RAYVENTORY®

Technology Asset Inventory

Installation Guide RayVentory Data Hub 12.4

•rayNET

Copyright © Raynet GmbH (Germany, Paderborn HRB 3524). All rights reserved.
Complete or partial reproduction, adaptation, or translation without prior written permission is prohibited.

Installation Guide RayVentory Data Hub RayVentory Data Hub 12.4

Raynet and RayFlow are trademarks or registered trademarks of Raynet GmbH protected by patents in European Union, USA and Australia, other patents pending. Other company names and product names are trademarks of their respective owners and are used to their credit.

The content of this document is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Raynet GmbH. Raynet GmbH assumes no responsibility or liability for any errors or inaccuracies that may appear in this document. All names and data used in examples are fictitious unless otherwise noted.

Any type of software or data file can be packaged for software management using packaging tools from Raynet or those publicly purchasable in the market. The resulting package is referred to as a Raynet package. Copyright for any third party software and/or data described in a Raynet package remains the property of the relevant software vendor and/or developer. Raynet GmbH does not accept any liability arising from the distribution and/or use of third party software and/or data described in Raynet packages. Please refer to your Raynet license agreement for complete warranty and liability information.

Raynet GmbH Germany
See our website for locations.

www.raynet.de

Contents

Data Hub 5
- Windows 5
  - Prerequisites 5
    - MS SQL Server permissions 6
    - Installing IIS on a Windows Server 7
    - Installing Web Hosting Bundle 13
  - Installation 14
    - SQL Configuration 14
    - Database Server 15
    - IIS Web Application Configuration 16
    - IIS AppPool Identity 17
  - Migration 17
    - Manual Database Update 19
  - Configuration 20
    - Server Backend 20
    - Server Frontend 21
    - Logging 21
- Docker 24
  - Prerequisites 24
  - Usage 24
  - Migration 27
  - Limitations 28

Data Hub Agent 29
- Windows 29
  - Prerequisites 29
  - Installation and Configuration 30
    - Download 30
    - Installation 31
    - Registration 31
    - Authorization 32
  - Configuration 32
    - Proxy Configuration 32
  - Migration 35
- Docker 36
  - Prerequisites 36
  - Usage 36

Data Hub

Windows

The following chapter describes how to install and set-up RayVentory Data Hub on Windows.

Prerequisites

Hardware requirements

Requirements when SQL Server and RayVentory Data Hub are installed on the same machine:

- Min. 4 CPU cores
- Min. 8 GB of RAM
- Min. 20 GB of disk space

Requirements when only RayVentory Data Hub is installed on the machine:

- Min. 4 CPU cores
- Min. 4 GB of RAM
- Min. 10 GB of disk space

Software requirements

The following are the minimum software requirements for the installation and running of RayVentory Data Hub:

- Microsoft Windows Server 2012 R2 or higher
- IIS 8 or higher
- Microsoft .NET 6.0 – Windows Server Hosting Bundle (https://dotnet.microsoft.com/en-us/download/dotnet/6.0)
- Microsoft SQL Server 2016 or SQL Server Express 2016
- If RayVentory Data Hub Agent is installed on the same machine as the server, then all requirements of Data Hub Agent also apply

Note:
In order to run hosting bundles, the "Universal C Runtime" is required. Modern Windows Servers should already have it, but it may be required to download for older ones. The oldest supported OS is currently Windows Server 2012 R2. More information

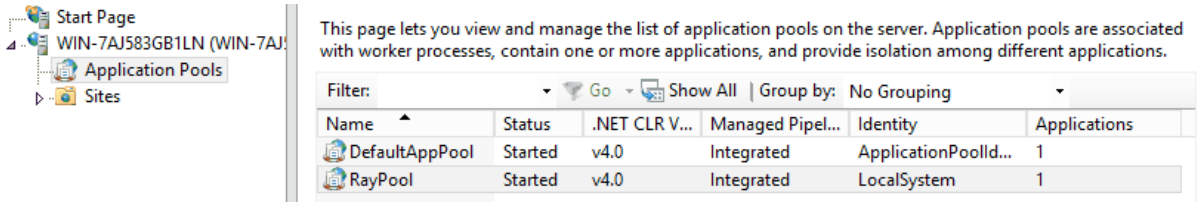
can be found here: https://support.microsoft.com/en-us/help/2999226/update-for-universal-cruntime-in-windows,

Supported web browsers

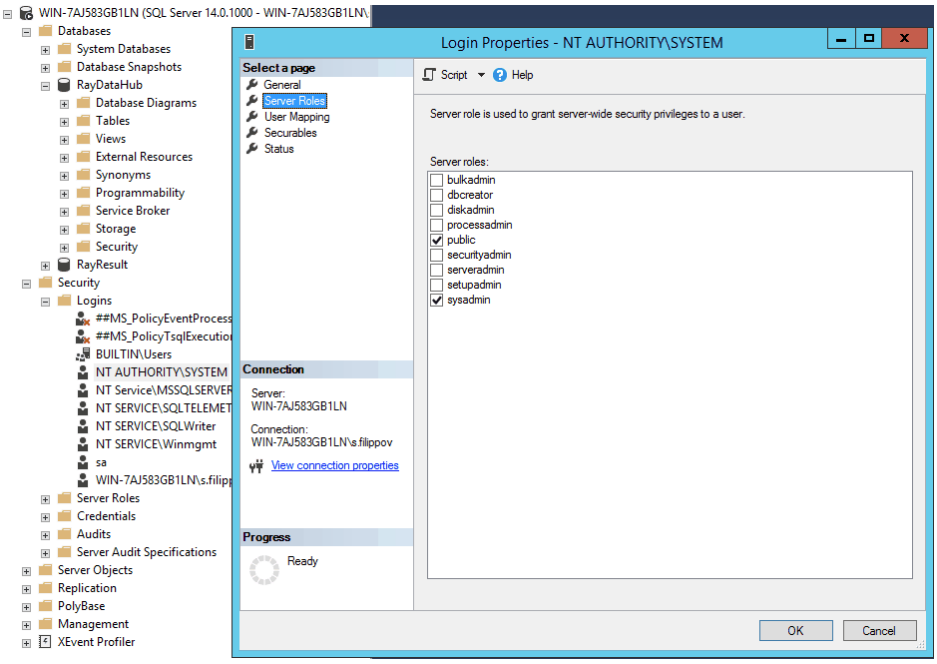
- Microsoft Edge version 80 and newer
- Mozilla Firefox version 74 and newer
- Google Chrome version 80 and newer

MS SQL Server permissions

An instance of MS SQL Server (or SQL Server Express) must be available. If installing on the same machine, permission for AppPool user should be granted.



For example

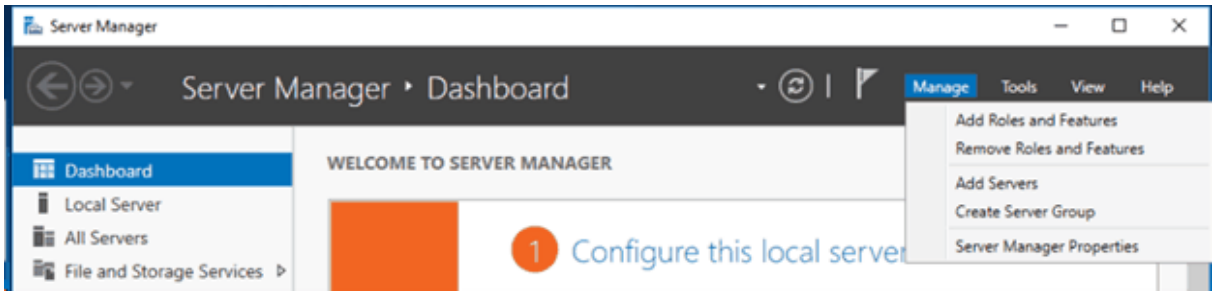


Minimum Permissions

The special service account that is used must have local administrator right. In order to install the server or to create a new tenant, `db_create` permissions are required. Furthermore, to be able to use RayVentory Data Hub the `db_owner` role for the Result and System Data Hub database is needed.

