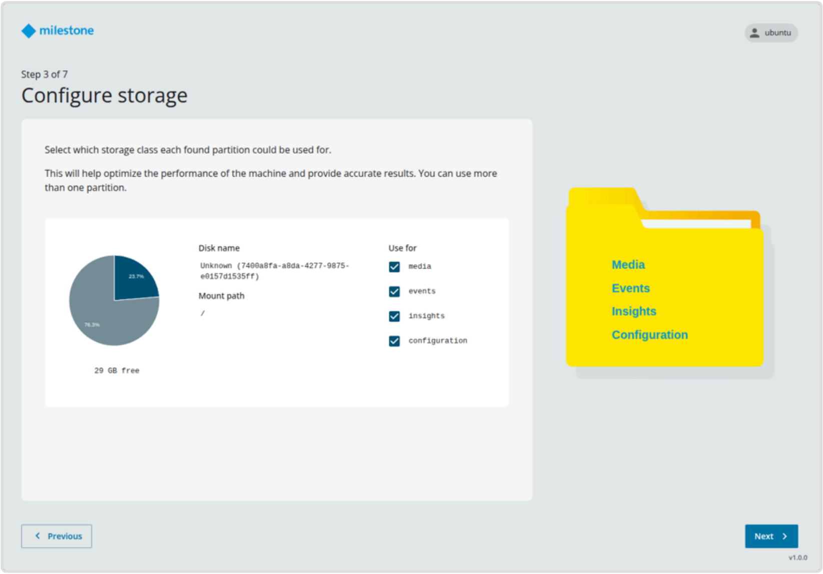
Wizard is collecting required information.

Configure storage

In this step, the Wizard displays each discovered partition along with its disk name , mount path and free space.
Please select one or more storage classes for each partition to match your deployment needs. Proper assignment helps optimize performance and ensures accurate operation of your Kubernetes cluster.

Available Storage Classes:

- **Media** – Stores recorded video streams and media files.



milestone

ubuntu

Step 3 of 7

Configure storage

Select which storage class each found partition could be used for.

This will help optimize the performance of the machine and provide accurate results. You can use more than one partition.

Disk name

```
Unknown (7400a8fa-a8da-4277-9875-e0157d1535ff)
```

Mount path

```
/
```

Use for

☑ `media`

☑ `events`

☑ `insights`

☑ `configuration`

`29 GB free`

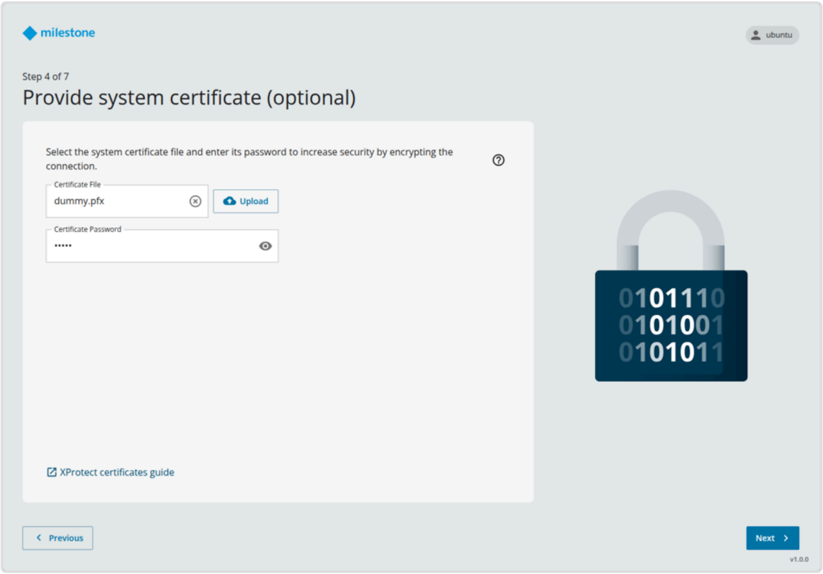
- **Events** – Stores event metadata and alarm history.
- **Insights** – Stores analytics data and reporting results.
- **Configuration** – Stores system configuration, logs and application state.

1. Review each partition’s size and mount point.
2. For each partition, tick the checkbox(es) beside the class(es) you wish to use.
3. When you have assigned classes to all required partitions, click **Next** (Næste) to proceed.

> You may assign multiple classes to a single partition if it has sufficient capacity. Conversely, you can distribute classes across separate partitions to isolate workloads.

Provide system certificate

In this optional step, you may upload a system certificate to encrypt communication between components in your Kubernetes cluster. If you prefer not to use a certificate at this time, simply click **Next** to skip this step.



milestone

ubuntu

Step 4 of 7

Provide system certificate (optional)

Select the system certificate file and enter its password to increase security by encrypting the connection.

Certificate File: dummy.pfx

Certificate Password: •••••

XProtect certificates guide

Procedure:

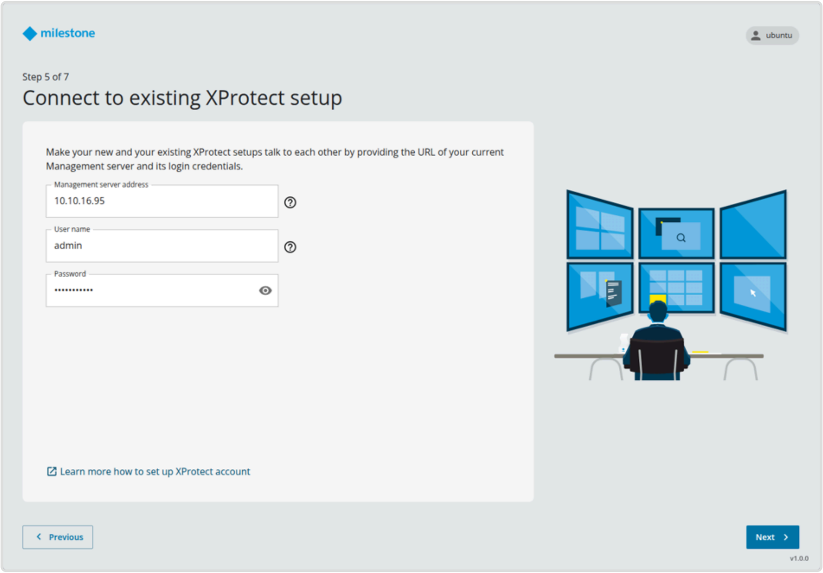
1. **Certificate File** – Click **Upload** to select a valid `.pfx` file containing your certificate and private key.
2. **Certificate Password** – Enter the password that protects the

`.pfx` file.

