

INSTALLATION GUIDE



Ofelia 2.2 Installation Guide

ascom

Abbreviations and Glossary

Abbreviations and Glossary

App	Application
NTP	Network Time Protocol
Small footprint installation	Small footprint installation is used to represent very simple installations: Single node, where Ofelia and RabbitMQ are installed on the same machine. <i>(Usually MS-SQL is too, but not necessarily).</i>
Clustered installation	Generally used to set up installations that must operate in HA mode. The installation is based on several RabbitMQ nodes <i>(necessarily an odd number)</i> , and Ofelia nodes. <i>(In this case, the MS-SQL cluster is not installed on the RabbitMQ or Ofelia nodes, for logical reasons of fault tolerance).</i>
Erlang/OTP	Erlang is a programming language and runtime environment. OTP (Open Telecom Platform) is a large collection of libraries for Erlang
RabbitMQ	RabbitMQ is the most widely deployed open-source message broker
SSL	Secure Sockets Layer A protocol that provides communications security over the Internet, now deprecated. A certificate that is referred to as “SSL” is actually a TLS certificate.
TLS	Transport Layer Security A protocol that provides communications security over a computer network. TLS is an updated version of the (now deprecated) SSL protocol. A TLS certificate might still be referred to as an “SSL” certificate
SSMS	SQL Server Management Studio
VM	Virtual Machine
HA	High availability
vHost	Virtual Host (RabbitMQ)
LB	Load Balancer
DTC	Distributed Transaction Coordinator The Distributed Transaction Coordinator (DTC or MSDTC) service is a component included in versions of Microsoft Windows from NT 4.0 onwards. It is used to coordinate operations that span multiple resource managers, such as databases, message queues and file systems.

About this document

This document describes the specificities of the components installation and their configuration required on a Windows server, and the Ofelia Setup for a small footprint installation, or for a clustered installation (high availability).



This document does not describe the installation of the Windows server.

This document does not describe the installation or HA configuration of SQL Server clusters.

This document does not describe the installation or configuration about Load Balancer.

Contents

1. Introduction:	5
1.1. Assumed Knowledge	5
1.2. Requirements	5
2. Windows environment setup:	6
2.1. Upgrading Windows:	6
2.2. How it works in a multi-node environment:	6
2.3. Snapshots and Live Migration:	7
2.4. Deployment considerations for High availability:	8
2.4.1. Number of nodes:	8
2.4.2. Rolling upgrades:	8
1 RabbitMQ:	8
2 Ofelia:	8
2.4.3. Load Balancer:	8
2.4.4. SQL cluster	8
2.5. Installation Steps:	9
2.5.1. Installation Steps for Small Footprint Ofelia	9
2.5.2. Installation Steps for multi-node architecture	10
3. Install .NET Framework:	11
4. Install Microsoft Visual C++ Redistributable:	12
5. Install the Microsoft SQL Server:	13
5.1. Small Footprint Ofelia – Install MS SQL Server Express:	13
5.2. Install SQL Server Management Studio:	17
5.3. Small Footprint Ofelia – Setting the Memory Limit for the SQL Server:	17
6. Install Erlang/OTP and RabbitMQ Server (Example on Windows® platforms):	18
6.1. Install Erlang/OTP	18
6.2. Change the Base Directory for RabbitMQ:	19
6.3. Install RabbitMQ Server:	20
6.4. Configure RabbitMQ Server:	21
6.4.1. For single node & clustered installation - Copy Erlang Cookie:	21
6.4.2. For single node & clustered installation - Enable the RabbitMQ Management GUI:	22
6.4.3. Optional – Create a specific RabbitMQ vHost for Ofelia:	23
6.4.4. Only for clustered installation – Define rabbitmq.conf file:	24
6.4.5. Only for clustered installation – Create a RabbitMQ User:	25
6.4.6. Only for clustered installation – Setup other RabbitMQ nodes to join the cluster:	26
6.4.7. Only for clustered installation – Check RabbitMQ configuration	29
7. Enable Voice synthesis capabilities (Only Windows® platforms):	31
7.1. Windows Audio service:	31
7.2. Voice package installed on Windows:	32
8. Install SmartInspect Redistributable Console:	33
9.	

11.4.	Adjust the Ofelia's http/https connection settings:	54
12.	Appendix B - Load Balancing:	58
12.1.	Health Check Monitor	58
12.2.	Load Balancing Methods.....	59
12.3.	Load Balancing of Services	60
13.	Appendix C - Using Ofelia through a reverse proxy:	61
14.	Appendix D - Flow Matrix:	62
14.1.	General ports used:	62
14.2.	Ports used by Ofelia's drivers:	63
14.3.	Ports used by Ofelia's 'External Applications':	69
15.	Related Documents	71
16.	Document History:	71

1. Introduction:

Before installing the Ascom Ofelia application on a Windows server, the latter requires specific software components and configuration.

1.1. Assumed Knowledge

You must understand the fundamentals of Microsoft Windows operating systems, Microsoft Windows SQL Server, firewall, security and networking.

1.2. Requirements

Supported Windows components and recommended minimum hardware requirements for Ascom Ofelia application are described in the product's data sheet.

Please check the third-party vendor documentation for minimum server hardware requirements.

2. Windows environment setup:

Prerequisites: Install and configure Microsoft Windows. The following is recommended:

- Disable Automatic Windows updates.
- Perform Windows updates under controlled conditions.
- Be sure Distributed Transaction Coordinator (DTC) on each server (RabbitMQ node, Ofelia, SQL) are enabled.



An NTP server must always be used for all components in the system for correct time synchronization.

2.1. Upgrading Windows:

Refer to Microsoft's documentation on how to upgrade Windows.

When upgrading Windows in a node cluster, upgrade one node at the time, that is, one node per maintenance window. A windows update can impact the performance of the system.

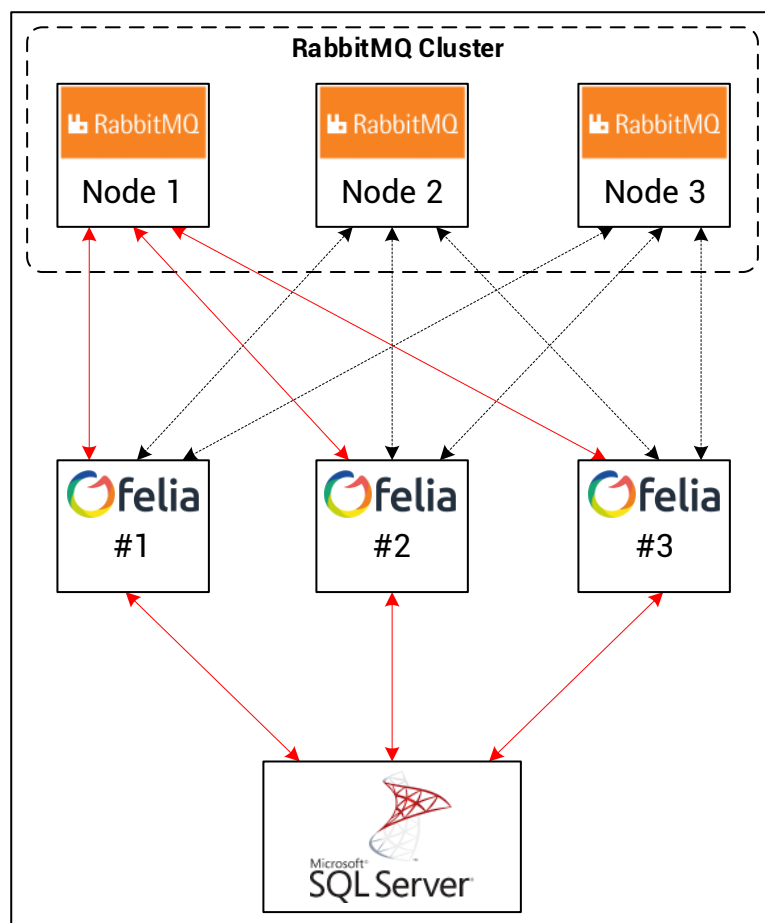
No manual work is required, so changing the active node is optional.

2.2. How it works in a multi-node environment:

For multi-cluster installations, a load balancer is required for connections to Ofelia. (*Web Browser / OfeliaBrowser / OfeliaSmart / etc.*)

➔ **The design of clustering assumes that network connections are reasonably reliable and provides a LAN-like latency. (For both RabbitMQ and Ofelia nodes)**

Refer to RabbitMQ requirements <https://www.rabbitmq.com/distributed.html>



2.3. Snapshots and Live Migration:

As all data associated with the operation and configuration of the software is retained within the prescribed SQL server instance, there is little to no use to create frequent backups of each node.

However, for disaster recovery, it is recommended that file-only backups are made and that backup routines **never include memory snapshots**, as these can negatively impact the real-time nature of the software when created and restored.

Snapshot, live migration, and backup tools that cause a temporary freeze of a node should be performed without options that snapshot the internal memory of the virtual machine. File quiescence is acceptable, but memory snapshots should never be used.

Some recommendations:

- **Do not use Snapshot on machine where RabbitMQ Messaging Broker are installed** *(both on Windows and Linux)*
While it's fine to run RabbitMQ clusters in virtualized environments, **make sure that VMs are not suspended while running.**

[Clustering and Network Partitions | RabbitMQ](#)

- For migration, remember at least the minimum number of RabbitMQ nodes need to stay alive to reach the Quorum. *(example: 3 nodes = 2 nodes need to stay alive // 5 nodes = 3 nodes need to stay alive)*

Note that some virtualization features such as migration of a VM from one host to another will tend to involve the VM being suspended, so be careful.

The last clustered RabbitMQ node that shuts down must be powered on first, otherwise the cluster is not started.

- **For clustered (HA) installation we recommend using a Linux distribution for RabbitMQ nodes.**
Please refer to RabbitMQ recommendation for supported platforms, and the associated installation guides.
[Supported Platforms | RabbitMQ](#)
- The time in the servers must also be synchronized with the same NTP server.

2.4. Deployment considerations for High availability:

2.4.1. Number of nodes:

When deciding on the number of nodes, give attention to the needs of High Availability (HA), Especially for RabbitMQ.

Example:

On an architecture with **3** RabbitMQ nodes:

- When the majority of RabbitMQ nodes (**2**) are lost, the system goes down.
- When a minority of nodes (**1**) are lost, the system stays operational.

➔ **The number of RabbitMQ nodes must always be an odd number!**

➔ The number of Ofelia nodes is not constrained (*obviously, at least 2...*).

2.4.2. Rolling upgrades:

1 RabbitMQ:

Rolling upgrades are possible only between compatible RabbitMQ and Erlang versions.

For more details on rolling upgrades of RabbitMQ nodes, please refer to the official documentation.

[Rolling upgrades | RabbitMQ](#)

2 Ofelia:

Ofelia currently only supports rolling upgrades when the upgrade is to a version with identical 'major' and 'revision' numbers. This means that a rolling upgrade can only be considered if only the build number changes.

Examples:



• Update from 2.1 Rev 3	to 2.1 Rev 3 Build 1:	Supported
• Update from 2.1 Rev 3	to 2.1 Rev 4:	Not supported.
• Update from 2.1 Rev 3	to 2.2:	Not supported.
• Update from 2.2	to 2.2 Build 1:	Supported
• Update from 2.2	to 2.2 Rev 1:	Not Supported
• Update from 2.2	to 2.2 Rev 1 Build 2:	Not Supported

MajorVersion.RevisionNumber(if exist).BuildNumber(if exist)

2.4.3. Load Balancer:

A Global Server Load Balancer (GSLB) that uses DNS for traffic routing is not supported.