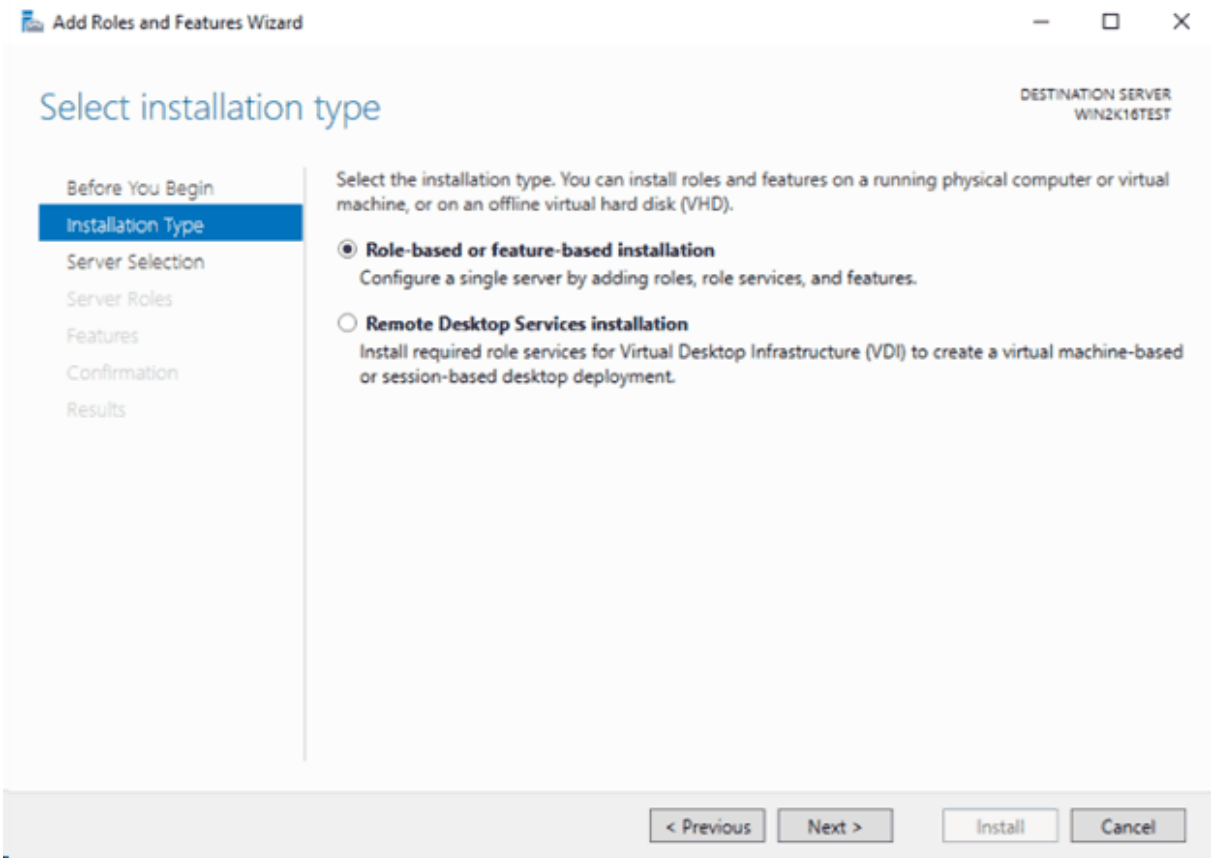
Installing IIS on a Windows Server

Either use the Server Manager and click on **Manage** -> **Add Roles and Features**.

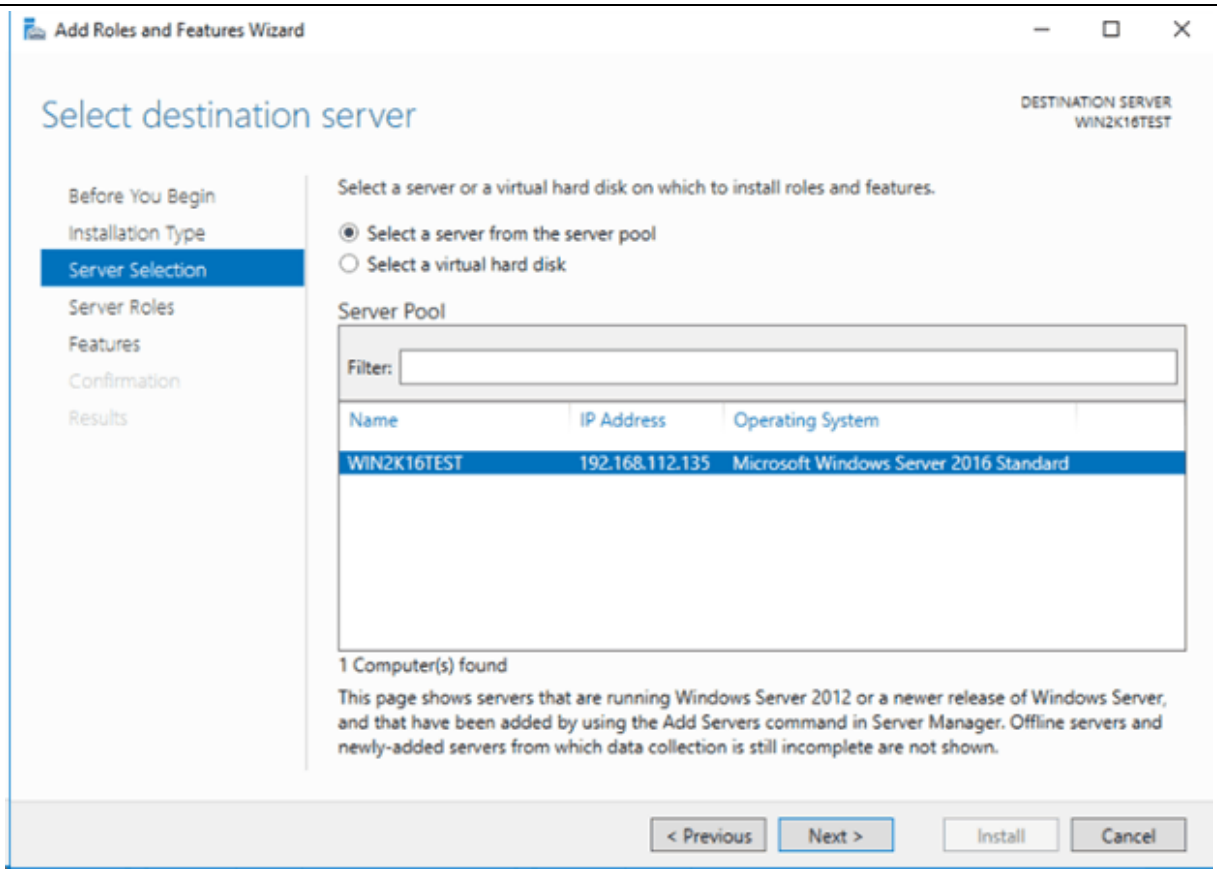


Or search for "Turn Windows Features on or off".

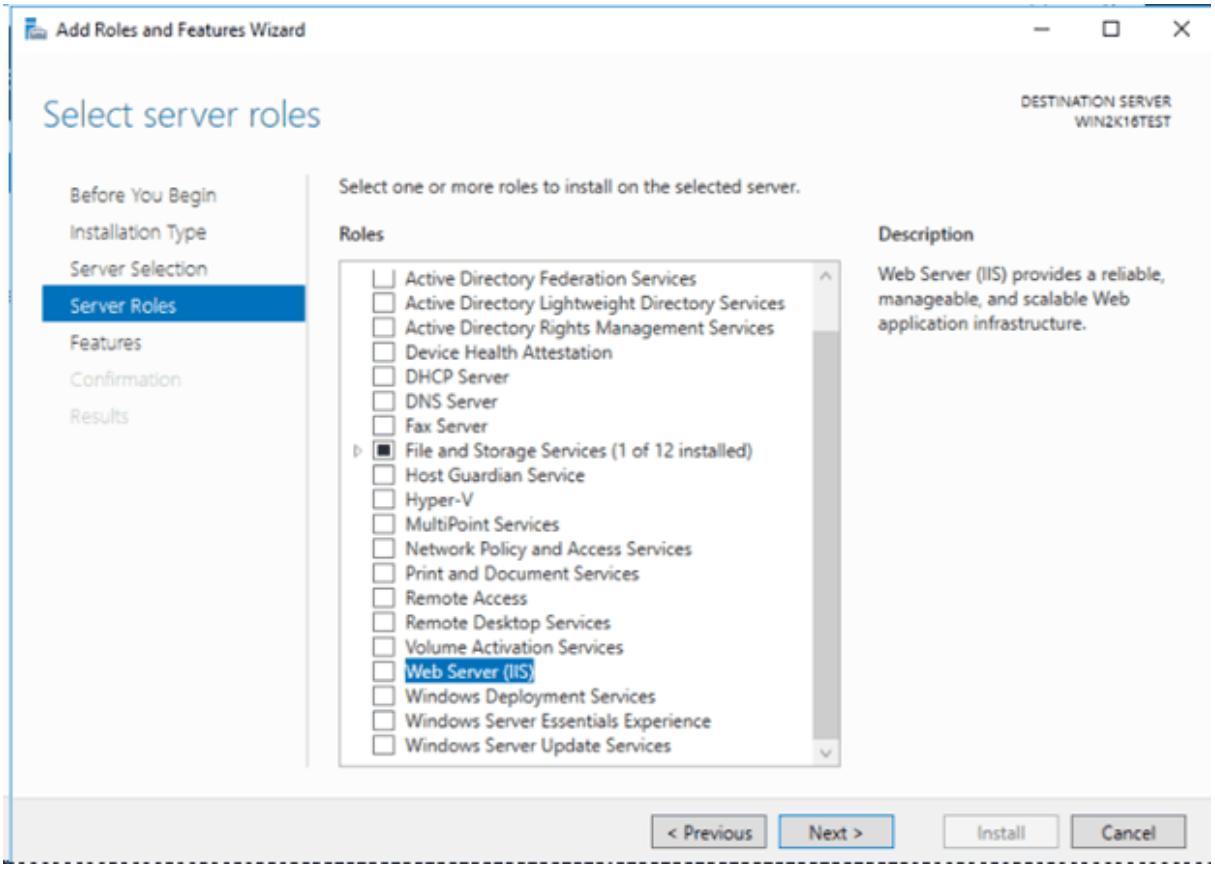
A new window opens as shown below.



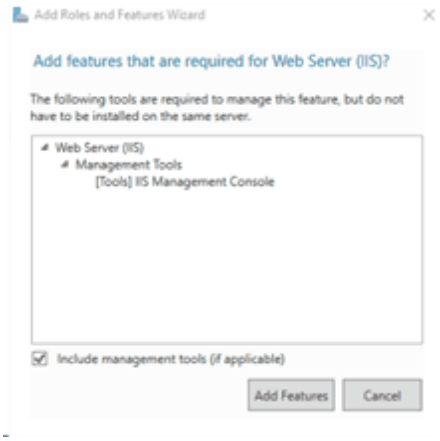
Select **Role-based or feature-based installation** and click **Next >**.