3. After selecting the file and entering its password, click **Next** to continue.

For guidance on generating or converting certificates, please refer to the [XProtect Certificates Guide]().

Connect to XProtect server



Step 5 of 7

Connect to existing XProtect setup

Make your new and your existing XProtect setups talk to each other by providing the URL of your current Management server and its login credentials.

Management server address: 10.10.16.95

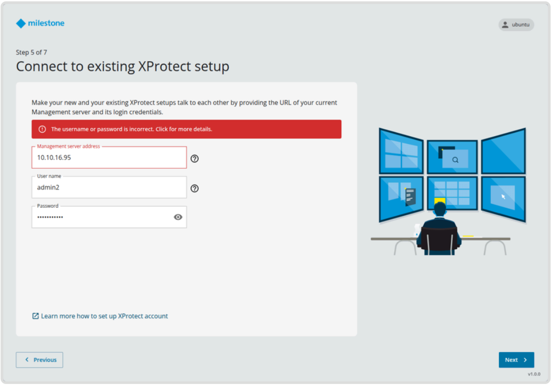
User name: admin

Password: •••••••••••

Learn more how to set up XProtect account

This **mandatory** step enables your Runtime **cluster** to communicate with the XProtect Management Server and leverage its APIs.

- **Management Server Address**
  - Enter the URL or IP address (or domain name) of your current XProtect Management Server.
  - Example: `10.10.16.95` or `xprotect.example.com`
- **User Name**
  - Provide a basic account with administrator privileges on the Management Server. This account will be used by the Runtime cluster to authenticate against XProtect APIs.
- **Password**
  - Enter the password for the above user.



Step 5 of 7

Connect to existing XProtect setup

Make your new and your existing XProtect setups talk to each other by providing the URL of your current Management server and its login credentials.

The username or password is incorrect. Click for more details.

Management server address: 10.10.16.95

User name: admin2

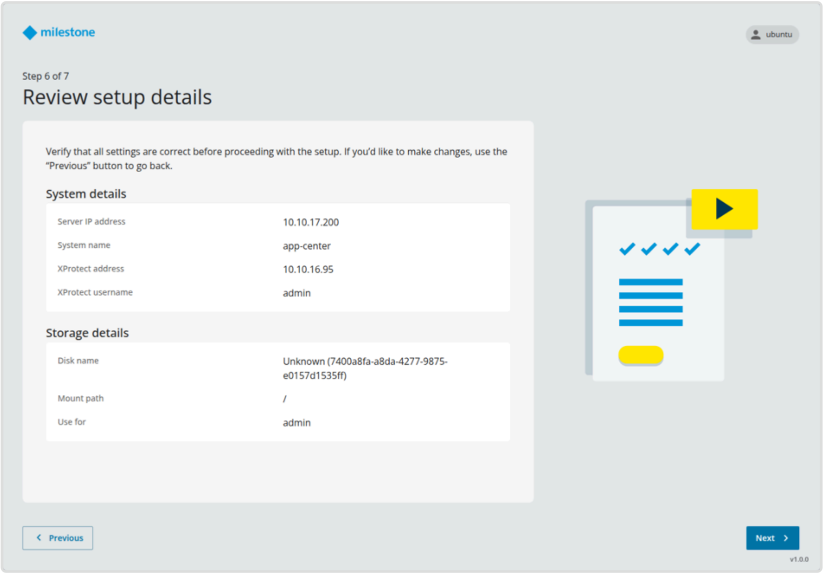
Password: •••••••••••

Learn more how to set up XProtect account

Something went wrong.

Once all fields are filled, click `Next` to validate the connection. The Wizard will verify credentials, discover available APIs.

Review setup details



Step 6 of 7

Review setup details

Verify that all settings are correct before proceeding with the setup. If you'd like to make changes, use the "Previous" button to go back.

System details

Server IP address	10.10.17.200
System name	app-center
XProtect address	10.10.16.95
XProtect username	admin

Storage details

Disk name	Unknown (7400a8fa-a8da-4277-9875-e0157d1535ff)
Mount path	/
Use for	admin

Review details window

Before initiating the deployment, please verify that all configuration settings are correct. If you need to make changes, click **Previous** to return to the relevant step.

System Details

- **Server IP address** – The address of the Ubuntu host where the Runtime will be deployed.
- **Deployment name** – The cluster name you provided.
- **XProtect address** – The URL or IP of your XProtect Management Server.
- **XProtect username** – The administrator account used to authenticate with XProtect.

Storage Details

- **Disk name** – The identifier of the target partition.
- **Mount path** – The filesystem mount point.
- **Assigned classes** – The storage classes (Media, Events, Insights, Configuration) selected for this partition.

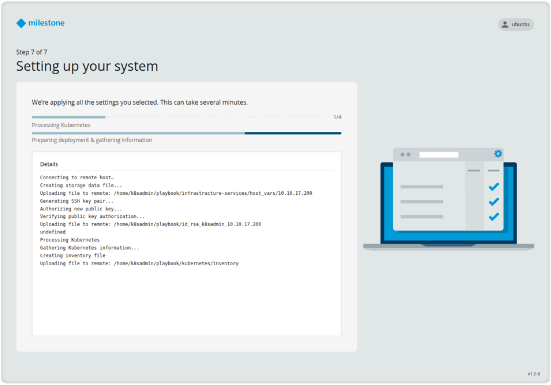
When you have confirmed that every detail is correct, click **Next** to begin the installation process. The Wizard will then execute the Ansible playbooks to provision your Kubernetes cluster.

Setting up your system

In this final step, the Wizard applies all of your selected settings and provisions the entire Kubernetes cluster. This process can take approximately **15–20 minutes** to complete.