Ensure the Load Balancer allow WebSocket connection for http/https.

2.4.4. SQL cluster

For HA purposes, it is recommended to include the Ofelia databases in an SQL Server Always on Availability Group

2.5. Installation Steps

Several dependencies are necessary for the Ofelia server, each of these dependencies must be installed manually before the installation of Ofelia.

The order of installation of the dependencies as indicated below must be respected.

2.5.1. Installation Steps for Small Footprint Ofelia

Perform the installation and configuration in the following order:

1. Be sure all the last **Windows updates** are installed.
Updates may be controlled through a Group Policy. In this case, you must contact the IT department to request an update of the operating system.
2. Install Microsoft **.NET framework**.
3. Install Microsoft Visual C++ Redistributable
4. Install and configure **Microsoft SQL Server**, or Retrieve information about the SQL server prepared by the customer's IT team (connection chain, identifiers, ...)
5. Install **Erlang/OTP**
6. Change the **base directory for RabbitMQ**.
7. Install **RabbitMQ Server**
8. Configure **RabbitMQ Server**:
 - a. Copy Erlang cookie
 - b. Enable RabbitMQ Management GUI
 - c. Optional – Create a specific vHost for Ofelia.
9. Install **SmartInspect** Redistributable Console.
10. Install **Ofelia**
11. Adjust **Ofelia's SQL connection** configuration.

2.5.2. Installation Steps for multi-node architecture

Perform the installation and configuration in the following order:

1. Be sure all the last **Windows updates** are installed.
2. Retrieve information about the **Microsoft SQL server** prepared by the customer's IT team (connection chain, identifiers, ...)
-
3. On each OTP/RabbitMQ node & Ofelia Servers: Install **Microsoft Visual C++ Redistributable**
4. On each OTP/RabbitMQ nodes: Install **Erlang/OTP**
5. On each OTP/RabbitMQ nodes: Change the **base directory for RabbitMQ**.
6. On each OTP/RabbitMQ nodes: Install **RabbitMQ Server**
12. **On a first** OTP/RabbitMQ node: Configure **RabbitMQ Server**:
 - a. Copy Erlang cookie
 - b. Enable RabbitMQ Management GUI
 - c. Optional – Create a specific vHost for Ofelia.
 - d. Define the rabbitmq.conf file.
 - e. Create a new RabbitMQ user.
 - f. Check RabbitMQ configuration.
7. **On the other** OTP/RabbitMQ nodes: Configure **RabbitMQ Server**
 - a. Copy the same Erlang cookie used on the 1st node.
 - b. Enable RabbitMQ Management GUI
 - c. Define the rabbitmq.conf file.
 - d. Set-up the other nodes to join the cluster.
 - e. Check RabbitMQ configuration.
-
8. On Ofelia servers only: Install Microsoft **.NET framework**.
9. On Ofelia servers only: Install **SmartInspect** Redistributable Console.
-  Ofelia nodes are generally the same as RabbitMQ nodes.
10. On each Ofelia nodes: Install **Ofelia**
13. On each Ofelia nodes: Adjust **Ofelia's SQL connection** configuration.
14. On each Ofelia nodes: Adjust **Ofelia's configuration files**.

3. Install .NET Framework:

Supported version is described in respective Ofelia product's data sheet.

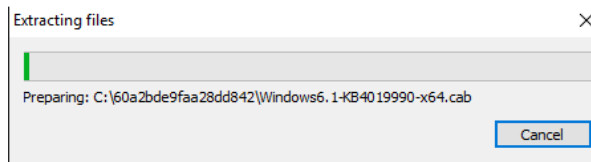


.NET Framework will only be needed on servers where Ofelia application will be installed.

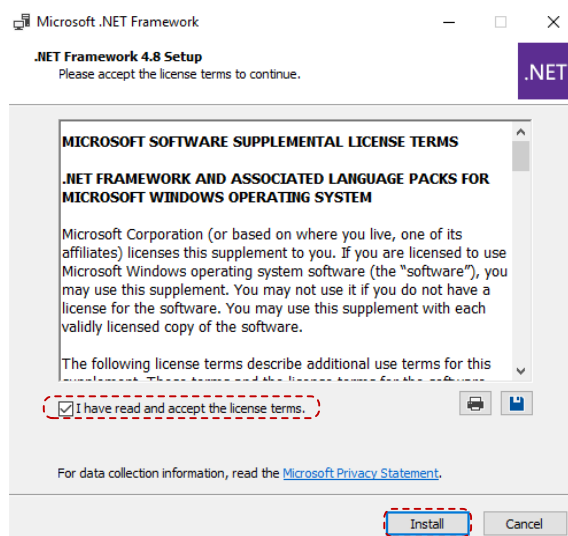
If Ofelia should be installed on multiple Windows Servers for scalability and high availability purposes, "Microsoft .NET Framework" needs to be installed on each Ofelia Windows Server.

However, please note that this component is not required on servers running only Erlang OTP & RabbitMQ applications.

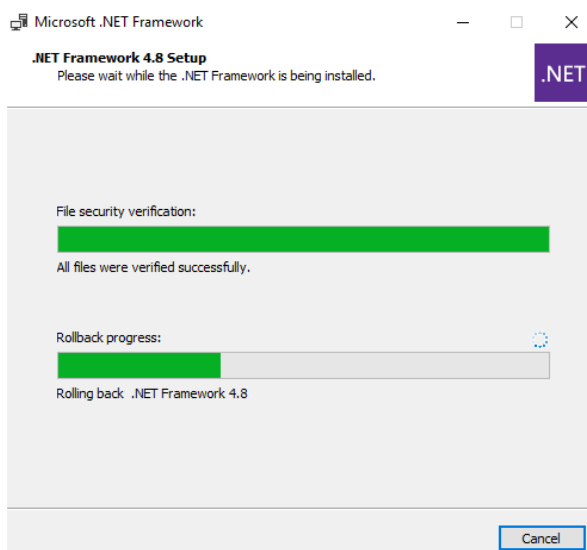
1. **Download and install** the appropriate .NET Framework.



The files are extracted



Validate the general conditions by checking the box "I have read the terms of the license agreement and I accept them.". Then start the installation using the "Install" button



Wait for the end of the installation then close the window with the "Close" button

2. **Restart** the server.

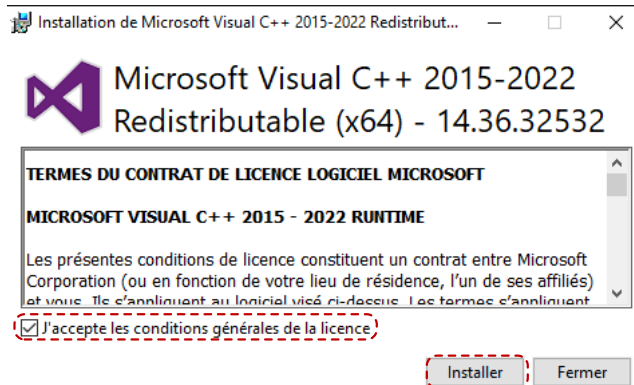
4. Install Microsoft Visual C++ Redistributable:

If the solution has multiple **RabbitMQ nodes**, the "Microsoft Visual C++ Redistributable" must be installed on each server.

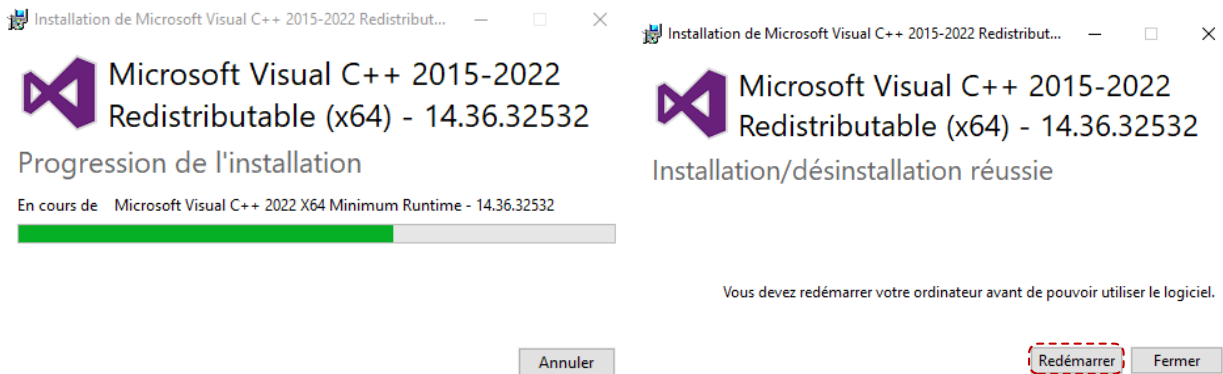
Also, "Microsoft Visual C++ Redistributable" needs to be installed on each Ofelia Windows Server.

1. **Download and install** the appropriate Microsoft Visual C++ Redistributable 2015–2022 x64

➤ https://aka.ms/vs/17/release/vc_redist.x64.exe



Validate the general conditions by checking the box "I accept the terms and conditions of the license".
Then start the installation using the "Install" button.



Wait for the end of the installation then close the window with the "**Restart**" button.

2. **Restart** the server.

5. Install the Microsoft SQL Server:

An SQL server should be installed by the customer's IT department. If a setup is intended for demo and test purposes, or if SQL Express should be installed on Small Footprint Ofelia, it is possible for a user to install an SQL server.

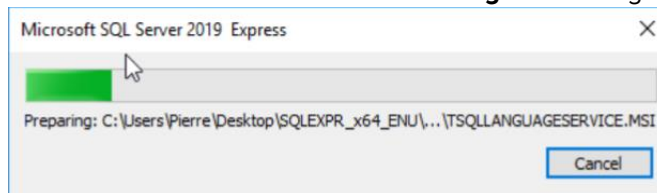
Supported versions are described in the Ofelia product's data sheet.

Depending on your server version, install the corresponding Microsoft SQL Server.

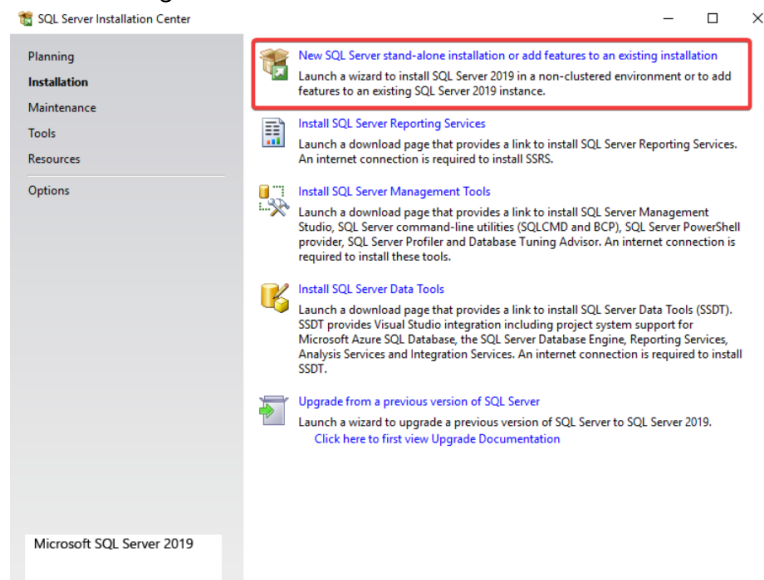
- For a Data Center installation, it is strongly recommended to install the SQL Server on a separate Windows Server. If a Windows Server where Ofelia is installed on goes down, another Ofelia can still access the SQL Server installed on the separate Windows Server.
- For a Small Footprint Ofelia installation, SQL Express should be installed on the Small Footprint Ofelia server.