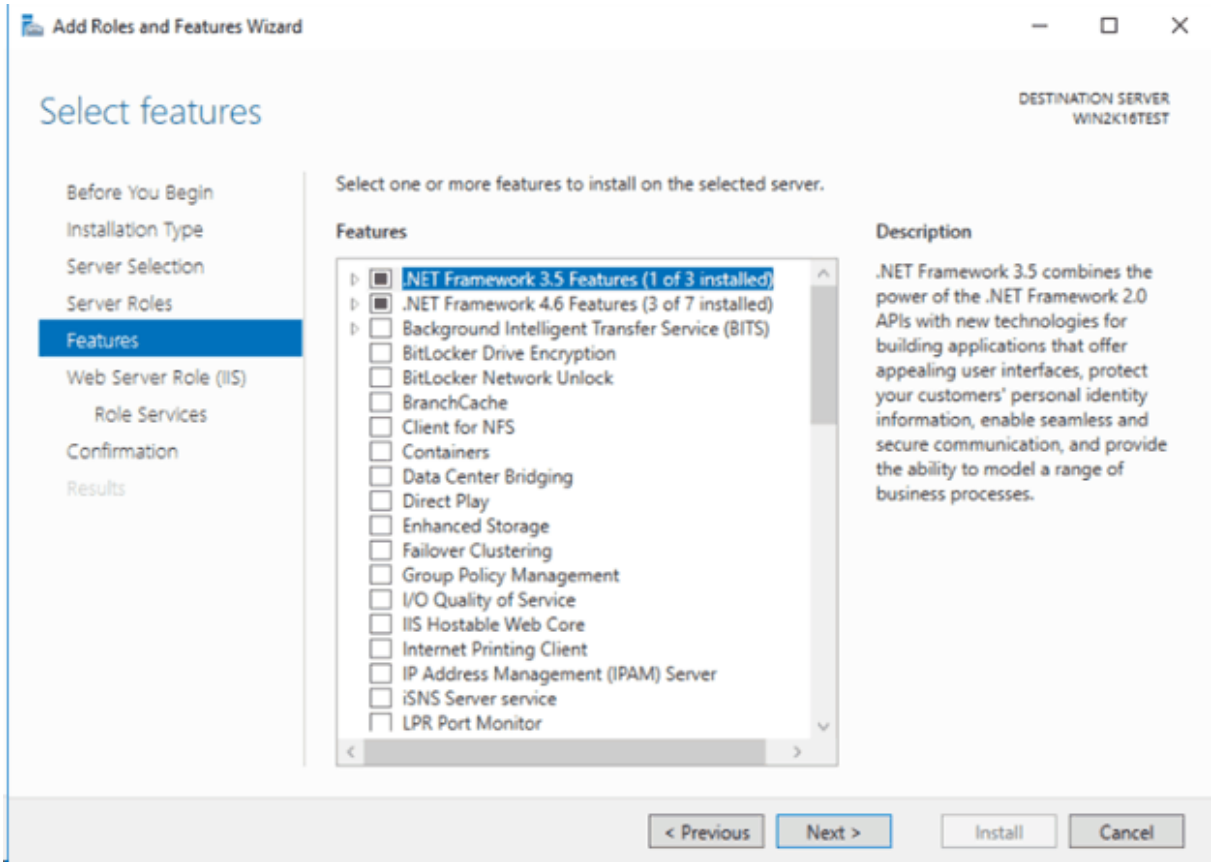
Select the Server you want the features to be installed on and click **Next >**.



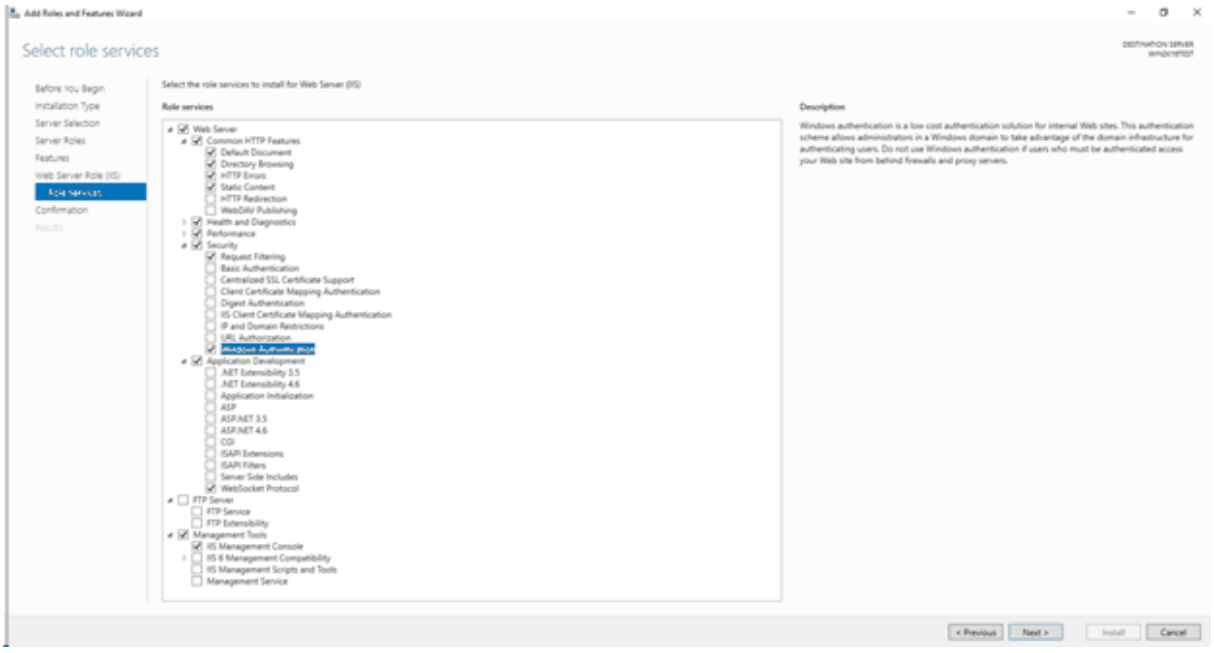
Select **Web Server (IIS)**. A new window is shown.



Click **Add Features** without changing anything and click **Next >**.



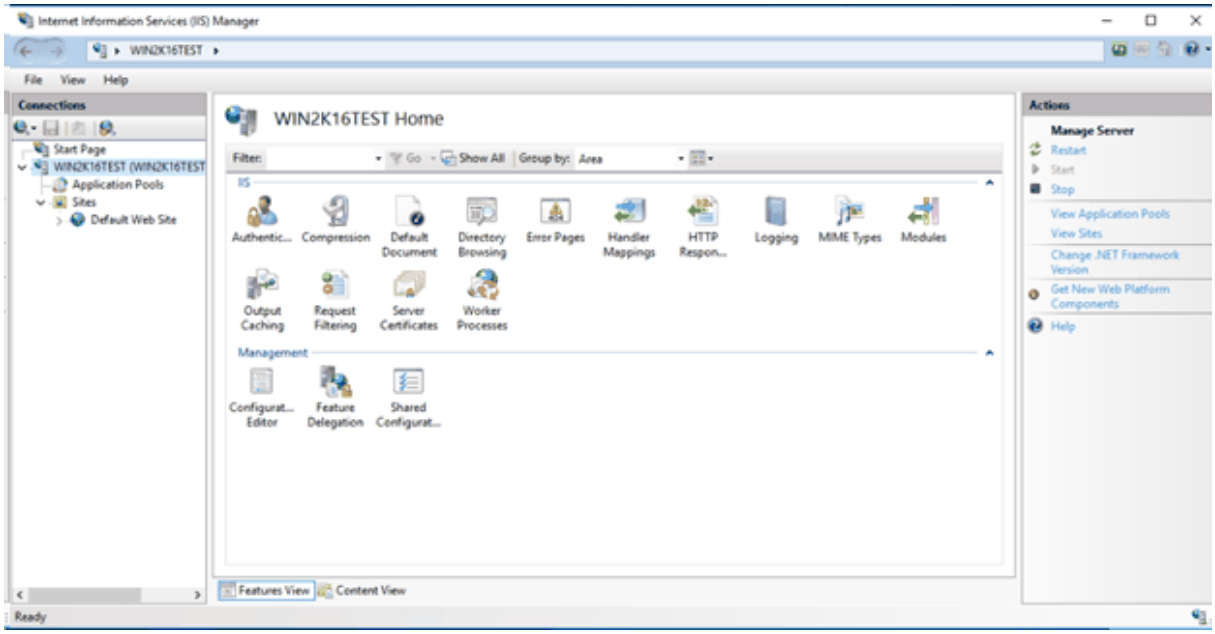
Feature configuration can be skipped, click **Next >** twice to get to the **Role Services**.