What happens during deployment:

1. **Processing Kubernetes and its dependencies** – The Wizard generates and uploads Ansible playbooks to configure your cluster.
2. **Preparing deployment & gathering information** – Storage data files, SSH keys, and inventory manifests are created and transferred to the server.
3. **Installing services** – All required components (Milestone Private Cloud, App Center, Client Plugin, gateways, etc.) are deployed.
4. **Finalizing configuration** – Certificates, credentials, and network settings are applied.



Step 7 of 7

Setting up your system

We're applying all the settings you selected. This can take several minutes.

1/4

Processing Kubernetes

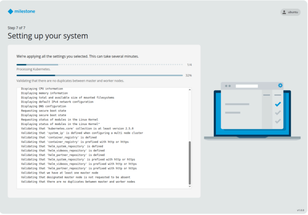
Preparing deployment & gathering information

Details

```
Connecting to remote host…
Creating storage data file...
Uploading file to remote: /home/k8sadmin/playbook/infrastructure-services/host_vars/10.10.17.200
Generating SSH key pair...
Authorizing new public key...
Verifying public key authorization...
Uploading file to remote: /home/k8sadmin/playbook/id_rsa_k8sadmin_10.10.17.200
undefined
Processing Kubernetes
Gathering Kubernetes information...
Creating inventory file
Uploading file to remote: /home/k8sadmin/playbook/kubernetes/inventory
```

A live progress shows the number of sub-tasks completed, and a detailed log pane displays each action. Once all stages reach 4/4, the deployment is finished successfully.

⚠ Please do not close the Wizard or interrupt your network connection during this phase.



Step 7 of 7

Setting up your system

We're applying all the settings you selected. This can take several minutes.

1/4

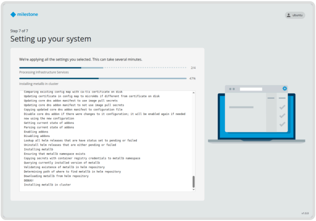
Processing Kubernetes.

32%

Validating that there are no duplicates between master and worker nodes.

```
Displaying CPU information
Displaying memory information
Displaying total and available size of mounted filesystems
Displaying default IPv4 network configuration
Displaying DNS configuration
Requesting secure boot state
Displaying secure boot state
Requesting status of modules in the Linux Kernel
Displaying status of modules in the Linux Kernel"
Validating that 'kubernetes.core' collection is at least version 2.5.0
Validating that 'system_ip' is defined when configuring a multi node cluster
Validating that 'container_registry' is defined
Validating that 'container_registry' is prefixed with http or https
Validating that 'helm_system_repository' is defined
Validating that 'helm_videoos_repository' is defined
Validating that 'helm_partner_repository' is defined
Validating that 'helm_system_repository' is prefixed with http or https
Validating that 'helm_videoos_repository' is prefixed with http or https
Validating that 'helm_partner_repository' is prefixed with http or https
Validating that we have at least one master node
Validating that designated master node is not requested to be absent
Validating that there are no duplicates between master and worker nodes
```

step 1



Step 7 of 7

Setting up your system

We're applying all the settings you selected. This can take several minutes.

2/4

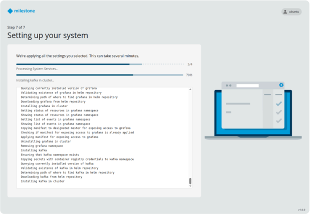
Processing Infrastructure Services

47%

Installing metallb in cluster

```
Comparing existing config map with ca-tls certificate on disk
Updating certificate in config map to microk8s if different from certificate on disk
Updating core dns addon manifest to use image pull secrets
Updating core dns addon manifest to not use image pull secrets
Copying updated core dns addon manifest to configuration file
Disable core dns addon if there were changes to it configuration; it will be enabled again if needed
now using the new configuration
Getting current state of addons
Parsing current state of addons
Enabling addons
Disabling addons
Lookup all helm releases that are have status set to pending or failed
Uninstall helm releases that are either pending or failed
Installing metallb
Ensuring that metallb namespace exists
Copying secrets with container registry credentials to metallb namespace
Querying currently installed version of metallb
Validating existence of metallb in helm repository
Determining path of where to find metallb in helm repository
Downloading metallb from helm repository
DEBUG!
Installing metallb in cluster
```

step 2



Step 7 of 7

Setting up your system

We're applying all the settings you selected. This can take several minutes.

3/4

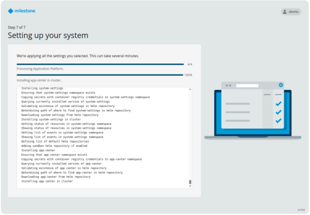
Processing System Services..

70%

Installing kafka in cluster..

```
Querying currently installed version of grafana
Validating existence of grafana in helm repository
Determining path of where to find grafana in helm repository
Downloading grafana from helm repository
Installing grafana in cluster
Getting status of resources in grafana namespace
Showing status of resources in grafana namespace
Getting list of events in grafana namespace
Showing list of events in grafana namespace
Copying manifest to designated master for exposing access to grafana
Checking if manifest for exposing access to grafana is already applied
Applying manifest for exposing access to grafana
Uninstalling grafana in cluster
Removing grafana namespace
Installing kafka
Ensuring that kafka namespace exists
Copying secrets with container registry credentials to kafka namespace
Querying currently installed version of kafka
Validating existence of kafka in helm repository
Determining path of where to find kafka in helm repository
Downloading kafka from helm repository
Installing kafka in cluster
```

step 3



Step 7 of 7

Setting up your system

We're applying all the settings you selected. This can take several minutes.

4/4

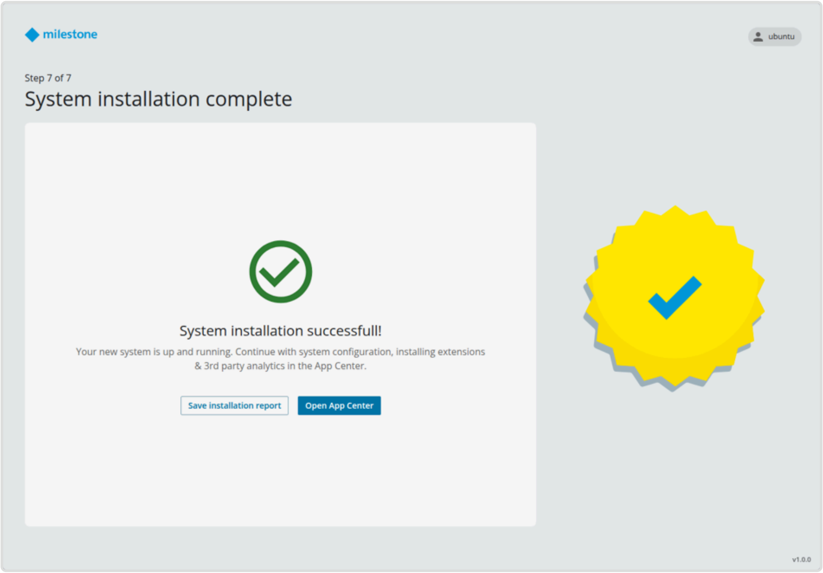
Processing Application Platform..