5.1. Small Footprint Ofelia – Install MS SQL Server Express:

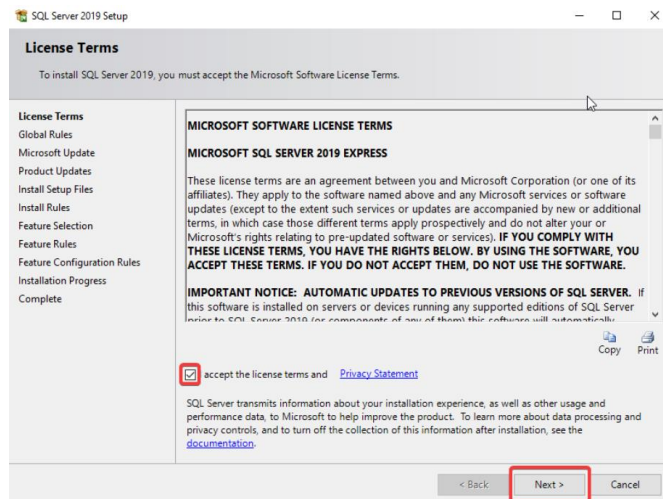
1. Double-click the installation file for SQL Server. An **Extracting Files** dialog box appears.



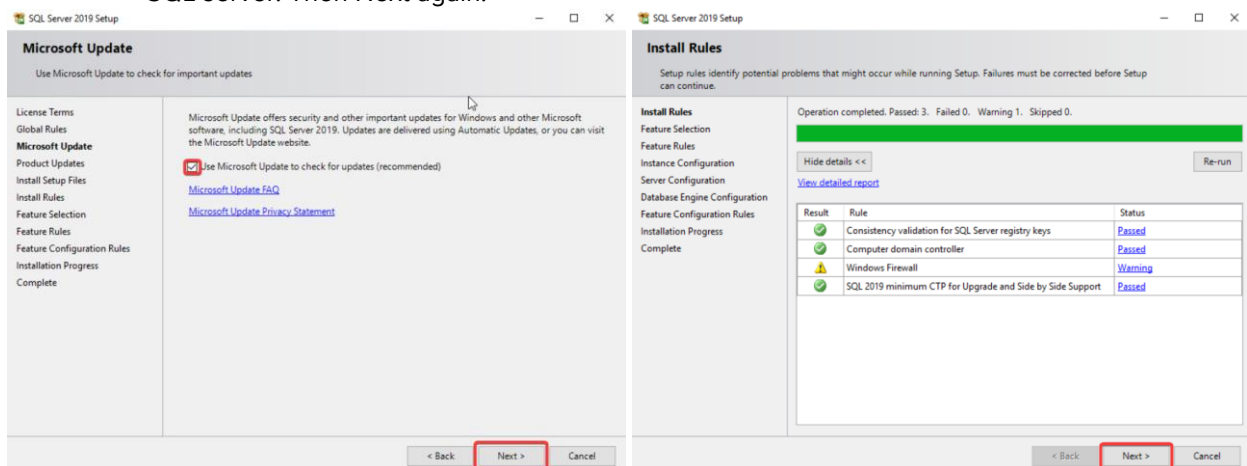
2. If a dialog box appears, which states that the SQL Server requires .NET Framework and an updated Windows Installer, click OK.
3. In the SQL Server Installation Center window, click New SQL Server stand-alone Installation or add features to an existing installation. The installation wizard starts.



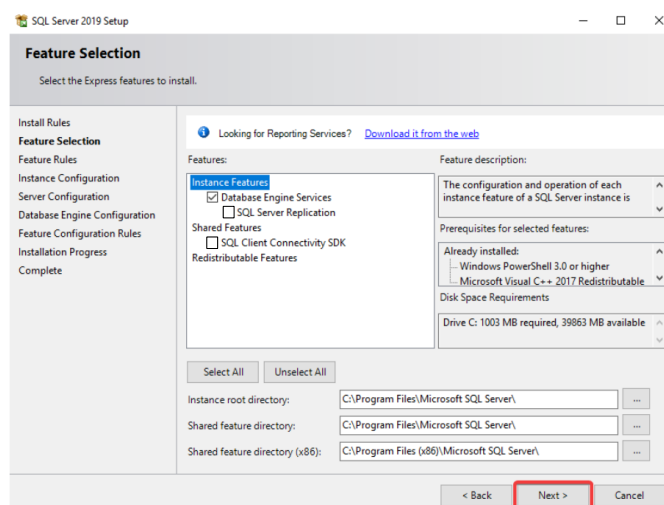
- Read through the end-user license agreement, select the I accept the license terms checkbox and click Next.



- The Product Updates window appears. Check the box to install any possible updates to the SQL server. Then Next again.



- The Feature Selection window appears. Select the SQL features to be installed. At least the “Database Engine Services” ..., then click Next.



7. In the Instance Configuration window, select/create the instance to be used, then click Next.



By default, Ofelia is configured to work with "SQLEXPRESS": if you modify it, you will have to adjust the configuration of Ofelia.

SQL Server 2019 Setup

Instance Configuration

Specify the name and instance ID for the instance of SQL Server. Instance ID becomes part of the installation path.

Install Rules
Feature Selection
Feature Rules
Instance Configuration
Server Configuration
Database Engine Configuration
Feature Configuration Rules
Installation Progress
Complete

☐ Default instance
☒ Named instance:

Instance ID:

SQL Server directory: C:\Program Files\Microsoft SQL Server\MSSQL15.SQLEXPRESS

Installed instances:

Instance Name	Instance ID	Features	Edition	Version
---------------	-------------	----------	---------	---------

< Back **Next >** Cancel

8. The Server Configuration window appears. No changes are required. Click Next.

SQL Server 2019 Setup

Server Configuration

Specify the service accounts and collation configuration.

Install Rules
Feature Selection
Feature Rules
Instance Configuration
Server Configuration
Database Engine Configuration
Feature Configuration Rules
Installation Progress
Complete

Service Accounts Collation

Microsoft recommends that you use a separate account for each SQL Server service.

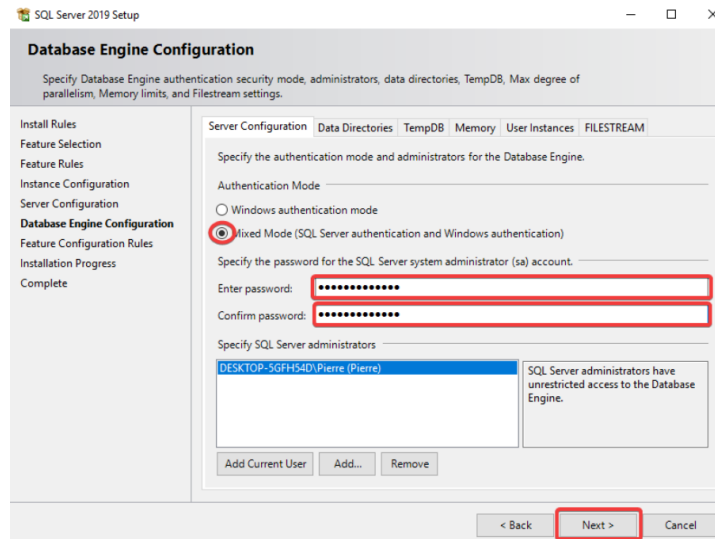
Service	Account Name	Password	Startup Type
SQL Server Database Engine	NT Service\MSSQLSQL...		Automatic
SQL Server Browser	NT AUTHORITY\LOCAL...		Disabled

☐ Grant Perform Volume Maintenance Task privilege to SQL Server Database Engine Service
This privilege enables instant file initialization by avoiding zeroing of data pages. This may lead to information disclosure by allowing deleted content to be accessed.
[Click here for details](#)

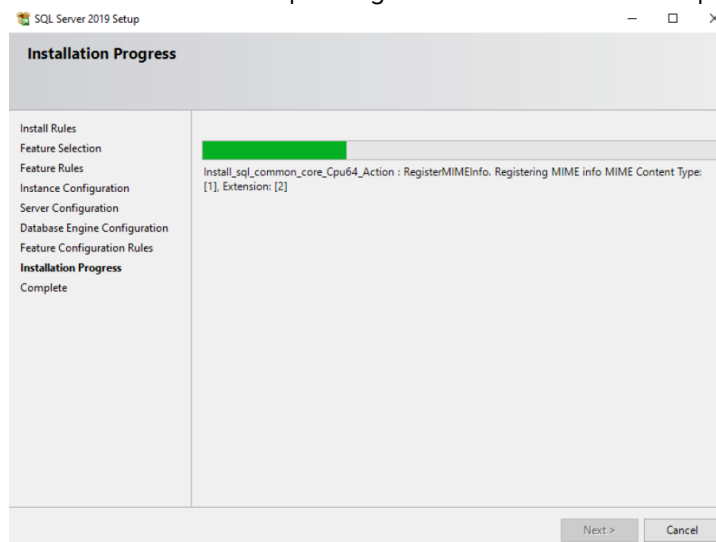
< Back **Next >** Cancel

9. In the Database Engine Configuration window:
 - **Select “Mixed Mode”**
"Mixed mode" authentication will allow you to connect to SQL either with a Windows account or with an account specific to SQL: this may be necessary if you are opening a Windows session different from the one that initiated this installation.
So, if needed, “add” other Windows users account (like service account)
 - **Enter a password for the system administrator (sa) account.** This can be used later to log on to the SQL database if the Windows account does not work.

Then click Next.



10. The Error Reporting window appears where no changes are needed. Click Next.
11. The installation can take some time depending on the server hardware capacity.



12. After a successful installation, click Close.

5.2. Install SQL Server Management Studio:

SQL Server Management Studio (SSMS) provides tools to configure, monitor, and administer instances of SQL Server and databases. SSMS is needed for granting access rights to one or more databases that will be created when Ofelia applications will be launched for the first time.

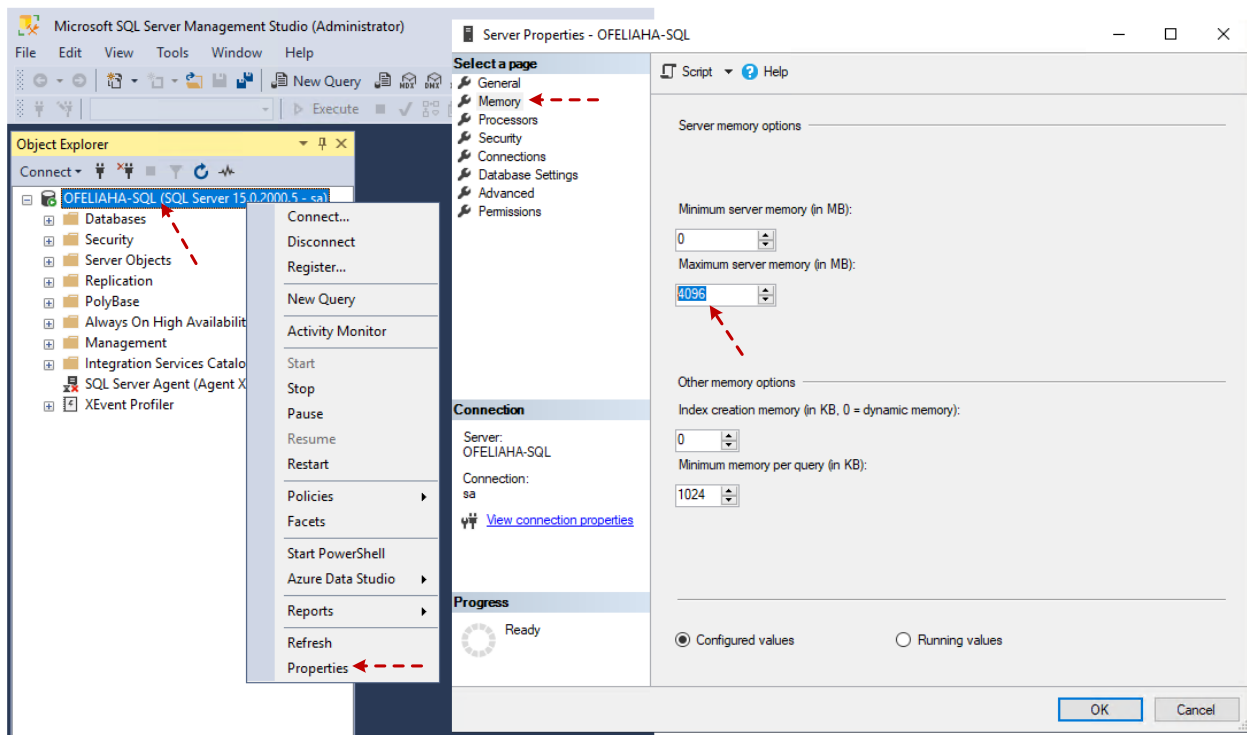
Run the Windows Installer and follow the instructions.