Make sure to check the services **Windows Authentication** and **WebSocket Protocol**. Leave all

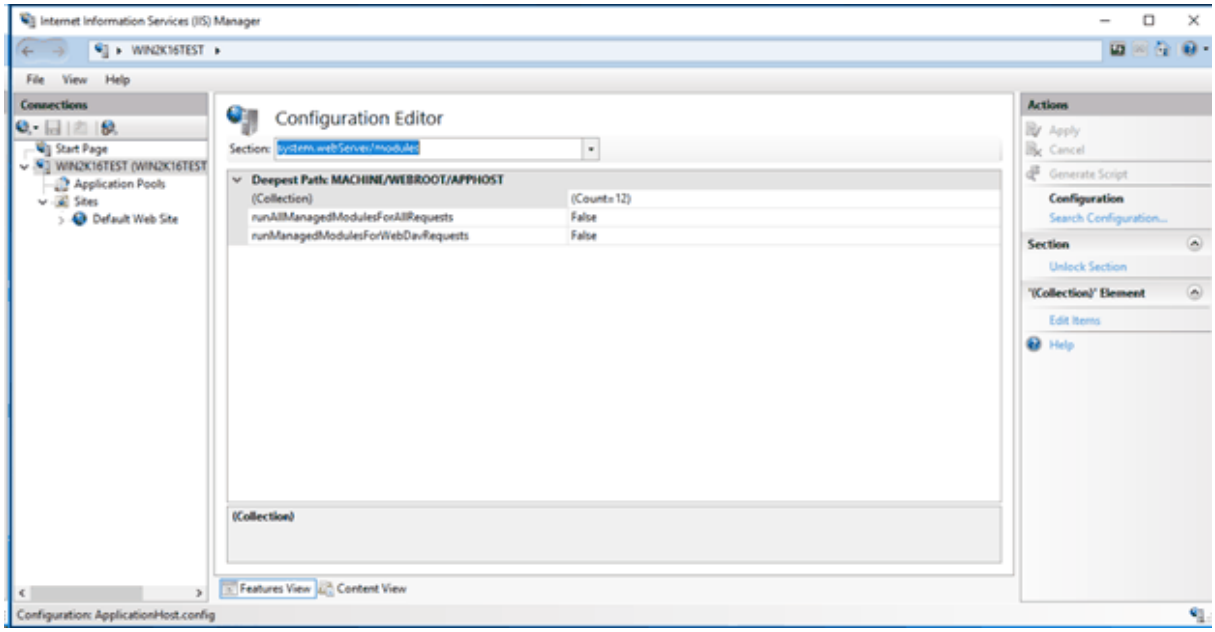
other options untouched. Click **Next >** and **Install**.

System.Webserver/modules configuration



Open the IIS Manager and select your IIS Server. From there click on Configuration Editor (below Management)

`System.Webserver/modules` configuration section must be unlocked for IIS server. This is required in order to remove the WebDav-module as it is blocking access to RayVentory Data Hub.



Installing Web Hosting Bundle

Visit the site https://dotnet.microsoft.com/en-us/download/dotnet/6.0 and click on the **Hosting Bundle** download link.

Execute the downloaded file to install the Web Hosting Bundle. Follow the steps as instructed by the install wizard.

Run apps - Runtime ⓘ

ASP.NET Core Runtime 6.0.3

The ASP.NET Core Runtime enables you to run existing web/server applications. **On Windows, we recommend installing the Hosting Bundle, which includes the .NET Runtime and IIS support.**

IIS runtime support (ASP.NET Core Module v2)
16.0.22055.3

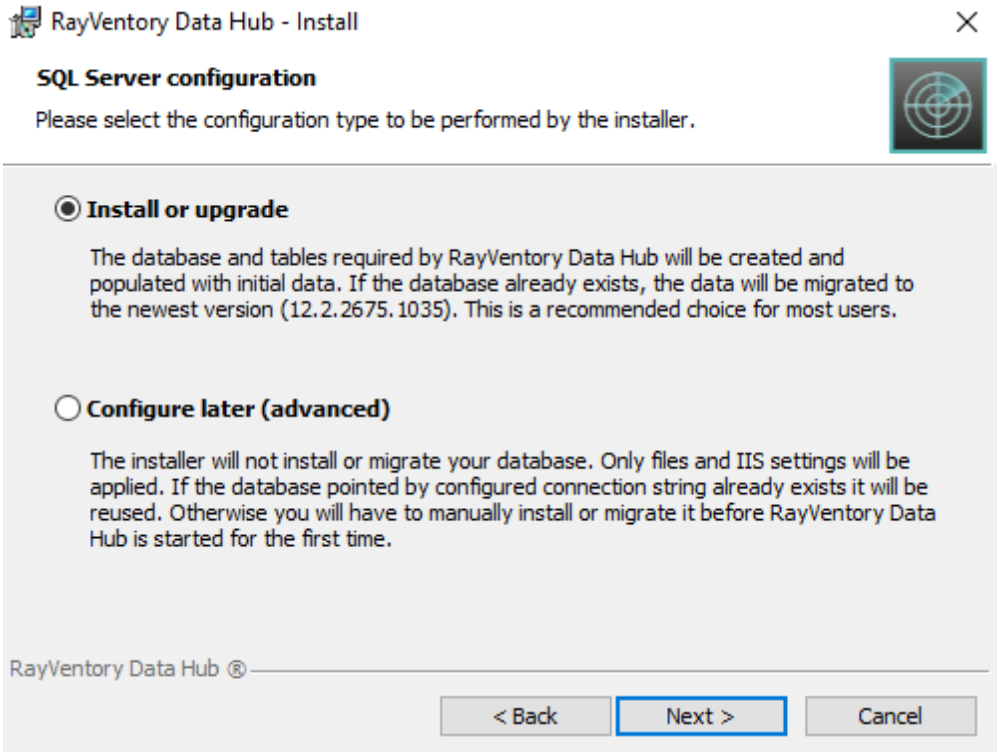
OS	Installers	Binaries
Linux	Package manager instructions	Arm32 | Arm32 Alpine | Arm64 | Arm64 Alpine | x64 | x64 Alpine
macOS		Arm64 | x64
Windows	Hosting Bundle x64 | x86	Arm64 | x64 | x86

Installation

This chapter shows how to install the Data Hub on Windows.

SQL Configuration

SQL Configuration allows to choose database default state after installation.



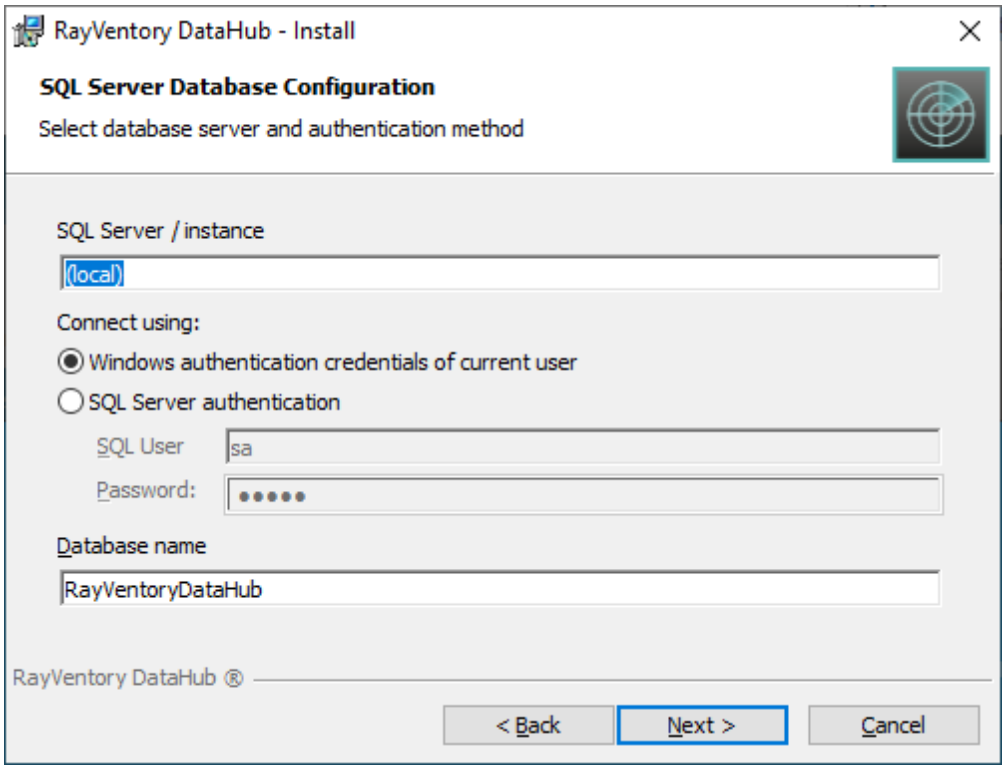
- **Install or upgrade (Recommended)**
  At first installation a new database is created with the necessary tables. If the product is upgraded to a newer version, the existing database is automatically migrated to the version required by the installation.

> **Note:**
> When upgrading from previous version 12.2: The installer will adapt your schema and data if necessary. There may be some manual adjustments required in case of reports / dashboard referencing complex tables / data sets.

- **Configure later**
  This option allows the user to perform the creation or migration of the database manually. See chapter Manual database update for more information.