100%

Installing app-center in cluster..

```
Installing system-settings
Ensuring that system-settings namespace exists
Copying secrets with container registry credentials to system-settings namespace
Querying currently installed version of system-settings
Validating existence of system-settings in helm repository
Determining path of where to find system-settings in helm repository
Downloading system-settings from helm repository
Installing system-settings in cluster
Getting status of resources in system-settings namespace
Showing status of resources in system-settings namespace
Getting list of events in system-settings namespace
Showing list of events in system-settings namespace
Defining list of default helm repositories
Adding sandbox helm repository if enabled
Installing app-center
Ensuring that app-center namespace exists
Copying secrets with container registry credentials to app-center namespace
Querying currently installed version of app-center
Validating existence of app-center in helm repository
Determining path of where to find app-center in helm repository
Downloading app-center from helm repository
Installing app-center in cluster
```

almost done



Step 7 of 7

System installation complete

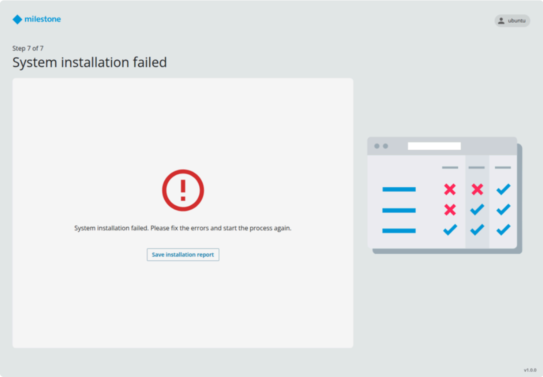
System installation successfull!

Your new system is up and running. Continue with system configuration, installing extensions & 3rd party analytics in the App Center.

Success!

Next Steps

- Click **Open App Center** to open the App Center in the default browser.
- Consult the Administration Guide for instructions on managing and upgrading your Runtime Environment.



Step 7 of 7

System installation failed

System installation failed. Please fix the errors and start the process again.

Oops! Something went wrong! Time to download the logs.

Install Sample Application (Httpd WebGL)

To validate that your Runtime Environment has been installed correctly, we provide a simple sample application. Follow these steps to deploy and verify **Httpd WebGL** from the App Center:

1. **Find & Install**
   - In the App Center's **Find new apps** section, locate **Httpd WebGL** and click **Install**.
2. **Confirm Installation**
   - Switch to the **Installed Apps** tab and ensure that **Httpd WebGL** appears in the list.
3. **Verify Service**
   - Open a new browser tab and navigate to `http://yourclusteripaddress/httpd-webgl`
4. **Observe Output**
   - You should see a rotating 3D cube rendered by WebGL. This confirms the application (application) is running as expected.
5. **Completion**
   - Once you see the 3D cube, you're done!

Thank you for using the `Milestone Runtime Installation Wizard`. Should you encounter any issues, please refer to the Troubleshooting section or contact Milestone Support.

Installing Milestone Runtime Environment (CLI)

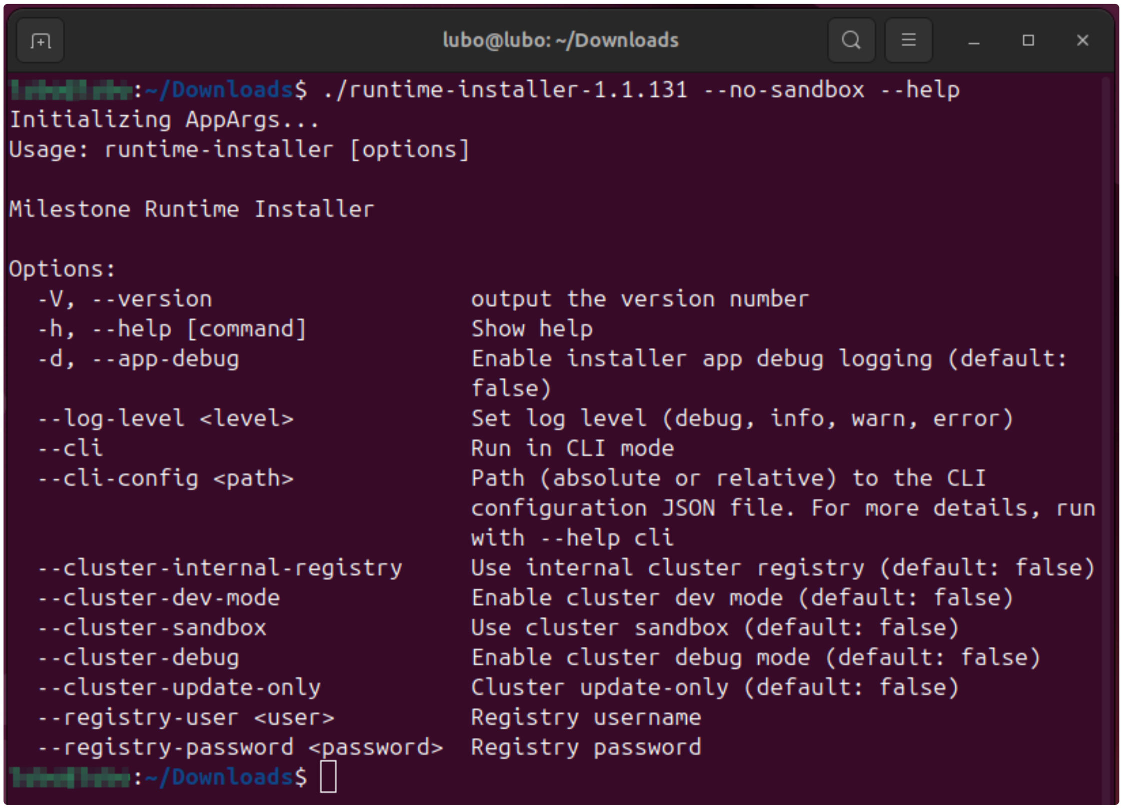
You can use the same installer file to install the App Center in CLI mode using Terminal in Linux or PowerShell in Windows.