5.3. Small Footprint Ofelia – Setting the Memory Limit for the SQL Server:

In a Small Footprint Ofelia installation, you must ensure that the SQL server does not consume all available memory as it otherwise may impact the messaging functionality. This is configured by setting the maximum server memory in the **SQL Server Management Studio**.

To set the memory limit, perform as follows:

1. Open SQL Server Management Studio.
2. Right-click the server and select Properties.
3. In Server Properties, select Memory.
4. In the Maximum server memory field, specify the amount of memory you need and click OK.



For Small Footprint Ofelia, 4096 MB (4 GB) is usually enough. The total amount will vary somewhat depending on the situation.

6. Install Erlang/OTP and RabbitMQ Server (Example on Windows® platforms):

For more information on installing these components on other platforms (Linux, etc...), please refer to the documentation available directly on the official websites of the publishers.

Supported versions of Erlang/OTP and RabbitMQ are described in Ofelia product's data sheet.



WARNING: RabbitMQ does not support UNC paths for users, that is roaming profiles.

Before installing RabbitMQ, make sure that required IP ports are open for communication in the firewall.

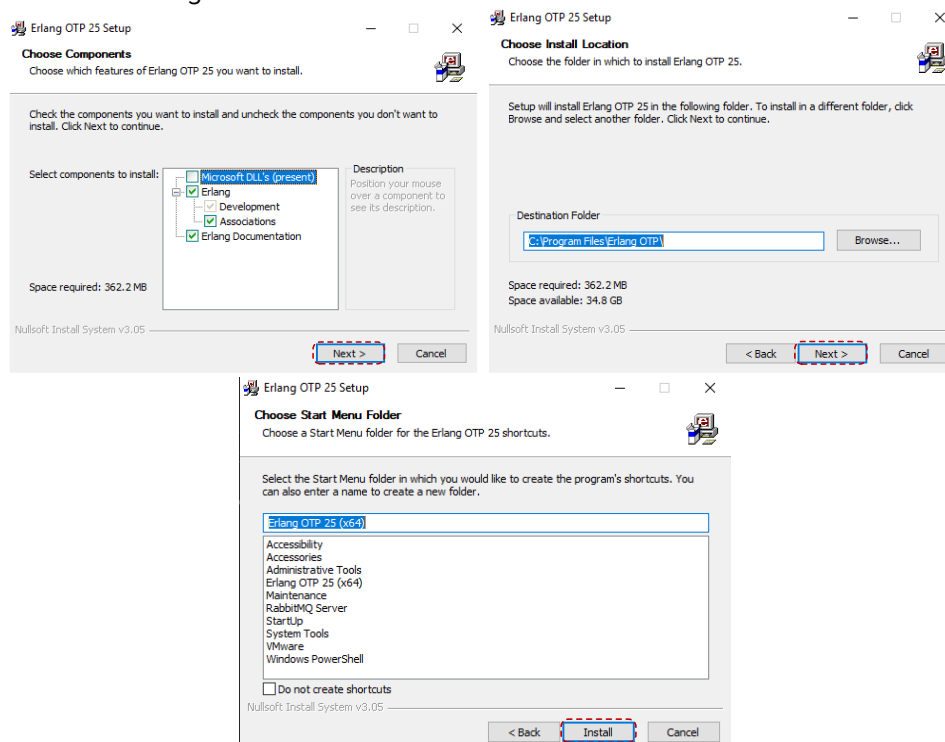
TCP 4369	Erlang EPDM daemon
TCP 6000 to 6500	RabbitMQ Stream replication
TCP 25672	Erlang distribution server port
TCP 35672 to 35682	Erlang distribution client ports
TCP 15672	RabbitMQ WEB Plugin Management

Ofelia use the RabbitMQ Server as a message broker, that acts as a message queue between applications that create/send messages and the ones that consume/receive the messages. Messages placed onto the message queue are stored until the receiving applications consume them.

6.1. Install Erlang/OTP

To be able to install a RabbitMQ Server, Erlang/OTP needs to be installed first.

1. From RabbitMQ website, check which Erlang/OTP version is required by the RabbitMQ server.
➤ <https://www.rabbitmq.com/which-erlang.html>
2. From Erlang website, download the Erlang/OTP installer file.
➤ <https://www.erlang.org/downloads>
3. Right-click the Erlang/OTP installation file and select **Run as administrator**.



4. Click **Next** twice, then click **Install**.
5. When the installation of Erlang/OTP is completed, click **Close**.
6. **If RabbitMQ is installed in a clustered installation:** repeat step 3–5 on all RabbitMQ additional nodes in the cluster.

6.2. Change the Base Directory for RabbitMQ:

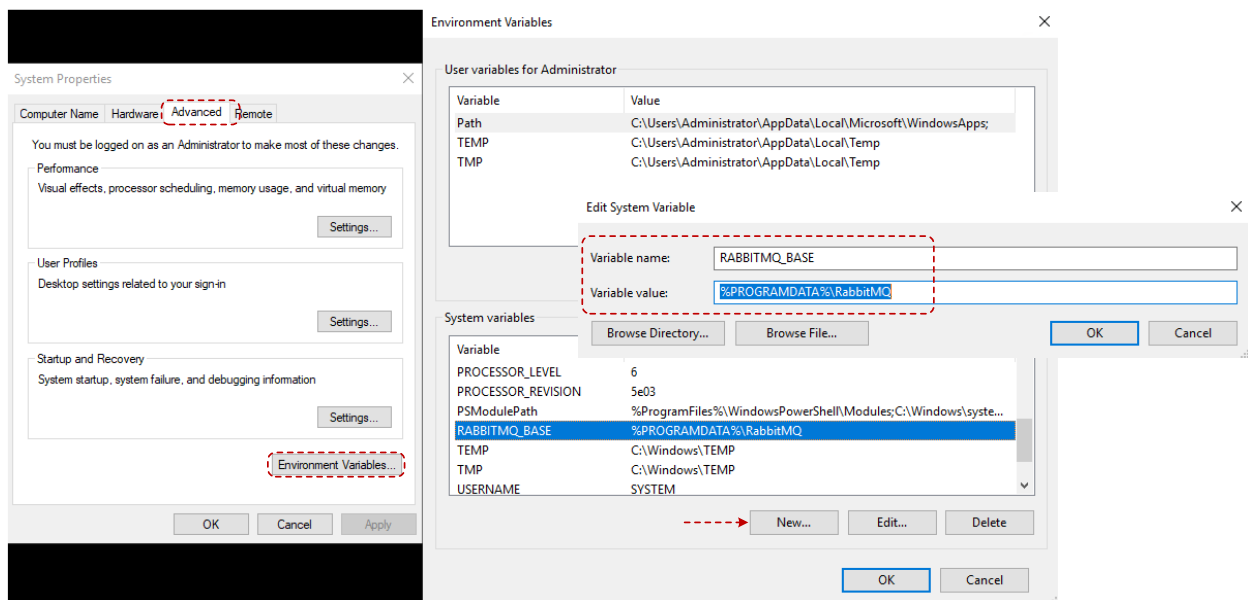
The base directory for RabbitMQ must be changed before the installation of the RabbitMQ Server.



RabbitMQ does not support UNC paths for users, that is roaming profiles.

This section describes how to change the base directory so it can be accessed by any Windows user logged in on the node. By default, the base directory for the RabbitMQ server database and log files is located in the user directory that belongs to the user who installed RabbitMQ (that is, C:\Users\<username>\AppData\Roaming\RabbitMQ).

1. In Windows **Control Panel**, select **System and Security** → **System** → **Advanced system settings**.
2. Click **Environment Variables**.
3. In the **System variables** section, click **New**.
4. Set the following system environment variable:
 - Variable name: **RABBITMQ_BASE**
 - Variable value: **%PROGRAMDATA%\RabbitMQ**



5. Click **OK**.
-
6. **If RabbitMQ is installed in a clustered installation:** repeat step 1–5 on all additional nodes in the cluster.

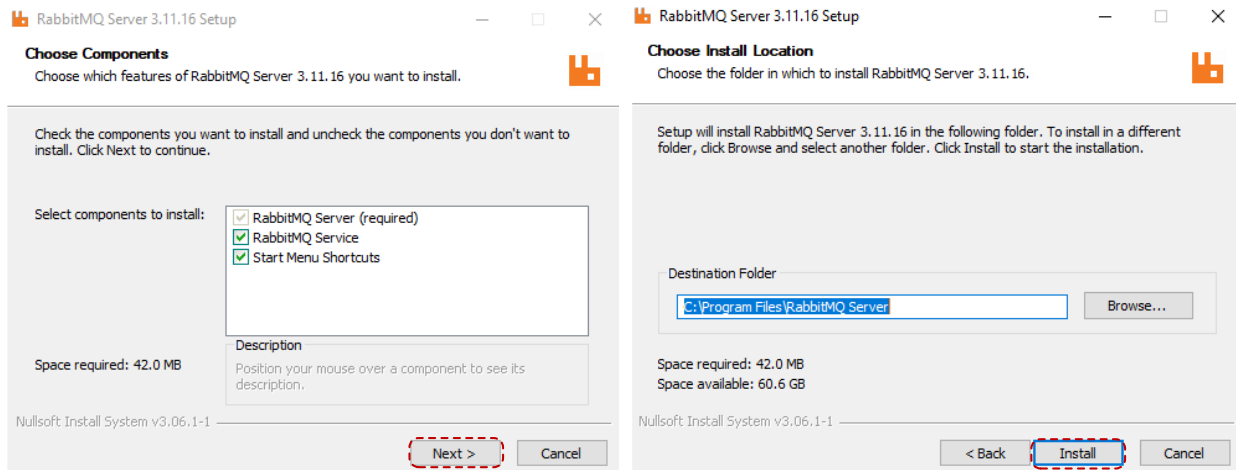
6.3. Install RabbitMQ Server:

1. **When the base directory for RabbitMQ has been changed**, download the RabbitMQ Server installer file.