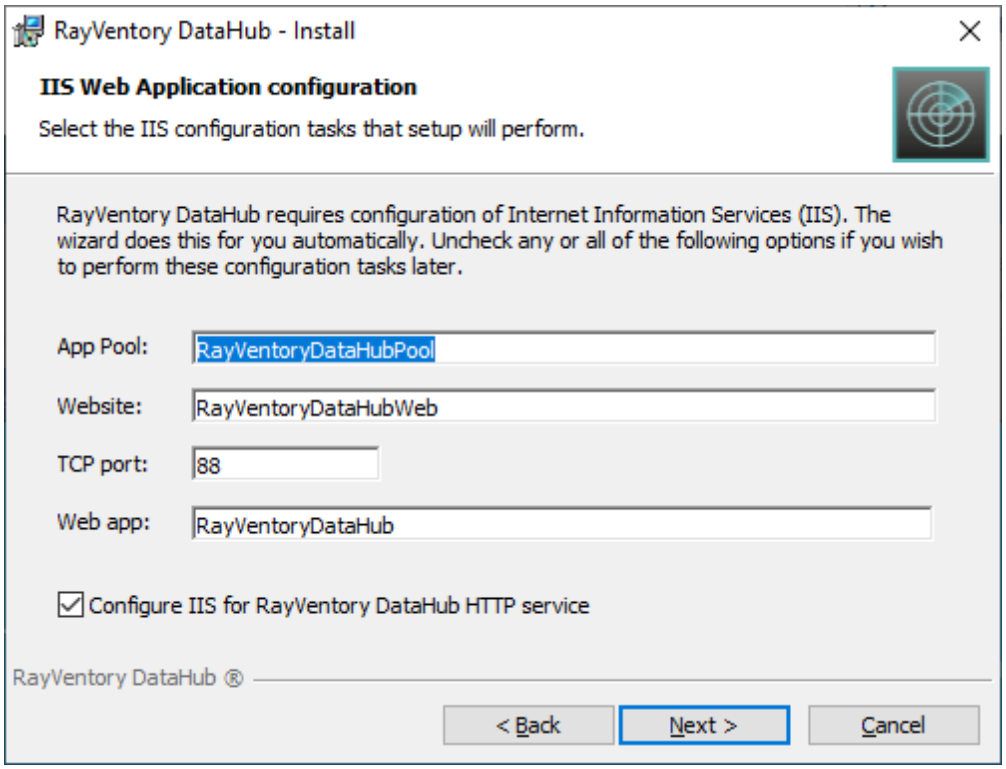
Database Server

Database server wizard page is shown only if **Install or Upgrade** SQL Configuration was selected on the previous page.



Specify the address of the server to connect to, an authentication type, and the name of the internal database for RayVentory Data Hub. When choosing Windows authentication as the authentication method, ensure that the pool user has read and write access to the configured database.

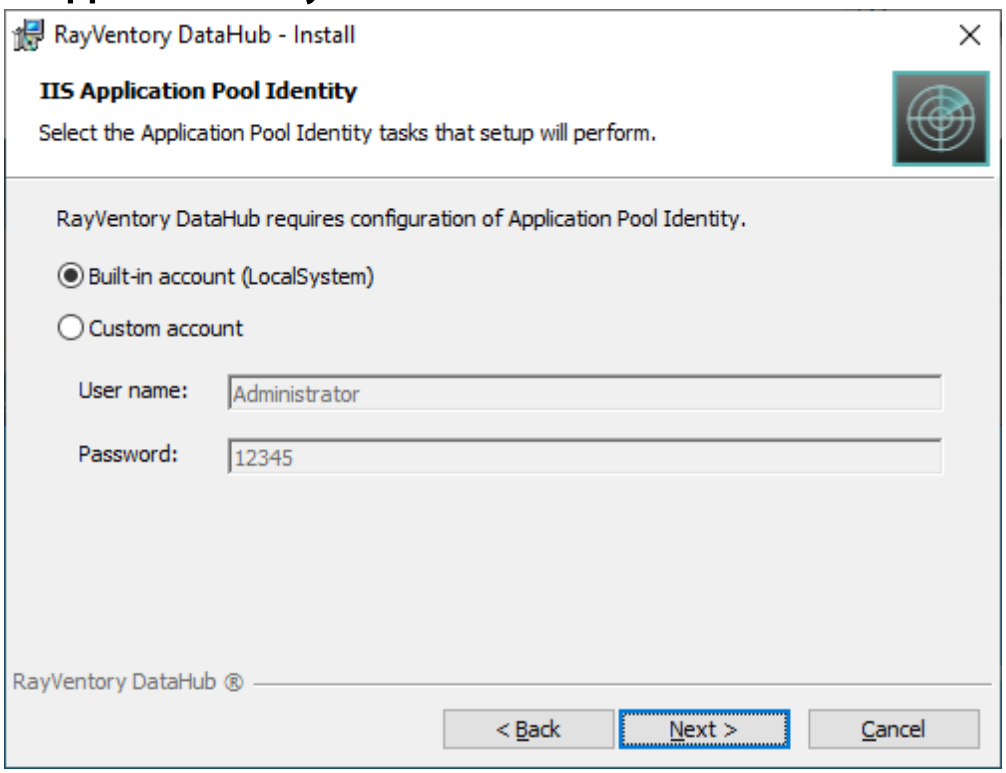
IIS Web Application Configuration



Configure the basic details for the IIS settings that will be used during the installation of RayVentory Data Hub.

Be aware:
The TCP port option is for the HTTP protocol only. The HTTPS bindings need to created after the installation using the IIS Manager.

IIS AppPool Identity



Define the user on behalf of whom the server component of the RayVentory Data Hub application will run. Make sure this user has required permissions to access the database if **Windows authentication** was selected as the authentication method for the database access.

Migration

This list shows migration paths and additional considerations:

Data Hub Backend

- Version 12.0 -> 12.4
  NOT SUPPORTED.
  Please contact Raynet for assistance.

- Version 12.1 -> 12.4
  SUPPORTED

- Version 12.2 -> 12.4
  SUPPORTED

- Version 12.3 -> 12.4
  SUPPORTED

The migration of database is supported out-of-the-box. It is recommended to uninstall the

previous version of the product (the database will stay intact) and install it again with the new installer. The database will be migrated automatically during the installation, or it can be also upgraded manually (see Manual database update for more information).

After the migration, the database will be not backward-compatible, which means that any previous instance of RayVentory Data Hub will be unable to use the same database.

Note:
During the upgrade, IIS settings will be reset to default ones. Make sure to provide the same parameters during the upgrade.

Data Migration

Reports and dashboard are not migrated automatically, and you can continue to use them. However, new version of Data Hub often come with new features and improvements, which are only available in the newer version of their respective templates. We recommend to create a copy of existing reports (using the backup functionality, available in RayVentory Data Hub 12.3), and then to import new reports and dashboards from the library.

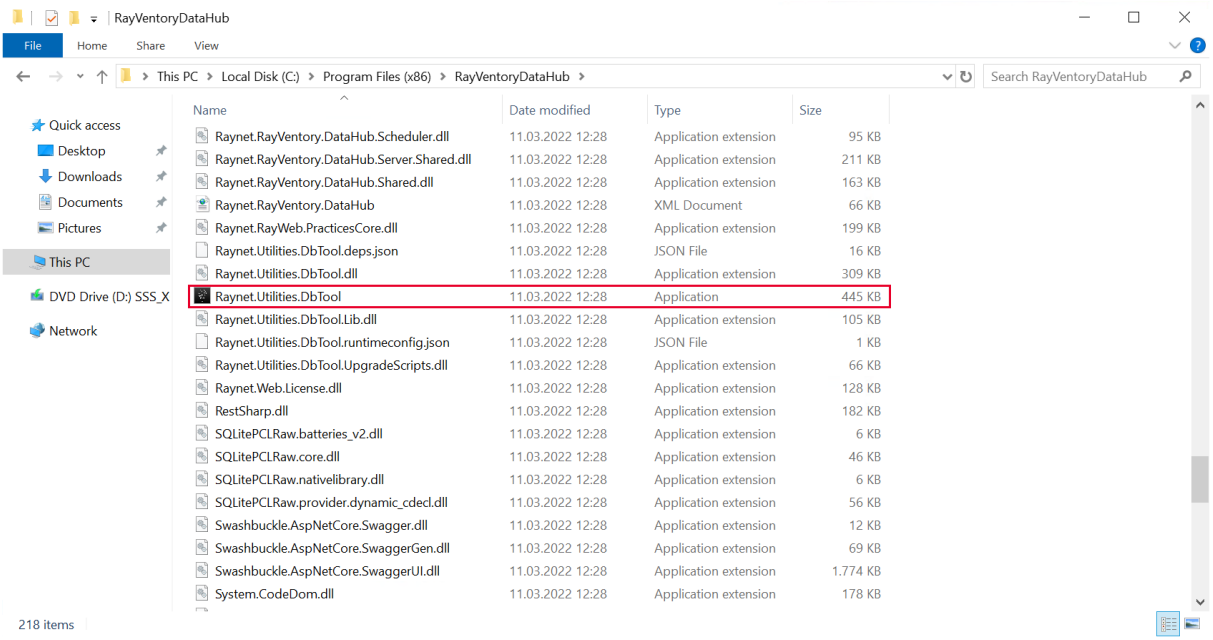
Data Hub Agent (formerly Data Collector up to 12.1)

It is recommended to use the same product version of Data Hub and Data Hub Agent. Failing to use matching version can lead to difficulties configuring and running your tasks. The agent must be updated on every machine that connects to the RayVentory Data Hub. You can check the version of the agent on the **Agents** page.

For more information about updating the Data Hub Agent, refer to the **Data Hub Agent / Migration** chapter.

Manual Database Update

RayVentory Data Hub provides a database CLI tool that can be used to create a database or migrate the database to a desired version. The CLI tool can be started from `[INSTALLDIR]\Raynet.Utilities.DbTool.exe`.



In most cases, the tool should be started with the following command line:

```
Raynet.Utilities.DbTool.exe update -t 12.4.#.#
```

where **12.4.#.#** is the current full version of the product to be installed. The tool detects the database configuration from the parent config file (`appsettings.json` in the root installation folder).

The tool has a command line interface help, shown when no matching parameter is found. Command line interface has further options, for example:

- Using custom connection strings or picking a right connection string from the configuration file
- Logging
- Checking the current version

Once started, the tool ensures that the database is set-up properly. If the database configured in the connection string does not exist, it will be created. Otherwise, the database will be migrated to the version specified via the command line argument.

The tool returns exit code 0 in case of successful operation, and non-zero if something failed.

Supported migration paths

In this version, the following scenarios are supported:

- Set-up of a new database (new version, clean install)
- Migration from version 12.1
- Migration from version 12.2
- Migration from version 12.3

Configuration

Server Backend

The settings for the backend of the server are stored in the `appsettings.json` located in the root directory of the RayVentory Data Hub installation (by default `C:\Program Files (x86)\RayVentoryDataHub` directory).