Runtime installer help section

The CLI mode of the Runtime Installer comes with a help section. By running the Linux app image or Windows executable with the **--help** command, the installer prints out a help section in the command line i.e.:

1. Linux Terminal

   Running the app image in the Linux Terminal also requires the **--no-sandbox** argument to be set.

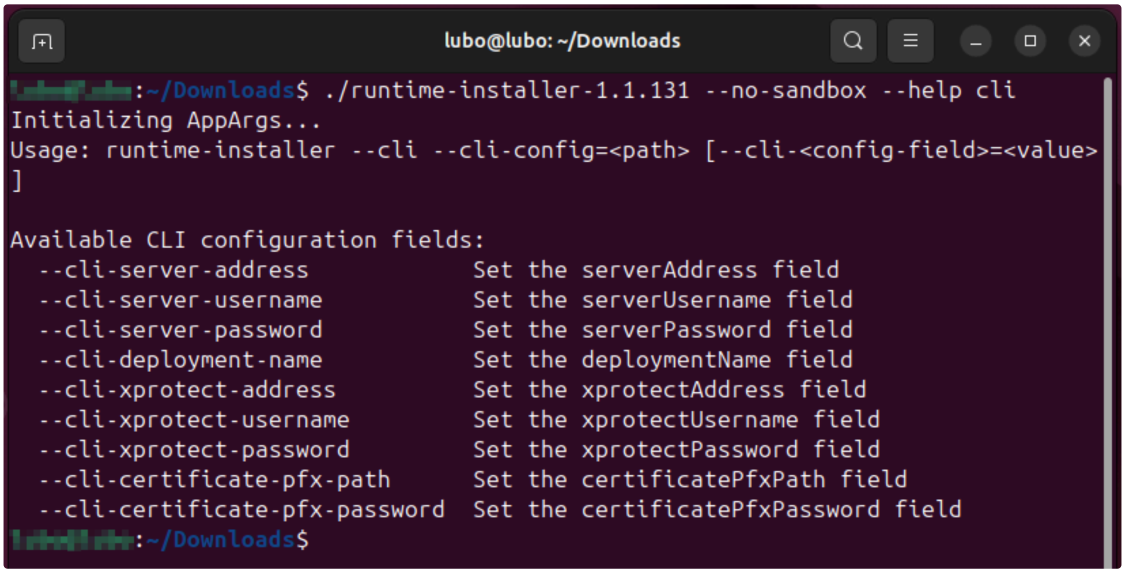


```
[redacted]:~/Downloads$ ./runtime-installer-1.1.131 --no-sandbox --help
Initializing AppArgs...
Usage: runtime-installer [options]

Milestone Runtime Installer

Options:
  -V, --version                  output the version number
  -h, --help [command]           Show help
  -d, --app-debug                Enable installer app debug logging (default:
                                 false)
  --log-level <level>            Set log level (debug, info, warn, error)
  --cli                          Run in CLI mode
  --cli-config <path>            Path (absolute or relative) to the CLI
                                 configuration JSON file. For more details, run
                                 with --help cli
  --cluster-internal-registry    Use internal cluster registry (default: false)
  --cluster-dev-mode             Enable cluster dev mode (default: false)
  --cluster-sandbox              Use cluster sandbox (default: false)
  --cluster-debug                Enable cluster debug mode (default: false)
  --cluster-update-only          Cluster update-only (default: false)
  --registry-user <user>         Registry username
  --registry-password <password>  Registry password
[redacted]:~/Downloads$ 
```

   There is also a help section specifically to the CLI mode. You can access it by appending **cli** after **--help** i.e.:



```
[redacted]:~/Downloads$ ./runtime-installer-1.1.131 --no-sandbox --help cli
Initializing AppArgs...
Usage: runtime-installer --cli --cli-config=<path> [--cli-<config-field>=<value>
]

Available CLI configuration fields:
  --cli-server-address           Set the serverAddress field
  --cli-server-username          Set the serverUsername field
  --cli-server-password          Set the serverPassword field
  --cli-deployment-name          Set the deploymentName field
  --cli-xprotect-address         Set the xprotectAddress field
  --cli-xprotect-username        Set the xprotectUsername field
  --cli-xprotect-password        Set the xprotectPassword field
  --cli-certificate-pfx-path     Set the certificatePfxPath field
  --cli-certificate-pfx-password  Set the certificatePfxPassword field
[redacted]:~/Downloads$
```

2. Windows PowerShell

⚠

> Currently in Windows, the Runtime installer in CLI mode accepts arguments, but does not display their output. This is a known issue and will be resolved in a future release.

Linux Installation (recommended)

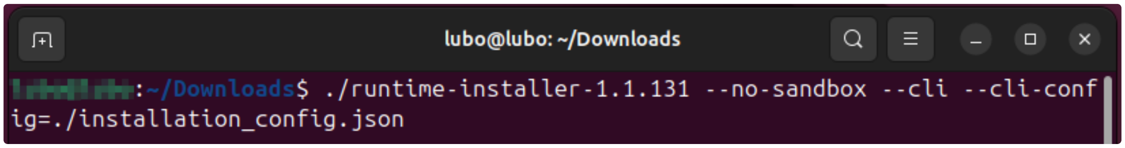
Using a configuration file

1. Create a **.json** configuration file with the following contents and store it somewhere on your computer (i.e. installation_config.json):

```
{
  "deploymentName": "Deployment Name",
  "serverAddress": "192.168.0.1",
  "serverUsername": "someUsername",
  "serverPassword": "somePassword",
  "xprotectAddress": "192.168.0.2",
  "xprotectUsername": "anotherUsername",
  "xprotectPassword": "anotherPassword",
  "certificatePfxPath": "./certificate.pfx",
  "certificatePfxPassword": "123456"
}
```

   The **certificatePfxPath** and **certificatePfxPassword** fields are optional, however if a **certificatePfxPath** is provided, the **certificatePfxPassword** field also becomes mandatory.