➤ <https://www.rabbitmq.com/>

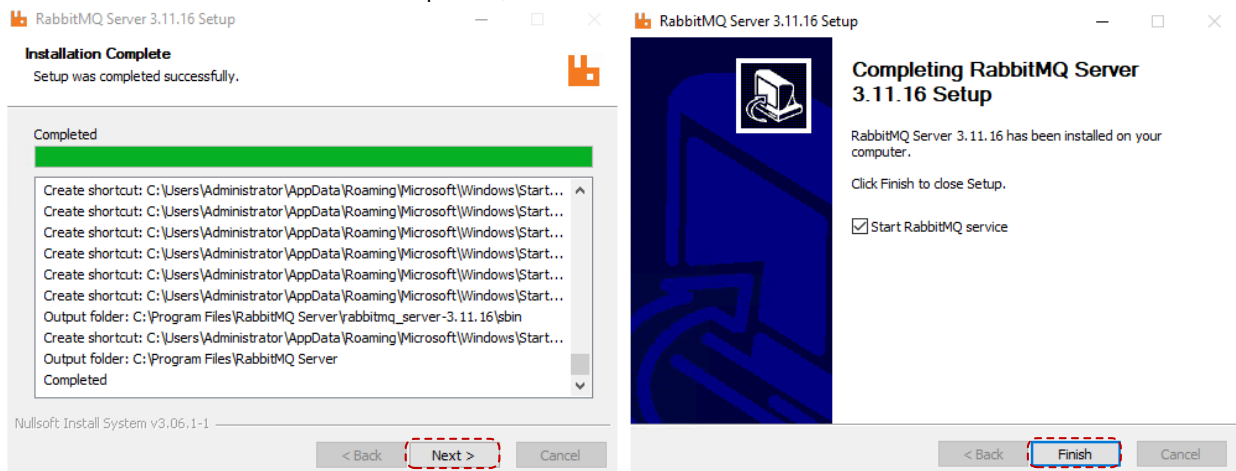
2. Right-click the RabbitMQ Server installation file and select **Run as administrator**.

3. Click **Next** and then **Install**.



During the installation, you might be asked by the firewall about the Erlang/OTP files epmd.exe and erl.exe can access to network. **Allow these applications to run.**

4. When the installation is completed, click **Next** and then **Finish**.



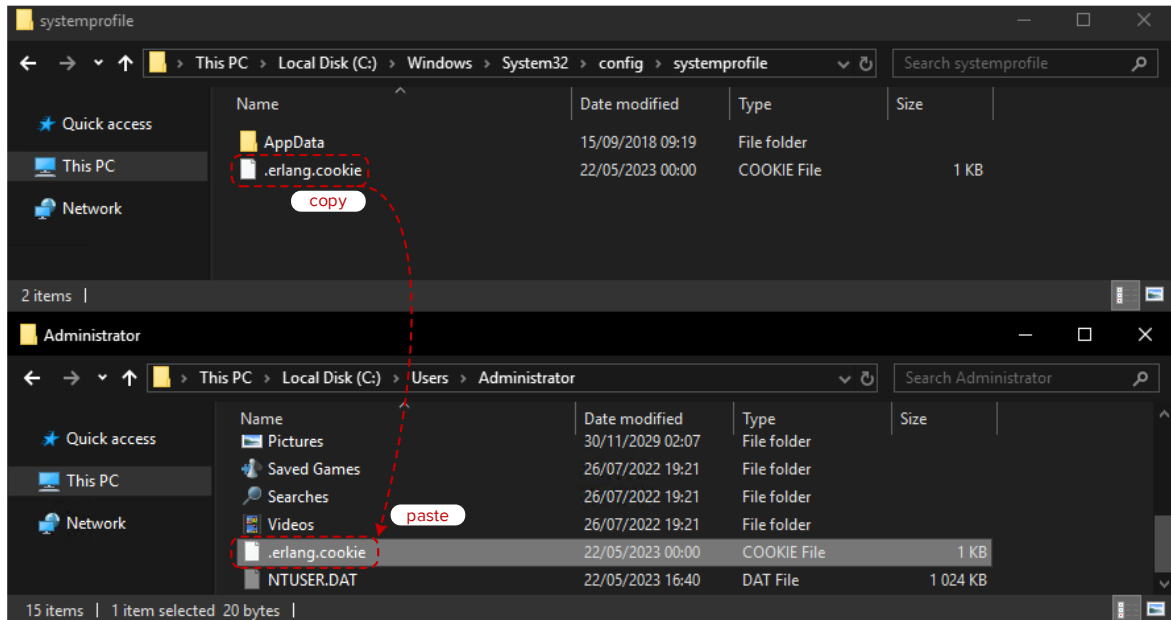
--

5. **If RabbitMQ is installed in a clustered installation:** repeat step 1–4 on all additional nodes in the cluster.

6.4. Configure RabbitMQ Server:

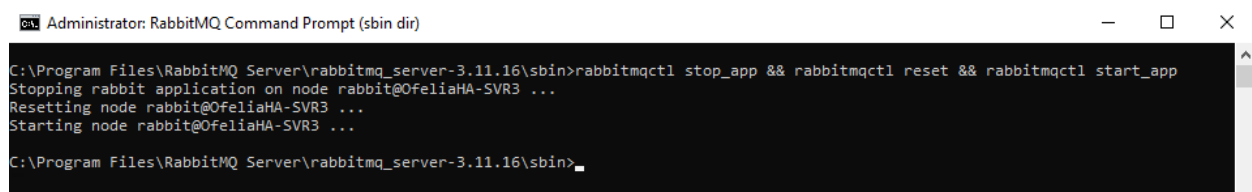
6.4.1. For single node & clustered installation - Copy Erlang Cookie:

1. Log on to the server with the Windows user account that should be used for RabbitMQ installation.
2. With Windows explorer, go to < %WINDIR%\System32\config\systemprofile\ > and copy the <.erlang.cookie> file to < %HOMEDRIVE%%HOMEPATH%\ >.



Also make sure that <.erlang.cookie> file is copied to the service account running the Ofelia services. This may be another account than the one used to configure the server. Usually in C:\Users\<user>

3. **Important: Restart** the server (Or just restart RabbitMQ Windows service)
4. Click the Windows Start button, type “**RabbitMQ Command Prompt**”, and click **Enter**.
➤ The RabbitMQ Command Prompt window appears.
5. Reset RabbitMQ by typing successively the following commands:
`rabbitmqctl stop_app && rabbitmqctl reset && rabbitmqctl start_app`



--

6. **If RabbitMQ is installed in a clustered installation:** repeat step 1–3 (but not step 4 and 5) for all additional nodes, using the <.erlang.cookie> of the first node (as configured in step 2).
➤ **The same <.erlang.cookie> must be used / copied on all nodes**

6.4.2. For single node & clustered installation - Enable the RabbitMQ Management GUI:

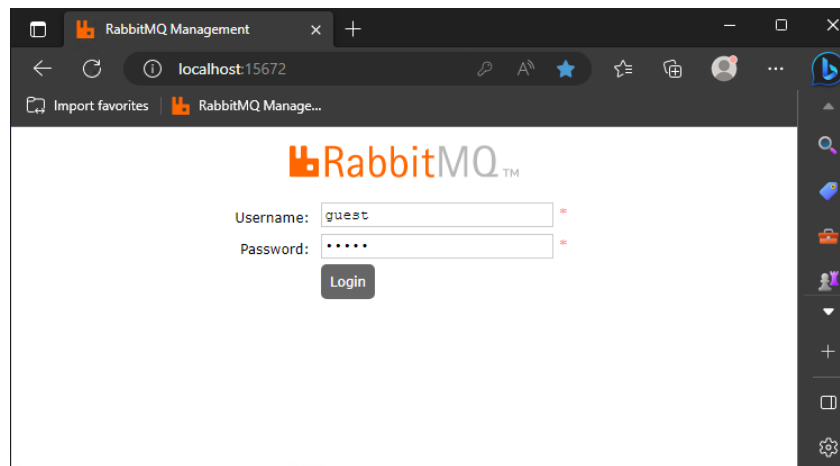
The RabbitMQ Management GUI is used for the management of RabbitMQ nodes from a web browser. The Ofelia Server monitors the RabbitMQ nodes through the RabbitMQ HTTP API, which requires that the RabbitMQ Management GUI is enabled. If the RabbitMQ Management GUI is not enabled, the monitoring will not work.

1. Log on to the server with the Windows user account that should be used for RabbitMQ installation.
2. Click the Windows Start button, type “**RabbitMQ Command Prompt**”, and click **Enter**.
 - The RabbitMQ Command Prompt window appears.
3. Enable the RabbitMQ web page GUI by typing:
`rabbitmq-plugins enable rabbitmq_management`



```
Administrator: RabbitMQ Command Prompt (sbin dir)
C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>rabbitmq-plugins enable rabbitmq_management
Enabling plugins on node rabbit@OfeliaHA-SVR3:
rabbitmq_management
The following plugins have been configured:
  rabbitmq_management
  rabbitmq_management_agent
  rabbitmq_web_dispatch
Applying plugin configuration to rabbit@OfeliaHA-SVR3...
The following plugins have been enabled:
  rabbitmq_management
  rabbitmq_management_agent
  rabbitmq_web_dispatch
started 3 plugins.
C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>
```

4. Open the RabbitMQ web page by visiting <http://localhost:15672>
5. Check by log-on with the default RabbitMQ user (*localhost connection only*): `guest / guest`



-
6. **If RabbitMQ is installed in a clustered installation:** repeat steps 1–5 for any additional nodes.

6.4.3. Optional – Create a specific RabbitMQ vHost for Ofelia:

This section describes how to create a new vHost from RabbitMQ Command Prompt.

For small footprint installations where Ofelia is installed on the same machine as RabbitMQ (single node), Ofelia will try to create the virtual host to be used if it does not already exist, using the RabbitMQ "guest" account.

- ➔ For the vast majority of installations with a small footprint (single node), this step will not be necessary, as Ofelia will perform it automatically during the first start.

For installations where Ofelia is not installed on the same machine as RabbitMQ or **in clustered installation**, the RabbitMQ "guest" account cannot be used (as it can only be used as localhost). It will then be necessary to configure Ofelia with a RabbitMQ account that has sufficient rights to create the virtual host.

Refer to chapter: [Only for clustered installation – Create a RabbitMQ User](#)

Only if the automatic creation of the virtual host at the start of Ofelia is not possible due to the RabbitMQ account used, it is still possible to create the virtual host manually from the command prompt:

1. Log on to the server with the Windows user account that should be used for RabbitMQ installation.
2. Click the Windows Start button, type "**RabbitMQ Command Prompt**", and click **Enter**.
 - The RabbitMQ Command Prompt window appears.
3. Create a new vHost, for example <Ofelia>, by typing:
`rabbitmqctl add_vhost Ofelia`
4. Give the default user RabbitMQ permission to access the created virtual host, by typing:
`rabbitmqctl set_permissions -p Ofelia guest "." "*" "*"`