Available settings:

- `InternalDatabase` – connection string to RayVentory Data Hub private database.
- `Connection strings` – the list of "DataSources" available for creating reports and dashboards. By default only one string will be available there – the connection string to "Result" database.

Default appsettings.json

```
{
  "Logging": {
    "LogLevel": {
      "Default": "Trace"
    }
  },
  "TemporaryFilesDirectory": "",
  "TasksManagement": {
    "LogsDirectory": "",
    "DeleteLogFilesAfterDays": 30
  },
  "CsvFileProcessing": {
    "NumThreads": 3
  },
  "TaskDataTransformationProcessing": {
    "NumLocalThreads": 3,
    "CleanupThresholdInMb": 500
  },
  "TokenManagement": {
    "secret": "RayVentoryDataHubTopSecretSecret1337",
    "issuer": "Raynet GmbH",
    "audience": "RayVentory Data Hub User",
    "accessExpiration": 30,
    "refreshExpiration": 720,
    "rememberMeRefreshExpiration": 10080
```

```
  },
  "connectionStrings": {
    "System": "Server=(local);Database=RayVentoryDataHub;Trusted_Connection=True",
    "ReportDatabase": "Data Source=(local);Initial Catalog=master;
      Trusted_Connection=True"
  },
  "LdapManagement": {
    "Protocol": "LDAP",
    "Timeout": 5,
    "CustomUserFilter": "",
    "OpenLDAP_SASL_NOCANON": true,
    "OpenLDAP_AuthType": "Digest"
  },
  "AllowedHosts": "*"
}
```

Server Frontend

There is no configuration for the settings of the frontend. All necessary settings are created automatically.

Logging

The logging is performed using the *log4net* library. Its configuration is in the root directory of RayVentory Data Hub installation (by default `C:\Program Files (x86)\RayVentoryDataHub`) in a file named `log4net.config`.

- Logs are written to the `logs` directory in the root installation folder.
- By default, only warning and errors are logged.

Default log4net.config

```
<log4net debug="true">
  <appender name="Console" type="log4net.Appender.ConsoleAppender">
    <layout type="log4net.Layout.PatternLayout">
      <!-- Pattern to output the caller's file name and line number -->
      <conversionPattern value="%date %5level %logger.%method [%line] - MESSAGE:
        %message%newline %exception" />
    </layout>
  </appender>

  <appender name="AdoNetAppender" type="MicroKnights.Logging.AdoNetAppender,
    MicroKnights.Log4NetAdoNetAppender">
    <bufferSize value="1" />
    <connectionType value="System.Data.SqlClient.SqlConnection,System.Data.SqlClient,
      Version=4.0.0.0,Culture=neutral,PublicKeyToken=b77a5c561934e089" />
    <reconnectonerror value="true" />
    <commandText value="INSERT INTO SystemLog ([Date],[Thread],[Level],[Logger],
      [Message],[Exception]) VALUES (@log_date, @thread, @log_level, @logger,
      @message, @exception)" />
    <parameter>
      <parameterName value="@log_date" />
      <dbType value="DateTime" />
      <layout type="log4net.Layout.RawUtcTimeStampLayout" />
    </parameter>
```

```
    <parameter>
      <parameterName value="@thread" />
      <dbType value="String" />
      <size value="255" />
      <layout type="log4net.Layout.PatternLayout">
        <conversionPattern value="%thread" />
      </layout>
    </parameter>
    <parameter>
      <parameterName value="@log_level" />
      <dbType value="String" />
      <size value="50" />
      <layout type="log4net.Layout.PatternLayout">
        <conversionPattern value="%level" />
      </layout>
    </parameter>
    <parameter>
      <parameterName value="@logger" />
      <dbType value="String" />
      <size value="255" />
      <layout type="log4net.Layout.PatternLayout">
        <conversionPattern value="%logger" />
      </layout>
    </parameter>
    <parameter>
      <parameterName value="@message" />
      <dbType value="String" />
      <size value="4000" />
      <layout type="log4net.Layout.PatternLayout">
        <conversionPattern value="%message" />
      </layout>
    </parameter>
    <parameter>
      <parameterName value="@exception" />
      <dbType value="String" />
      <size value="2000" />
      <layout type="log4net.Layout.ExceptionLayout" />
    </parameter>
  </appender>

  <appender name="RollingFile" type="log4net.Appender.RollingFileAppender">
    <file value="logs/RayVentoryDataHubService.log" />
    <appendToFile value="true" />
    <maximumFileSize value="1000KB" />
    <maxSizeRollBackups value="2" />
    <layout type="log4net.Layout.PatternLayout">
      <param name="header" value="/****************************** LOG HEADER
        ******************************/ &#13;&#10;" />
      <param name="footer" value="/****************************** LOG FOOTER
        ******************************/ &#13;&#10;" />
      <conversionPattern value="%date %5level %logger.%method [%line] - MESSAGE:
        %message%newline %exception" />
    </layout>
  </appender>
```

```
  <root>
    <level value="OFF" />
    <appender-ref ref="AdoNetAppender" />
    <appender-ref ref="RollingFile" />
  </root>
</log4net>
```

Docker

The following chapter describes how to install and set-up RayVentory Data Hub as a docker container.

Prerequisites

Hardware requirements

Requirements when SQL Server and RayVentory Data Hub are installed on the same machine:

- Min. 4 CPU cores
- Min. 8 GB of RAM
- Min. 20 GB of disk space

Requirements when only RayVentory Data Hub is installed on the machine:

- Min. 4 CPU cores
- Min. 4 GB of RAM
- Min. 10 GB of disk space

Software requirements

In order to run this container you'll need docker installed.

- Windows
- OS X
- Linux

On Windows, make sure to use Linux Containers, WSL2 is recommended.

Usage

The default configuration consists of three containers:

- A database powered by Microsoft SQL Server Express 2019 (image `mssql/server:2019-latest`)
- A container with backend and frontend for Data Hub server (image `raynetgmbh/rayventory-datahub`)
- A container with Agent (image `raynetgmbh/rayventory-datahubagent`)

The default configuration is standalone and should work out-of-the-box.

Manual installation

DataHub Server can be installed from the following image:

`raynetgmbh/rayventory-datahub`

The following variables are available when creating the container from image `raynetgmbh/rayventory-datahub`:

Environment variable	Description and sample value
`DataHub_ConnectionStrings_`	A connection string used to connect to the program database, containing global settings and metadata. Example value: `mydatabaseserver.local,1433;Database=datahub;User ID=sa;Password=Start123`
`DataHub_ConnectionStrings_`	A base connection string to the server, where tenant databases will be stored. Bear in mind, that Data Hub takes over the creation of the database when launched for the first time. Example value: `Server=mydatabaseserver.local,1433;Initial Catalog=master;User ID=sa;Password=Start123`
`DataHub_InitialTenantId`	A GUID value, representing the default tenant ID. When started for the first time, a tenant ID with the given ID will be created. If the value is not provided, a new random GUID will be selected. Example value: `{72ba6fc2-d5fa-49ee-8281-841e762aea05}`
`BASEURL`	The base URL, under which the browser will access the front-end. This URL must not reference internal Docker services. You should also include a port, which is exposed by the Docker engine. The DataHub runs inside the Docker on port 80 - this port should be exposed externally, either as-is or using a different port number. Example value: `http://rayventory-datahub.local:80`

The following variables are available when creating the container from image `raynetgmbh/rayventory-datahubagent`:

Environment variable	Description and sample value
`DataHubAgent_DataHubUrl`	The URL of the Data Hub server. This may be an URL referring the internal service name, when the image is started from a docker compose file. Example value: `http://web:80`
`DataHubAgent_TenantId`	A GUID value, representing the tenant ID. You can get your tenant ID by opening the **Administration > Agents** page, and pressing a button to install a new agent. Example value: `{72ba6fc2-d5fa-49ee-8281-841e762aea05}`

Installation with docker-compose

The easiest way to run the image with reasonable default is to use docker-compose command.

1. Get the `docker-compose.yml` file from Git Hub repository https://github.com/RaynetEALM/RayVentoryDataHub/blob/main/docker-compose.yml.
   - Default compose file starts three containers: database, server and agent. It exposes internal port 80 and makes it available as port 81 to the host. Additionally, it defines a connection string using SQL authentication, with default user `sa` and password `Start123!@#`.
2. Adjust the content of the compose file, for example by setting custom user and password to the database. See section **Manual installation** for more information.
   - To use another SQL Server (outside of the Docker container), adjust the connection string in the **web** service, and drop the **database** service and **sql_data** volume from the definition.
   - To prevent starting a Docker-based agent, drop the **agent** service from the definition.
3. In the folder containing the downloaded definition, run the following command: `docker compose up -d`.
4. The server will be started. This may take a while.
5. Login to `http://localhost:81` and use the following credentials:

   Login: `root`
   Password: `raynet`

6. Provide the valid license for the product.
7. After log-in, change your root password and create application users.

RayVentory Data Hub is available on docker hub: https://hub.docker.com/r/raynetgmbh/rayventory-datahub.
You can use tags 12.3 (recommended) or stable to get the last 12.3 or the last stable version respectively.

Migration

This list shows migration paths and additional considerations:

Data Hub Backend

- Version 12.0 -> 12.4
  NOT SUPPORTED.
  Please contact Raynet for assistance.

- Version 12.1 -> 12.4
  SUPPORTED

- Version 12.2 -> 12.4
  SUPPORTED

- Version 12.3 -> 12.4
  SUPPORTED

  The migration of database is supported out-of-the-box. A check and required actions are performed once the server container starts.