2. Open the Linux Terminal and run the installer app image with the following CLI arguments:

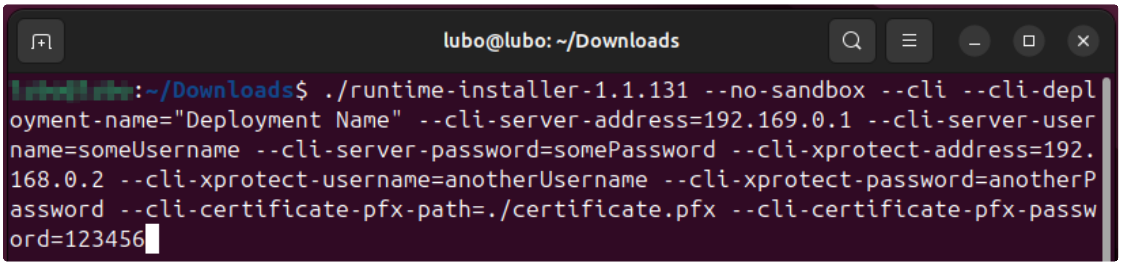


```
[redacted]:~/Downloads$ ./runtime-installer-1.1.131 --no-sandbox --cli --cli-conf
ig=./installation_config.json
```

   The **--cli-config** argument should point to the location of your configuration file either using an absolute or relative file path.

Using CLI arguments instead of a configuration file

Open the Linux Terminal and run the installer app image with the following CLI arguments:



```
[redacted]:~/Downloads$ ./runtime-installer-1.1.131 --no-sandbox --cli --cli-depl
oyment-name="Deployment Name" --cli-server-address=192.169.0.1 --cli-server-user
name=someUsername --cli-server-password=somePassword --cli-xprotect-address=192.
168.0.2 --cli-xprotect-username=anotherUsername --cli-xprotect-password=anotherP
assword --cli-certificate-pfx-path=./certificate.pfx --cli-certificate-pfx-passw
ord=123456
```

The **--cli-certificate-pfx-path** and **--cli-certificate-pfx-password** arguments are optional, however if a **--cli-certificate-pfx-path** is provided, the **--cli-certificate-pfx-password** argument also becomes mandatory.

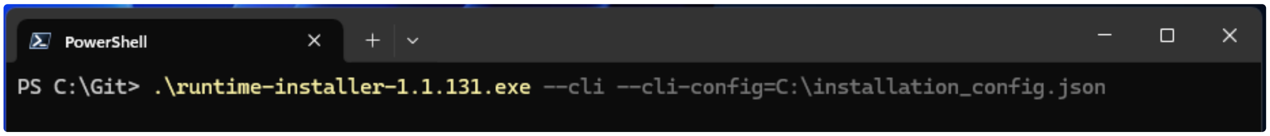
Windows Installation

⚠ Running the Windows installer in CLI mode will execute the installation process but does not yield any output to the PowerShell console. This is an issue we are aware of and will fix in a future release. If you try to run the installer in Windows using the CLI mode, be aware that you are doing so blindly until a new installer with the appropriate fix has been released.

Using a configuration file

1. Create a **.json** configuration file (i.e. installation_config.json) with the contents shown in the Linux example and store it somewhere on your computer.

2. Open Windows PowerShell and run the installer executable with the following CLI arguments:

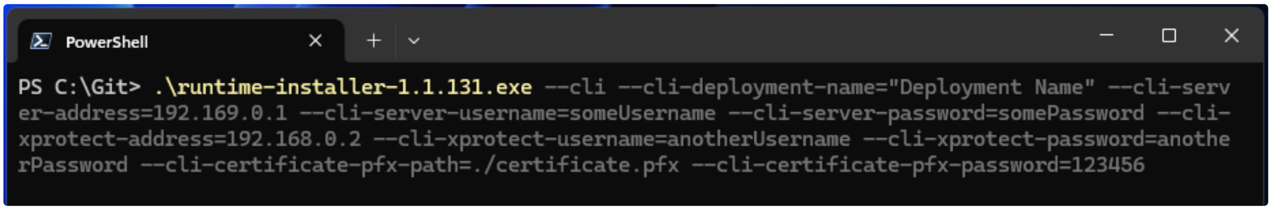


```
PS C:\Git> .\runtime-installer-1.1.131.exe --cli --cli-config=C:\installation_config.json
```

The **--cli-config** argument should point to the location of your configuration file either using an absolute or relative file path.

Using CLI arguments instead of a configuration file

Open Windows PowerShell and run the installer executable with the following CLI arguments:



```
PS C:\Git> .\runtime-installer-1.1.131.exe --cli --cli-deployment-name="Deployment Name" --cli-serv
er-address=192.169.0.1 --cli-server-username=someUsername --cli-server-password=somePassword --cli-
xprotect-address=192.168.0.2 --cli-xprotect-username=anotherUsername --cli-xprotect-password=anothe
rPassword --cli-certificate-pfx-path=./certificate.pfx --cli-certificate-pfx-password=123456
```

The **--cli-certificate-pfx-path** and **--cli-certificate-pfx-password** arguments are optional, however if a **--cli-certificate-pfx-path** is provided, the **--cli-certificate-pfx-password** argument also becomes mandatory.

Integrate App Center in Management Client

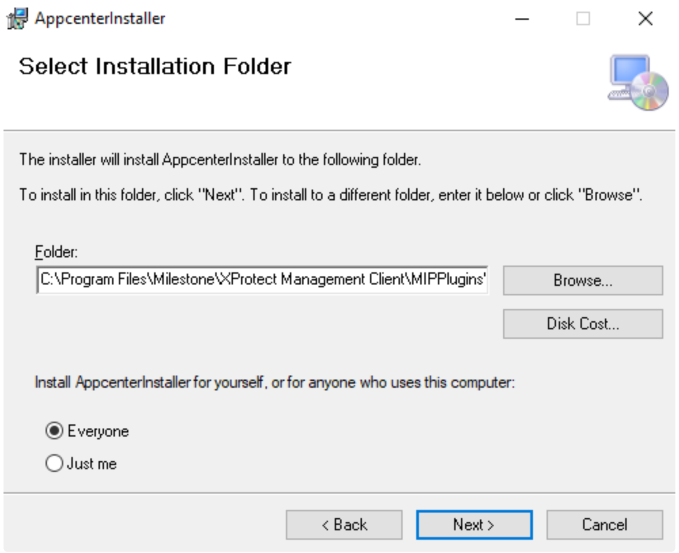
Prerequisites

- **XProtect Management Client** – A working installation of the XProtect Management Client.