It is strongly recommended that the name of the vHost does not contain any spaces or special characters.

```
Administrator: RabbitMQ Command Prompt (sbin dir)

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>rabbitmqctl add_vhost Ofelia
Adding vhost "Ofelia" ...

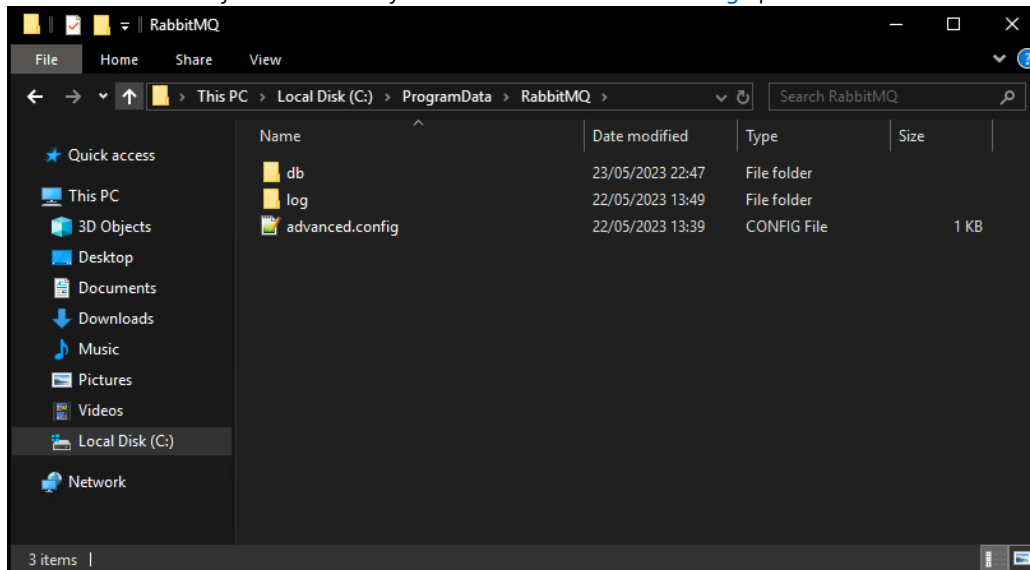
C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>rabbitmqctl set_permissions -p Ofelia guest "." "*" "*"
Setting permissions for user "guest" in vhost "Ofelia" ...

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>
```

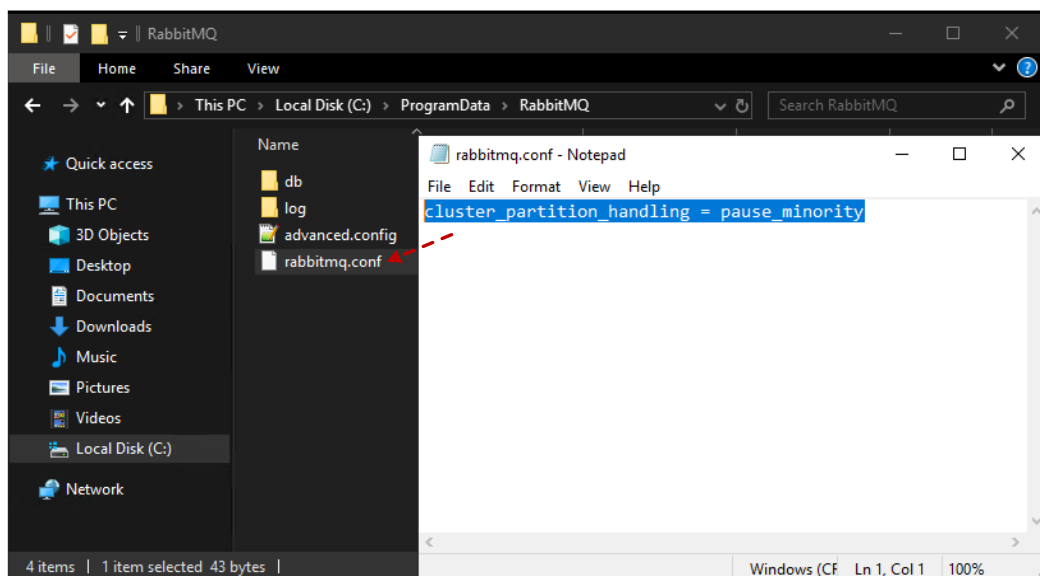
6.4.4. Only for clustered installation – Define rabbitmq.conf file:

 The following actions must be performed on each RabbitMQ node in the cluster.

1. With Windows explorer, **open** `%RABBITMQ_BASE%` folder
➤ Where you have already a file called `<advanced.config>` present.



2. **Create** a new file called `<rabbitmq.conf>`
3. **Edit** this file with a text editor, and insert the following line to disable minority nodes:
`cluster_partition_handling = pause_minority`



4. **Save** modifications and **close** the text editor.
5. **Important: Restart.** (Or just restart RabbitMQ Windows service)
6. **Repeat step 1–5** on all additional nodes in the cluster.

6.4.5. Only for clustered installation – Create a RabbitMQ User:

This section describes how to create a RabbitMQ user from RabbitMQ Command Prompt.



If RabbitMQ is installed in a clustered installation, **perform the following actions only on the first node of the cluster** (usually used as 'start node' for cluster configuration)

1. Log on to the server with the Windows user account that should be used for RabbitMQ installation.
2. Click the Windows Start button, type “**RabbitMQ Command Prompt**”, and click **Enter**.
 - The RabbitMQ Command Prompt window appears.
3. Create a new user by typing, for example:
`rabbitmqctl add_user OfeliaSystemUser OfeliaSystemPassword`
4. Give the user administration privileges by typing to user created in step 3, for example:
`rabbitmqctl set_user_tags OfeliaSystemUser administrator`
5. Give the user permission to the virtual host configured in RabbitMQ:
If the default RabbitMQ virtual host “/” should be used, by typing:
`rabbitmqctl set_permissions -p / OfeliaSystemUser ".*" ".*" ".*"`

If a specific virtual host should be used, as for example <OfeliaHA>, by typing:

`rabbitmqctl set_permissions -p Ofelia OfeliaSystemUser ".*" ".*" ".*"`

```
Administrator: RabbitMQ Command Prompt (sbin dir)

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>rabbitmqctl add_user OfeliaSystemUser OfeliaSystemPassword
Adding user "OfeliaSystemUser" ...
Done. Don't forget to grant the user permissions to some virtual hosts! See 'rabbitmqctl help set_permissions' to learn more.

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>rabbitmqctl set_user_tags OfeliaSystemUser administrator
Setting tags for user "OfeliaSystemUser" to [administrator] ...

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>rabbitmqctl set_permissions -p Ofelia OfeliaSystemUser ".*" ".*" ".*"
Setting permissions for user "OfeliaSystemUser" in vhost "Ofelia" ...

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>
```

6. **Take a note of the virtual host, username, and password**, as they will be used later during the configuration of Ofelia software.

6.4.6. Only for clustered installation – Setup other RabbitMQ nodes to join the cluster:

This section describes how to ask each of the additional RabbitMQ nodes to join the cluster.
To do this, ask each of the other nodes to join the first node prepared.

Before performing the following actions (only on additional nodes) ensure that:

- 1- Required IP ports are open for communication in the firewall (on each RabbitMQ servers' nodes)
 - TCP 4369 Erlang EPDM daemon
 - TCP 6000 to 6500 RabbitMQ Stream replication
 - TCP 25672 Erlang distribution server port
 - TCP 35672 to 35682 Erlang distribution client ports
 - TCP 15672 RabbitMQ WEB Plugin Management
- 2- For each node (the 1st node configured, like the additional):
 - Be able to resolve their host name with each other.
 - Communicate over TCP/IP...
 - Have the above-mentioned ports opened in the firewall.
 - That the same erlang cookie file has been used on all machines.
- 3- The target node for joining the cluster is running (usually the 1st node).

If one of these points is not checked and validated upstream, the following error may appear:

```
CA: Select Administrator: RabbitMQ Command Prompt (sbin dir)

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>rabbitmqctl stop_app && rabbitmqctl reset
Stopping rabbit application on node rabbit@OfeliaHA-SVR2 ...
Resetting node rabbit@OfeliaHA-SVR2 ...

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>rabbitmqctl join_cluster rabbit@OfeliaHA-SVR1
Clustering node rabbit@OfeliaHA-SVR2 with rabbit@OfeliaHA-SVR1
Error: unable to perform an operation on node 'rabbit@OfeliaHA-SVR1'. Please see diagnostics information and suggestions below.

Most common reasons for this are:

* Target node is unreachable (e.g. due to hostname resolution, TCP connection or firewall issues)
* CLI tool fails to authenticate with the server (e.g. due to CLI tool's Erlang cookie not matching that of the server)
* Target node is not running

In addition to the diagnostics info below:

* See the CLI, clustering and networking guides on https://rabbitmq.com/documentation.html to learn more
* Consult server logs on node rabbit@OfeliaHA-SVR1
* If target node is configured to use long node names, don't forget to use --longnames with CLI tools

DIAGNOSTICS
=====
attempted to contact: ['rabbit@OfeliaHA-SVR1']

rabbit@OfeliaHA-SVR1:
* unable to connect to epmd (port 4369) on OfeliaHA-SVR1: timeout (timed out)

Current node details:
* node name: 'rabbitmqcli-156-rabbit@OfeliaHA-SVR2'
* effective user's home directory: c:/Users/Administrator
* Erlang cookie hash: KEVzCF4mum62Fut8qXCikg==
```

How to join a cluster:



Ensure the first node of the cluster is correctly started, then that will join the first node in the cluster

1. Log on to each server with the Windows user account that should be used for RabbitMQ installation.
2. Click the Windows Start button, type “**RabbitMQ Command Prompt**”, and click **Enter**.
➤ The RabbitMQ Command Prompt window appears.
3. Join the cluster a new user by typing successively the following commands:

```
rabbitmqctl stop_app && rabbitmqctl reset
rabbitmqctl join_cluster rabbit@<1st node's name>
rabbitmqctl start_app
```

In the example below:

`rabbitmqctl join_cluster rabbit@OfeliaHA-SVR1`
where <OfeliaHA-SVR1> is the 1st node's hostname.

```
Administrator: RabbitMQ Command Prompt (sbin dir)

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>rabbitmqctl stop_app && rabbitmqctl reset
Stopping rabbit application on node rabbit@OfeliaHA-SVR3 ...
Resetting node rabbit@OfeliaHA-SVR3 ...

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>rabbitmqctl join_cluster rabbit@OfeliaHA-SVR1
Clustering node rabbit@OfeliaHA-SVR3 with rabbit@OfeliaHA-SVR1

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>rabbitmqctl start_app
Starting node rabbit@OfeliaHA-SVR3 ...

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>
```

By connecting to RabbitMQ web page you can check that the nodes have successfully joined the cluster:

RabbitMQ Management Overview page showing cluster status. The page indicates 3 nodes are connected. The 'Nodes' table lists the following details:

Name	File descriptors	Socket descriptors	Erlang processes	Memory	Disk space	Uptime	Info	Reset stats
rabbit@OfeliaHA-SVR1	280 65536 available	0 58893 available	403 1048576 available	69 MiB 3.2 GiB high watermark	35 GiB 48 MiB low watermark	1d 22h	basic disc 1 rss	This node All nodes
rabbit@OfeliaHA-SVR2	278 65536 available	0 58893 available	395 1048576 available	62 MiB 3.2 GiB high watermark	35 GiB 48 MiB low watermark	1d 22h	basic disc 1 rss	This node All nodes
rabbit@OfeliaHA-SVR3	316 65536 available	0 58893 available	396 1048576 available	69 MiB 3.2 GiB high watermark	61 GiB 48 MiB low watermark	2h 56m	basic disc 1 rss	This node All nodes

How to remove a node from the running cluster:

In case of error, maintenance, migration, etc... you may need remove a node from a cluster (e.g. a node that is no longer relevant to keep) from a node still present in the cluster.

On another node than the one to be deleted, being still active in the cluster:

1. Log on to a server with the Windows user account that should be used for RabbitMQ installation.
2. Click the Windows Start button, type "**RabbitMQ Command Prompt**", and click **Enter**.
➤ The RabbitMQ Command Prompt window appears.
3. Stop the node need to be removed by typing:
`rabbitmqctl -n rabbit@<node's name needs to be removed> stop_app`

In the example below:

`rabbitmqctl -n rabbit@OfeliaHA-SVR3 stop_app`

where <OfeliaHA-SVR3> is the node's hostname need to be removed from the cluster.

```
Administrator: RabbitMQ Command Prompt (sbin dir)

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>rabbitmqctl -n rabbit@OfeliaHA-SVR3 stop_app
Stopping rabbit application on node rabbit@OfeliaHA-SVR3 ...

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>
```

At this point, by connecting to RabbitMQ web page you can check that the node is not running anymore:

The screenshot shows the RabbitMQ web interface with the 'Nodes' tab selected. The 'Nodes' table lists three nodes: 'rabbit@OfeliaHA-SVR1', 'rabbit@OfeliaHA-SVR2', and 'rabbit@OfeliaHA-SVR3'. The 'rabbit@OfeliaHA-SVR3' row is highlighted in red and has a red dashed box around it with the text 'Node not running' next to it. The other two nodes are in a green state.