  After the migration, the database will be not backward-compatible, which means that any previous instance of RayVentory Data Hub will be unable to use the same database.

Data Migration

Reports and dashboard are not migrated automatically, and you can continue to use them. However, new version of Data Hub often come with new features and improvements, which are only available in the newer version of their respective templates. We recommend to create a copy of existing reports (using the backup functionality, available in RayVentory Data Hub 12.3), and then to import new reports and dashboards from the library.

Data Hub Agent

It is recommended to use the same product version of Data Hub and Data Hub Agent. Failing to use matching version can lead to difficulties configuring and running your tasks. The agent must be updated on every machine that connects to the RayVentory Data Hub. You can check the version of the agent on the **Agents** page.

For more information about updating the Data Hub Agent, refer to the **Data Hub Agent / Migration** chapter.

Limitations

Data Hub Agent in Docker has several limitations in comparison to its Windows counterpart:

- It is not possible to run Active Directory data collection. The collector **Active Directory** will fail with the message, saying that the platform is not supported.
- It is not possible to run PowerShell data collection. The collector **PowerShell** will fail with the message, saying that the platform is not supported.
- It is not possible to run SCCM data collection via WMI. The collector **SCCM (WMI)** will fail with the message, saying that the platform is not supported.

Docker assigns a random host name to each started container. When the agent connects to the server, it will contain this cryptic name instead of the host name.

Data Hub Agent

The Data Hub Agent is a Windows Service which operates in the background under the name "RayVentory Data Hub Agent". This service is starting itself after installation and is automatically started after each reboot.

Note:
The Data Hub Agent is required in order to work with RayVentory Data Hub.

The process of setting up a data collector has the following steps:

1. Identify the machine, from which scanned services are available and where there are enough permissions
2. Check the prerequisites before installing the collector
3. Install the collector
4. Start the collector
5. Authorize the collector

Windows

Prerequisites

Support Operating Systems

The following represents the list of supported operating systems at the time of release:

- Windows Vista SP2
- Windows 7 SP1
- Windows 8
- Windows 8.1
- Windows 10
- Windows 11
- Windows Server 2008 R2
- Windows Server 2008 SP1
- Windows Server 2012
- Windows Server 2012 R2
- Windows Server 2016
- Windows Server 2019
- Windows Server 2022

Hardware requirements

- CPU: Intel Core i5
- RAM: 4GB
- Disk space: 500 MB

Software requirements

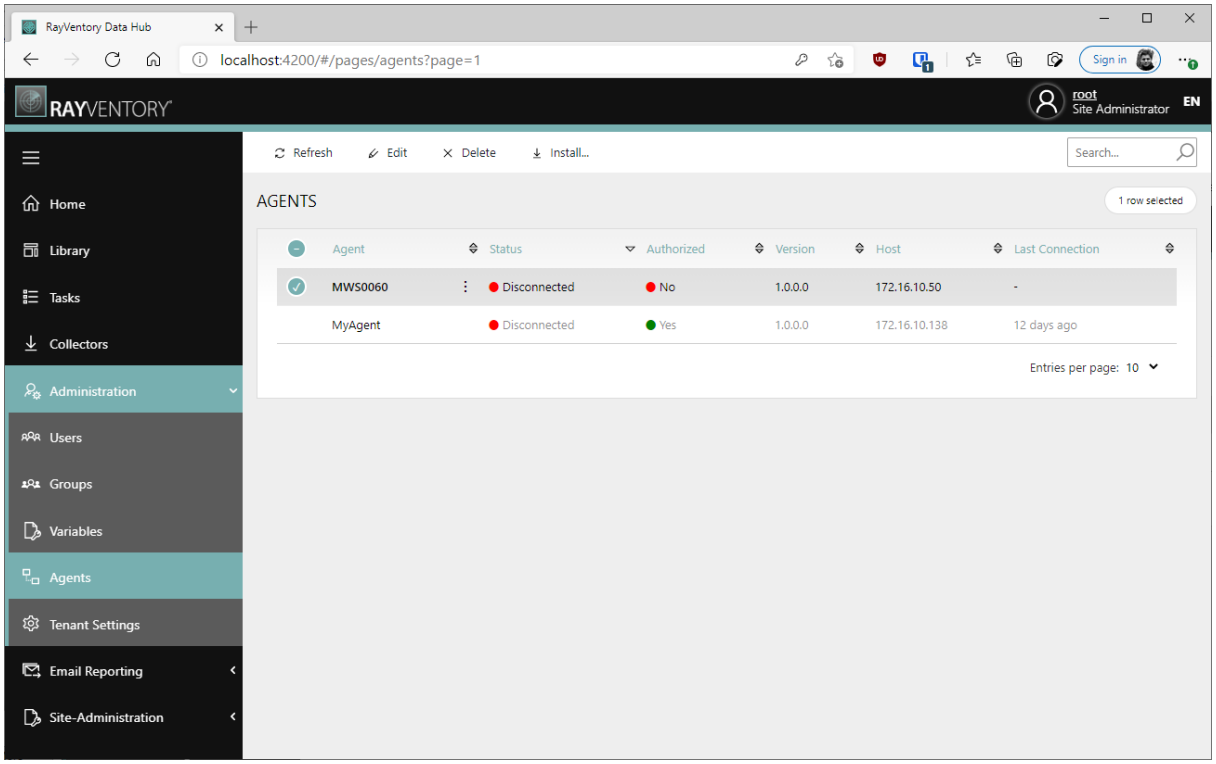
- Microsoft .NET Core 6.0.3 – Windows Server Hosting Bundle (https://dotnet.microsoft.com/en-us/download/dotnet/6.0)
- Java / OpenJDK version 11 or newer is required to execute data collection from SaaS platforms

Installation and Configuration

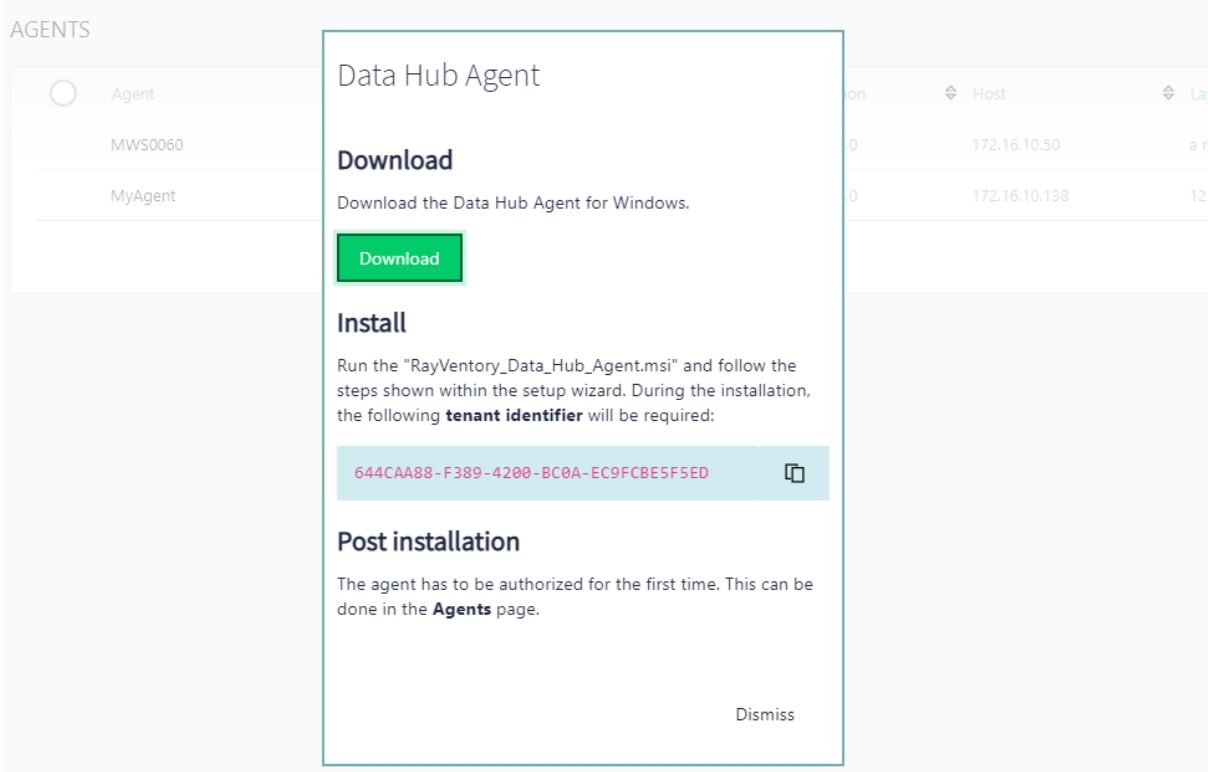
This chapter shows how to install the Data Hub Agent on Windows.

Download

Navigate to the **Agents** page under **Administration** using the navigation menu on the left panel. The list shows all agents which have been installed so far.