- **Microsoft WebView2** – Required by the App Center plugin. If WebView2 is not present, download the Evergreen Bootstrapper from Microsoft’s website: https://developer.microsoft.com/en-us/Microsoftedge/webview2/?form=MA13LH#download

Download & Install the App Center Plugin

1. On the same machine where the XProtect Management Client is installed, download **AppcenterInstaller.msi** from here.
2. Run the **AppcenterInstaller.msi** package. By default, it will install into the Management Client’s plugin folder.
   - If your Management Client resides in a custom directory, click **Browse** during installation and select the correct path.
3. When installation completes, **restart** the XProtect Management Client.



AppcenterInstaller

Select Installation Folder

The installer will install AppcenterInstaller to the following folder.

To install in this folder, click "Next". To install to a different folder, enter it below or click "Browse".

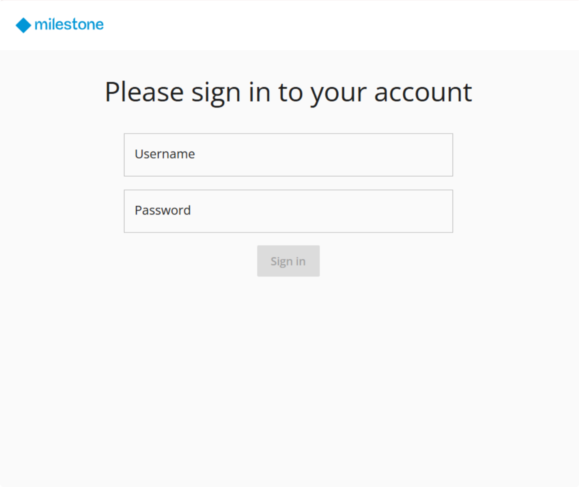
Folder:
C:\Program Files\Milestone\XProtect Management Client\MIPPlugins\

Install AppcenterInstaller for yourself, or for anyone who uses this computer:

● Everyone
○ Just me

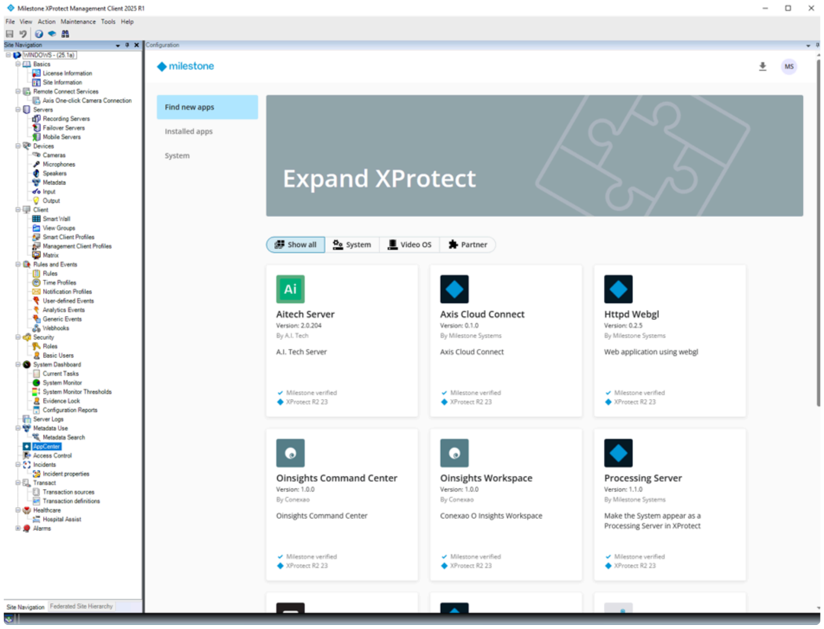
Access the App Center Node

1. Open the XProtect Management Client.
2. In the **Navigation Tree**, locate the new **App Center** node under **Plugins**.
3. Click **App Center** to launch the interface.
4. When prompted, log in with a basic admin Management Client user



Please sign in to your account

You can now manage and deploy services directly from within your XProtect Management Client—just as you would when using App Center in a web browser.