Name	File descriptors ?	Socket descriptors ?	Erlang processes	Memory ?	Disk space	Uptime	Info	Reset stats
rabbit@OfeliaHA-SVR1	280 65536 available	0 58893 available	403 1048576 available	67 MIB 3.2 GiB high watermark 48 MIB low watermark	35 GiB	1d 21h	basic disc 1 rss	This node All nodes
rabbit@OfeliaHA-SVR2	276 65536 available	0 58893 available	394 1048576 available	61 MIB 3.2 GiB high watermark 48 MIB low watermark	35 GiB	1d 21h	basic disc 1 rss	This node All nodes
rabbit@OfeliaHA-SVR3	Node not running							

4. Now, it is possible to remove the desired node by typing:
`rabbitmqctl forget_cluster_node rabbit@<node's name needs to be deleted>`

In the example below:

`rabbitmqctl forget_cluster_node rabbit@OfeliaHA-SVR3`

where <OfeliaHA-SVR3> is the node's hostname need to be removed from the cluster.

```
Administrator: RabbitMQ Command Prompt (sbin dir)

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>rabbitmqctl -n rabbit@OfeliaHA-SVR3 stop_app
Stopping rabbit application on node rabbit@OfeliaHA-SVR3 ...

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>rabbitmqctl forget_cluster_node rabbit@OfeliaHA-SVR3
Removing node rabbit@OfeliaHA-SVR3 from the cluster

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>
```

By connecting to RabbitMQ web page you can check that the node is removed from the cluster:

Nodes								
Name	File descriptors ?	Socket descriptors ?	Erlang processes	Memory ?	Disk space	Uptime	Info	Reset stats
rabbit@OfeliaHA-SVR1	280 65536 available	0 58893 available	403 1048576 available	68 MIB 3.2 GiB high watermark 48 MIB low watermark	35 GiB	1d 21h	basic disc 1 rss	This node All nodes
rabbit@OfeliaHA-SVR2	276 65536 available	0 58893 available	394 1048576 available	61 MIB 3.2 GiB high watermark 48 MIB low watermark	35 GiB	1d 21h	basic disc 1 rss	This node All nodes

6.4.7. Only for clustered installation – Check RabbitMQ configuration

This section describes how to check if the specific RabbitMQ's configuration file created is correctly loaded and if RabbitMQ cluster is running with the specific parameter.

(See chapter 6.4.4 Only for clustered RabbitMQ installation – Define rabbitmq.conf file)



The following verification must be performed on each RabbitMQ node in the cluster.

Check if the <rabbitmq.conf> file is correctly loaded.

By connecting to RabbitMQ web page you can check, by clicking on each node's name, that the nodes have successfully loaded the <rabbitmq.conf> config file:

The screenshot displays the RabbitMQ web interface. On the left, the 'Overview' tab is active, showing a list of nodes: rabbit@OfeliaHA-SVR1, rabbit@OfeliaHA-SVR2, and rabbit@OfeliaHA-SVR3. A red dashed box highlights the 'rabbit@OfeliaHA-SVR1' node. To the right, the 'Node rabbit@OfeliaHA-SVR1' configuration page is shown. The 'Config file' section is highlighted with a red dashed box, displaying the path: c:/ProgramData/RabbitMQ/advanced.config; c:/ProgramData/RabbitMQ/rabbitmq.conf. Below this, the 'Process statistics' table is visible, showing various metrics for the node.

Process statistics	
65536 available	58893 available
283	0
65536 available	58893 available

Check if the cluster run with the parameter <cluster_partition_handling = pause_minority> .

1. Log on to each server with the Windows user account that should be used for RabbitMQ installation.
2. Click the Windows Start button, type “**RabbitMQ Command Prompt**”, and click **Enter**.
➤ The RabbitMQ Command Prompt window appears.
3. Run the following commands:
`rabbitmqctl eval application:get_all_env(rabbit).` (with the ‘.’ At the end)
4. Check for a line that look like: `{cluster_partition_handling,pause_minority},`

```

Administrator: RabbitMQ Command Prompt (sbin dir)

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.11.16\sbin>rabbitmqctl eval application:get_all_env(rabbit).
[{handshake_timeout,10000},
 {fhc_write_buffering,true},
 {default_user_tags,[administrator]},
 {server_properties,[],},
 {frame_max,131072},
 {dead_letter_worker_consumer_prefetch,32},
 {proxy_protocol,false},
 {default_pass,<<"guest">>},
 {ssl_options,[],},
 {ssl_handshake_timeout,5000},
 {disk_monitor_failure_retry_interval,120000},
 {quorum_cluster_size,3},
 {release_series_eol_date,none},
 {num_tcp_acceptors,10},
 {config_entry_decoder,[{passphrase,undefined}]},
 {mirroring_flow_control,true},
 {password_hashing_module,rabbit_password_hashing_sha256},
 {autocluster,[{peer_discovery_backend,rabbit_peer_discovery_classic_config}]},
 {tcp_listeners,[{"auto",5672}]},
 {default_consumer_prefetch,{false,0}},
 {ranch_connection_max,infinity},
 {writer_gc_threshold,1000000000},
 {channel_max,2047},
 {vm_memory_high_watermark_paging_ratio,0.5},
 {default_user,<<"guest">>},
 {vhost_restart_strategy,continue},
 {msg_store_index_module,rabbit_msg_store_ets_index},
 {background_gc_enabled,false},
 {mnesia_table_loading_retry_timeout,30000},
 {vm_memory_high_watermark,0.4},
 {plugins_expand_dir,
  "c:/ProgramData/RabbitMQ/db/rabbit@OfeliaHA-SVR1-plugins-expand"},
 {memory_monitor_interval,2500},
 {ssl_apps,[asn1,crypto,public_key,ssl]},
 {disk_monitor_failure_retries,10},
 {track_auth_attempt_source,false},
 {collect_statistics,fine},
 {background_gc_target_interval,60000},
 {dead_letter_worker_publisher_confirm_timeout,180000},
 {channel_tick_interval,60000},
 {default_permissions,[<<".*">>,<<".*">>,<<".*">>]},
 {mnesia_table_loading_retry_limit,10},
 {stream_messages_soft_limit,256},
 {tcp_listen_options,
  [{backlog,128},{nodelay,true},{linger,{true,0}},{exit_on_close,false}]},
 {msg_store_io_batch_size,4096},
 {disk_free_limit,500000000},
 → {cluster_partition_handling,pause_minority}, ←
 {enabled_plugins_file,"c:/ProgramData/RabbitMQ/enabled_plugins"},
 {feature_flags_file,
  "c:/ProgramData/RabbitMQ/db/rabbit@OfeliaHA-SVR1-feature_flags"}]
```

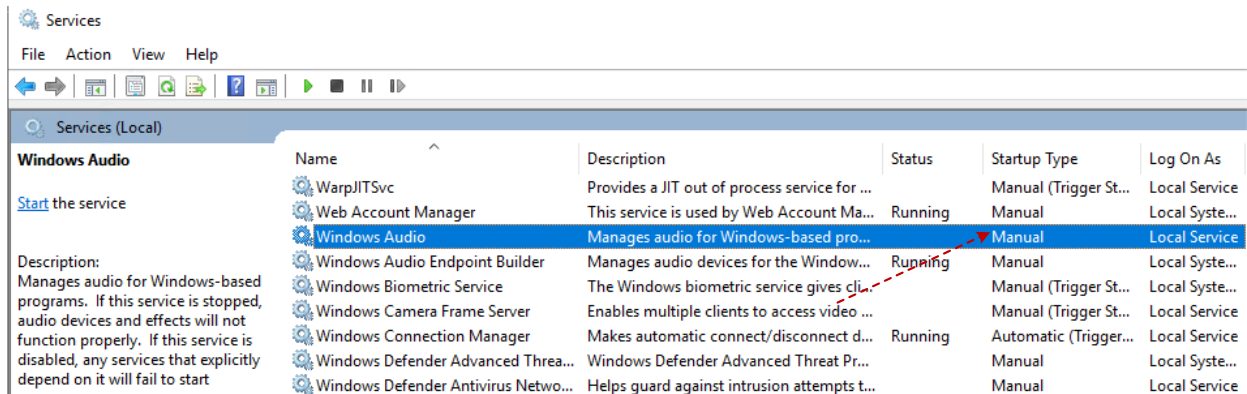
7. Enable Voice synthesis capabilities (Only Windows® platforms):

To generate text-to-speech messages, Ofelia uses the SAPI5 voices present in Windows editions.

7.1. Windows Audio service:

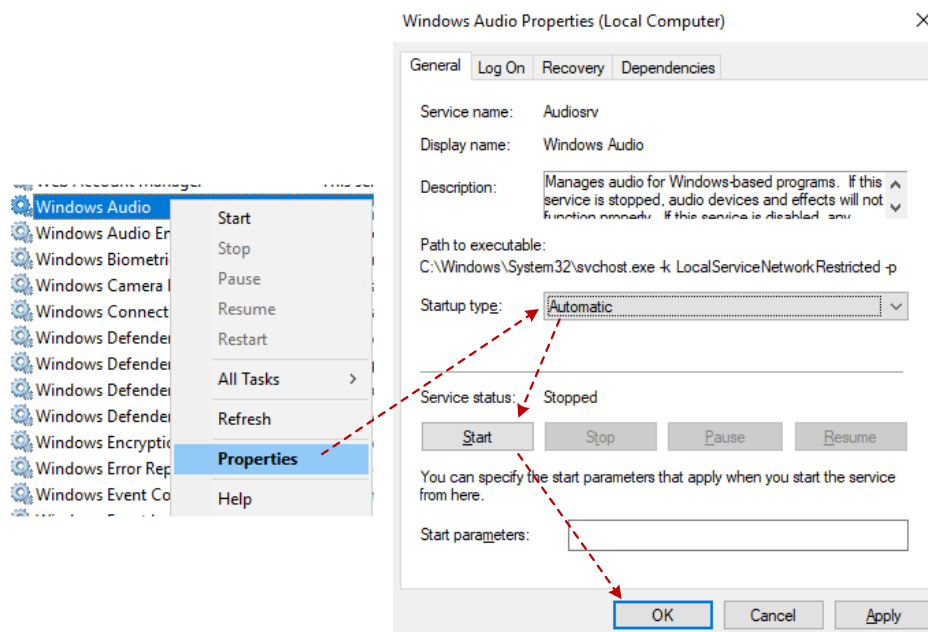
The first thing to check is that the Windows Audio service is configured to start automatically and is running.

By default, when Windows is installed on VMs (e.g. Windows® Server editions) this service is not configured in automatic start-up mode. You need to change this.