To install a new agent, press the **Install...** button. A pop-up with link and quick instructions will be shown:



Installation

Run the "`RayVentory_Data_Hub_Agent.msi`" and follow the steps shown within the setup wizard. You will be asked for two important properties:

- The URL of RayVentory Data Hub - the full URL, together with the protocol and port number, for example `https://rayventorydatahub.local:8090`. When in doubt what the correct URL is, check out the address bar in your browser or ask your administrator
- The tenant ID. This information is visible in the tenant selector, which is available from the login screen, settings or from tenant switcher. You can also copy the tenant ID directly from the Download pop-up

Registration

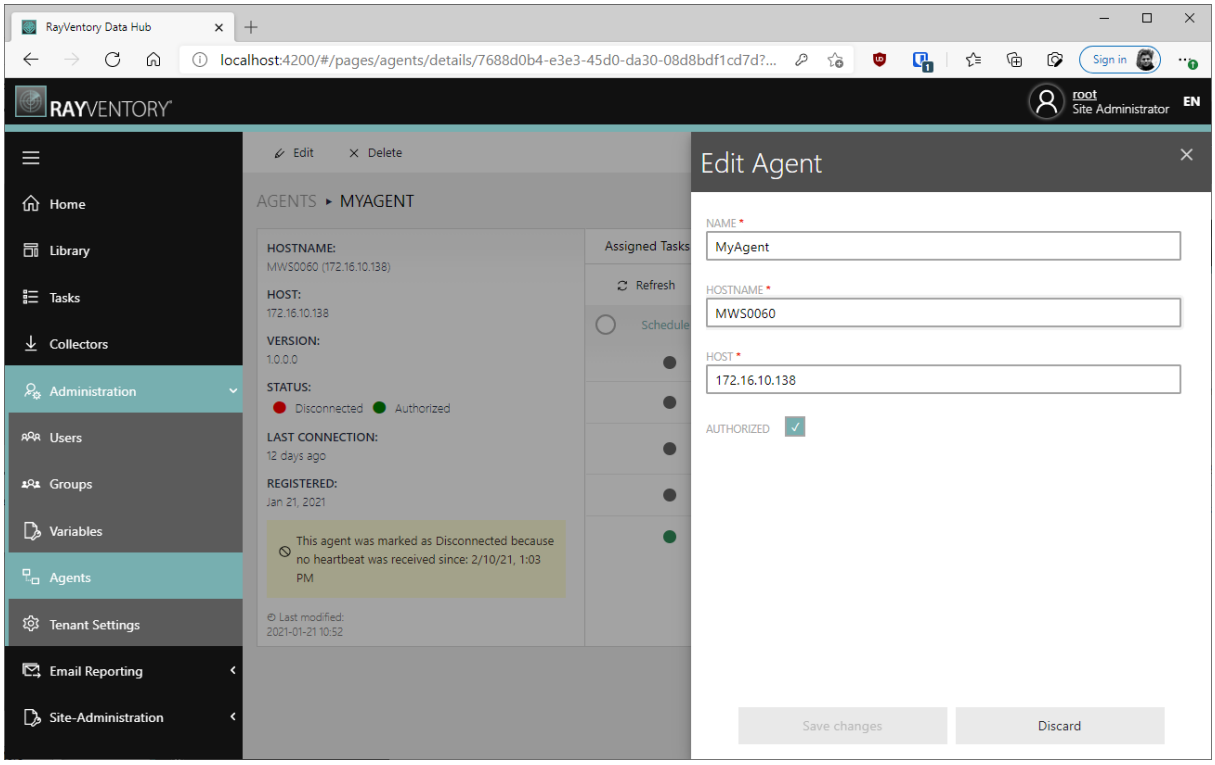
After installing the Data Hub the agent automatically connects to the provided RayVentory Data Hub server URL.

Visit the **Agents** page and search for an agent named after the machine the Agent has been installed on.

Note:
If the Data Hub Agent of the machine cannot be found, make sure that the 'RayVentory Data Hub Data Hub Agent service is actively running on your machine. If it is not actively running, start it manually and visit the collector page again.

Authorization

Visit the agent detail page by clicking on its name in the table. Edit its settings by clicking on **Edit** in the top action bar. A side panel appears to the right. Enable the "AUTHORIZE" check box in order to authorize the agent and save your changes. This agent is now authorized to request scheduled Jobs that are assigned to it.

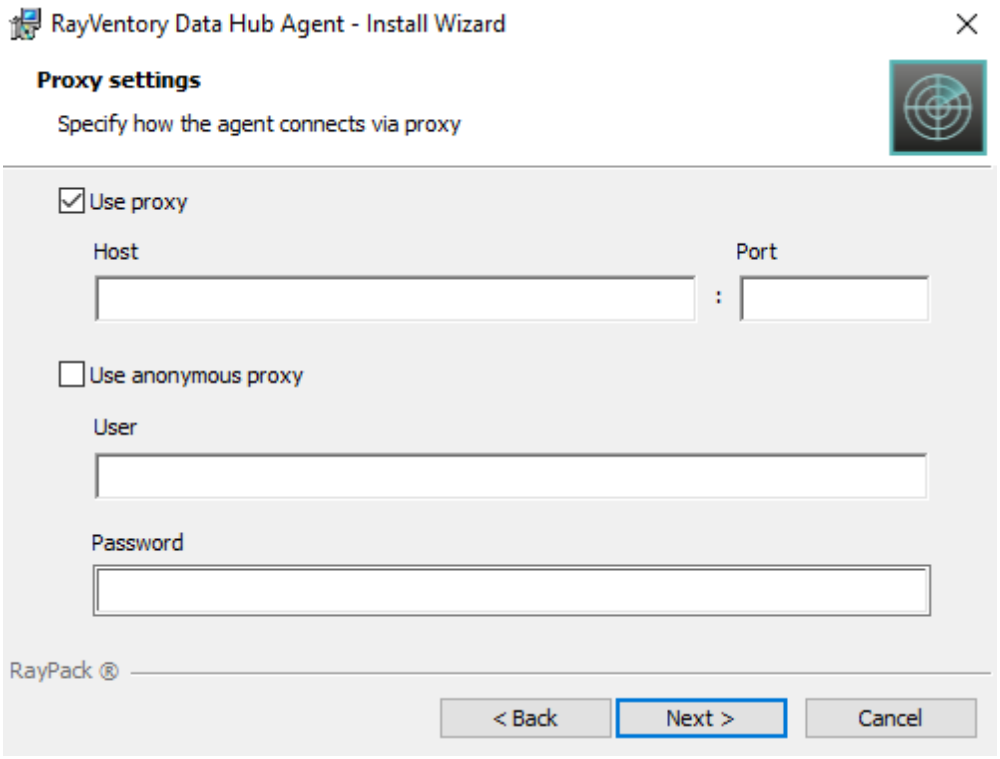


The agent is now ready to accept the tasks

Configuration

Proxy Configuration

You can use the installer to configure basic proxy properties (host, port and credentials). To configure proxy, make sure to use **Custom installation** type.



If you install the agent on your own or a custom configuration is required, the changes can be done post-mortem via the configuration file.

The configuration is stored in file `Raynet.RayVentory.DataHub.Agent.dll.config` located in the installation folder.

The following XML Attributes can be set in the `<appSettings>` XML node:

Parameter	Required	Description
`ProxyHost`	Yes	The host of the proxy
`ProxyPort`	No	The port of the proxy
`ProxyUsername`	No	The user to be used to authenticate against the proxy
`ProxyPassword`	No	The password to be used to authenticate against the proxy
`BypassProxyOnLocal`	No	A boolean value that indicates whether to bypass the proxy server for local addresses. true to bypass the proxy server for local addresses; otherwise, false. The default value is false.
`BypassList`	No	Set list of wildcards that describe URLs that do not use the proxy server when accessed - separated by a pipe character '|'. You can use the following special characters for matching: • `*` (asterisk) - matches zero or more characters • `?` (question mark) - matches exactly a single character

Migration

This list shows migration paths and additional considerations:

Data Hub Agent (formerly Data Collector up to 12.1)

- Version 12.0 -> 12.3
  SUPPORTED
  Simply install the new MSI over the old product.

- Version 12.1 -> 12.3
  SUPPORTED
  Simply install the new MSI over the old product.

- Version 12.2 -> 12.3
  SUPPORTED
  Simply install the new MSI over the old product.

Note:
It is NOT recommended TO UNINSTALL the previous version, but rather install a new one on top of it. The old version will be migrated to the new one, and the existing settings will be taken over. If you uninstall the previous version before installing a new one, a clean installation will be started, meaning that all required details will have to be provided again.

It is recommended to perform the upgrade with UI. This way it is possible to ensure that the previous settings were correctly detected and will be properly restored. The installer will automatically switch to **Custom installation** type if previous proxy configuration has been detected.

The installer will pick-up the previous settings (installation directory, tenant ID, URL, proxy etc.) and migrate/save them in the new location (`Raynet.RayVentory.DataHub.Agent.dll.config` located in the installation folder). After the migration, the previous (now obsolete) configuration files may be deleted, depending on the location in which they were originally present.

Docker

Prerequisites

Hardware requirements

- CPU: Intel Core i5
- RAM: 4GB
- Disk space: 500 MB

Software requirements

In order to run this container you'll need docker installed.

- Windows
- OS X
- Linux

On Windows, make sure to use Linux Containers, WSL2 is recommended.

Usage

Manual installation

DataHub Agent for Docker can be installed from the following image:

```
raynetgmbh/rayventory-datahubagent
```

The following variables are available when creating the container from image `raynetgmbh/rayventory-datahubagent`:

Environment variable	Description and sample value
`DataHubAgent_DataHubUrl`	The URL of the Data Hub server. This may be an URL referring the internal service name, when the image is started from a docker compose file. Example value: `http://web:80`
`DataHubAgent_TenantId`	A GUID value, representing the tenant ID. You can get your tenant ID by opening the **Administration > Agents** page, and pressing a button to install a new agent. Example value: `{72ba6fc2-d5fa-49ee-8281-841e762aea05}`

Installation with docker-compose

You can install an agent from the standard Docker compose file. For more information, refer to the following chapter: Usage

Raynet GmbH
Technologiepark 22
33100 Paderborn, Germany
T +49 5251 54009-0
F +49 5251 54009-29
info@raynet.de

www.raynet.de