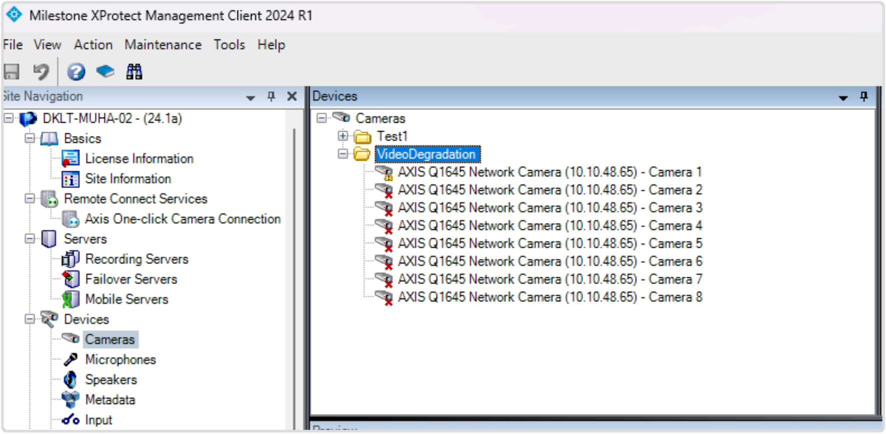
Install Video Degradation App

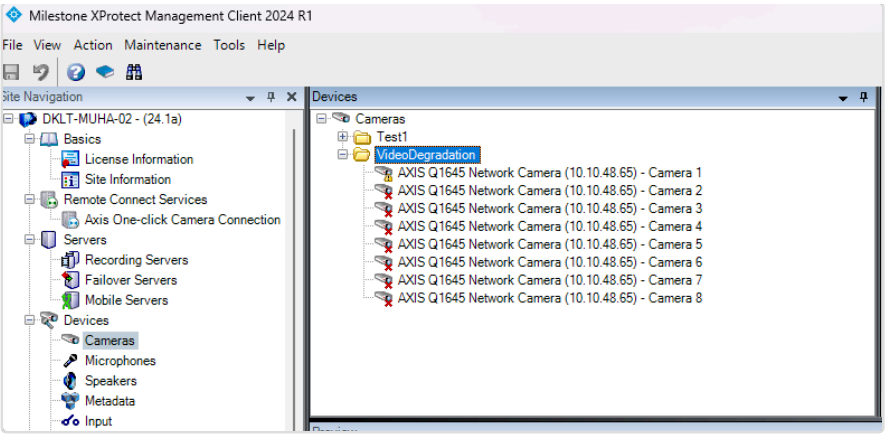
Prerequisites

- **App Center Plugin** – Must be installed before proceeding with the Video Degradation App installation.
- **Processing Server** – Required for video analysis.

Download & Install the Processing Server

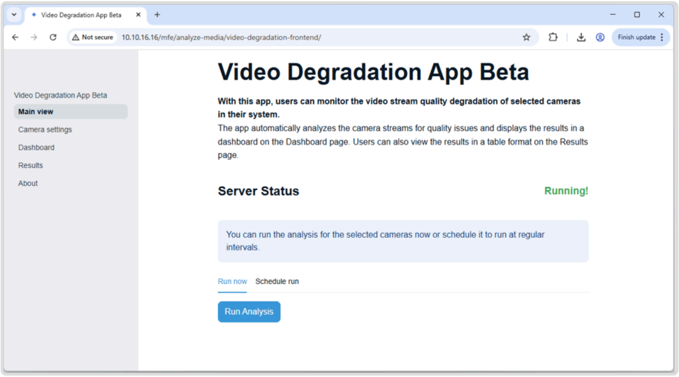
1. Open the **XProtect Management Client**.
2. In the **Navigation Tree**, go to **App Center**.
3. Install the **Processing Server**.
4. Install **Video Degradation App** once the processing server is installed.
5. Create a Camera group named **VideoDegradation.**
6. Add cameras to the VideoDegrdation group.


Milestone XProtect Management Client 2024 R1
File View Action Maintenance Tools Help
Site Navigation
DKLT-MUHA-02 - (24.1a)
Basics
License Information
Site Information
Remote Connect Services
Axis One-click Camera Connection
Servers
Recording Servers
Failover Servers
Mobile Servers
Devices
Cameras
Microphones
Speakers
Metadata
Input
Devices
Cameras
Test1
VideoDegradation
AXIS Q1645 Network Camera (10.10.48.65) - Camera 1
AXIS Q1645 Network Camera (10.10.48.65) - Camera 2
AXIS Q1645 Network Camera (10.10.48.65) - Camera 3
AXIS Q1645 Network Camera (10.10.48.65) - Camera 4
AXIS Q1645 Network Camera (10.10.48.65) - Camera 5
AXIS Q1645 Network Camera (10.10.48.65) - Camera 6
AXIS Q1645 Network Camera (10.10.48.65) - Camera 7
AXIS Q1645 Network Camera (10.10.48.65) - Camera 8


Milestone XProtect Management Client 2024 R1
File View Action Maintenance Tools Help
Site Navigation
DKLT-MUHA-02 - (24.1a)
Basics
License Information
Site Information
Remote Connect Services
Axis One-click Camera Connection
Servers
Recording Servers
Failover Servers
Mobile Servers
Devices
Cameras
Microphones
Speakers
Metadata
Input
Devices
Cameras
Test1
VideoDegradation
AXIS Q1645 Network Camera (10.10.48.65) - Camera 1
AXIS Q1645 Network Camera (10.10.48.65) - Camera 2
AXIS Q1645 Network Camera (10.10.48.65) - Camera 3
AXIS Q1645 Network Camera (10.10.48.65) - Camera 4
AXIS Q1645 Network Camera (10.10.48.65) - Camera 5
AXIS Q1645 Network Camera (10.10.48.65) - Camera 6
AXIS Q1645 Network Camera (10.10.48.65) - Camera 7
AXIS Q1645 Network Camera (10.10.48.65) - Camera 8

7. Once installation is complete, access the app at:

```
http://<your-system-ip>/mfe/analyze-media/video-degradation-frontend/
```


Video Degradation App Beta
Not secure
10.10.16.16/mfe/analyze-media/video-degradation-frontend/
Finish update
Video Degradation App Beta
Main view
Camera settings
Dashboard
Results
About
Run now
Schedule run
Run Analysis

Video Degradation App Beta

With this app, users can monitor the video stream quality degradation of selected cameras in their system.
The app automatically analyzes the camera streams for quality issues and displays the results in a dashboard on the Dashboard page. Users can also view the results in a table format on the Results page.

Server Status Running!

You can run the analysis for the selected cameras now or schedule it to run at regular intervals.

Using the Video Degradation App

1. Navigate to the **Cameras Setting** page and activate the camera(s) you want to check.
   - The app retrieves the list of cameras from **AI Bridge**
2. Return to the **Main** page and click **Run Inference**.
   - The process takes less than a minute, but a short delay may occur.
3. Once complete, explore the **Dashboard** and **Results** pages to review findings.
4. On the **Results** page, you can export all data to CSV format.
   - The CSV includes all collected data, not just what is visible in the table.

Troubleshooting

Should you encounter any issues during installation or operation, this section provides guidance to help you identify and resolve common errors. Each entry outlines the symptom, possible cause, and recommended solution so you can quickly restore full functionality. If you do not find your specific issue listed, please consult the Milestone Support portal for further assistance.

Log File Locations

To assist with troubleshooting, the installer writes detailed logs that capture every action and error. Please refer to the appropriate path below to locate the `log.txt` file:

Linux:

```
~/.config/runtime-installer/logs/log.txt
```

Windows:

```
%appdata%\runtime-installer\logs\log.txt
```

When contacting Milestone Support, please attach the relevant `log.txt` file so we can diagnose any issues more effectively