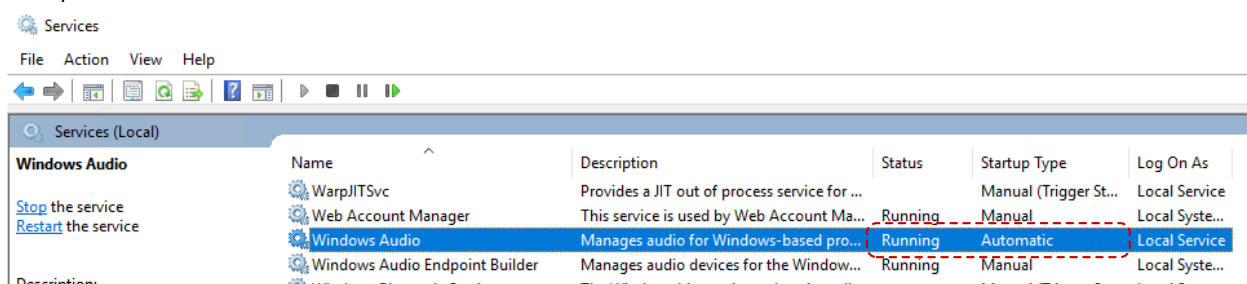


To change this, right-click on the "Windows Audio" service then "Properties":

- Change the '**Startup type**' to '**Automatic**'.
- Click on '**Start**' button.
- Then click on '**OK**' to save changes.



Expected result:

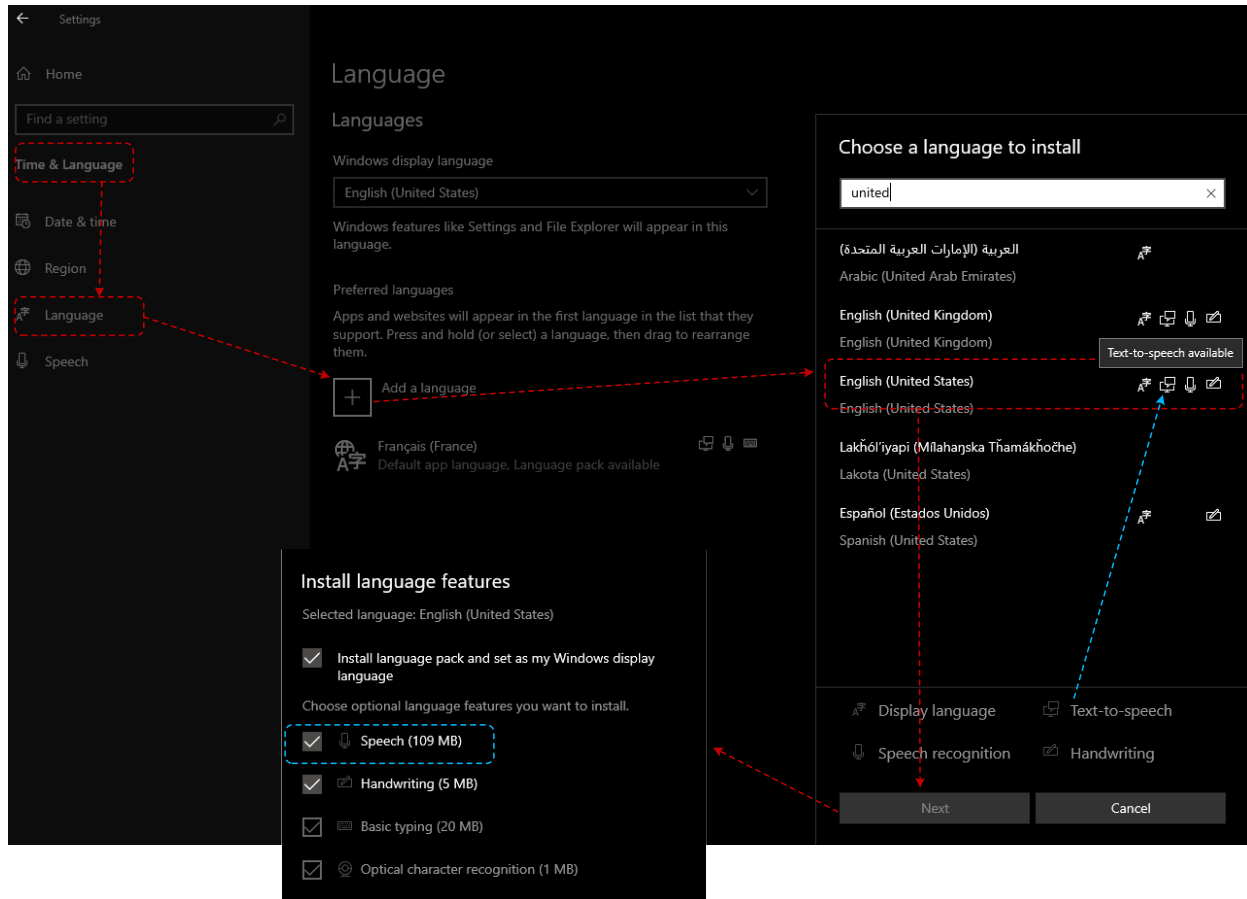


7.2. Voice package installed on Windows:

To ensure that the audio messages generated are in the correct language, you need to install the Windows language package corresponding to the desired language.



The method of adding Windows language packs and voice packages for text-to-speech capabilities may differ from one Windows edition to another. **Please refer to the Microsoft documentation for your edition of Windows®.**



Windows® Server 2019 example

8. Install SmartInspect Redistributable Console:

SmartInspect is a tool to exploit Ofelia system logs.

It will be very useful for your Ofelia configuration sessions, for real-time troubleshooting during your deployments, as well as for any ticket opening in case of problem on the software.

It is not a mandatory component for Ofelia, but it is strongly recommended to install it!



SmartInspect will only be usefull on servers where Ofelia application will be installed.

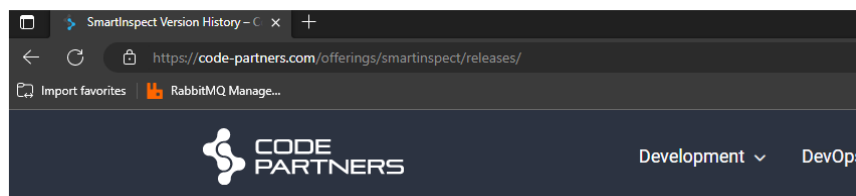
If Ofelia should be installed on multiple Windows Servers for scalability and high availability purposes, “SmartInspect” can be installed on each Ofelia Windows Server.

At the very least, the one on which you would like to carry out troobleshooting at a given time.

However, please note that this component is not required on servers running only Erlang OTP & RabbitMQ applications.

1. From Erlang website, download the SmartInspect “**Redistributable Console**” installation file.

➤ <https://code-partners.com/offerings/smartinspect/releases/>



SmartInspect

Version History

SmartInspect 3.4.1 Release

Changelog:

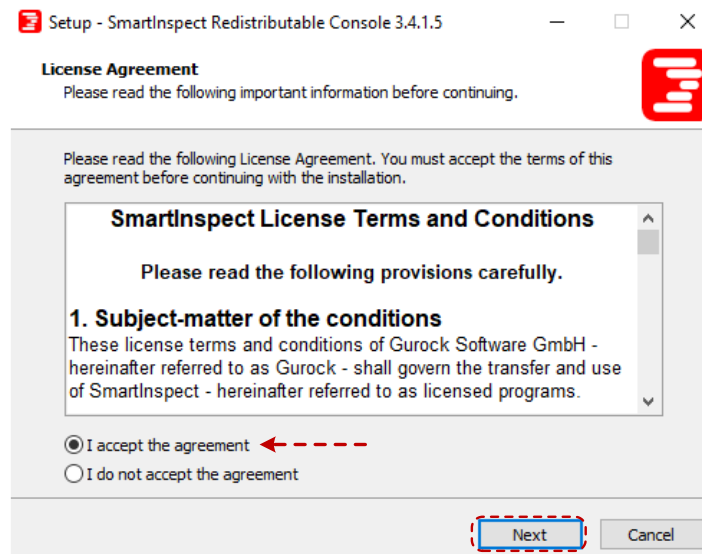
- BUG: Fix layout issues when running Builder in scaled environment
- BUG: Fix issues in Builder when protocols other than File or Text
- Enhancement: Add TPointF and TRectF support to FMX lib

Downloads

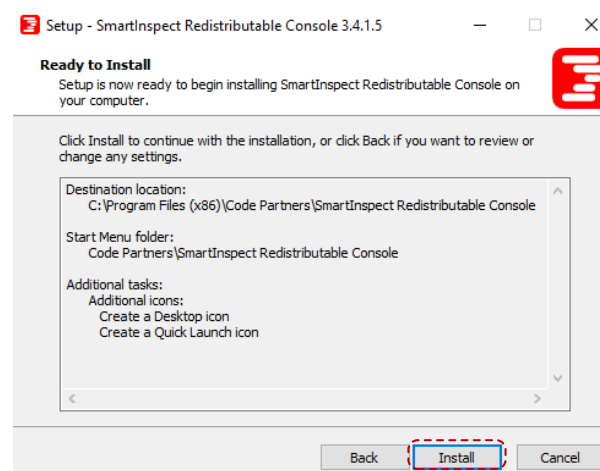
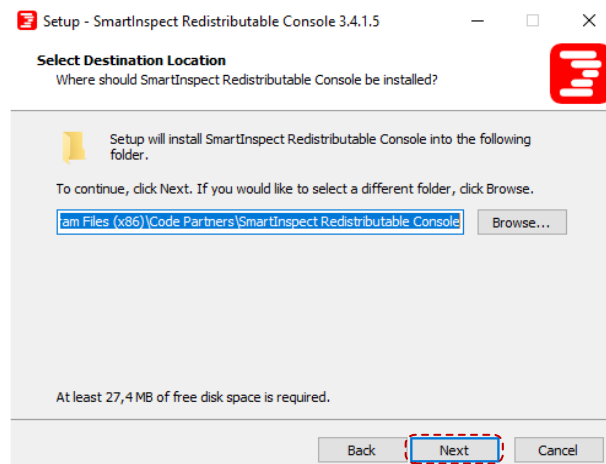
- SmartInspect Full Setup : [sipro-3.4.1.5.exe](#)
- **Redistributable Console** : [siredist-3.4.1.5.exe](#) ← ---
- SmartInspect Receiver : [sireceiver-3.4.1.5.exe](#)

2. Right-click the SmartInspect installation file and select **Run as administrator**.

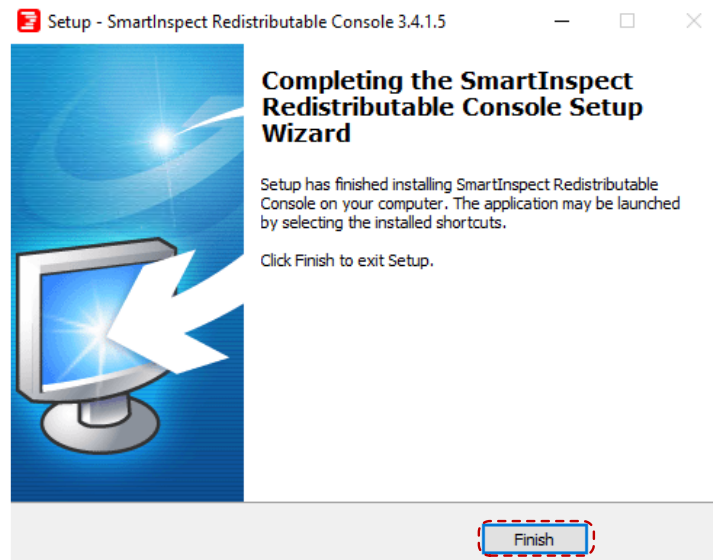
3. Read through the license agreement, select the I accept the agreement checkbox and click Next.



4. In each page of the installer, **leave all options by default**, simply click on **Next**, then on **Install**



5. Once the installation is complete, click on **Finish**.



9. Ofelia Setup Program:

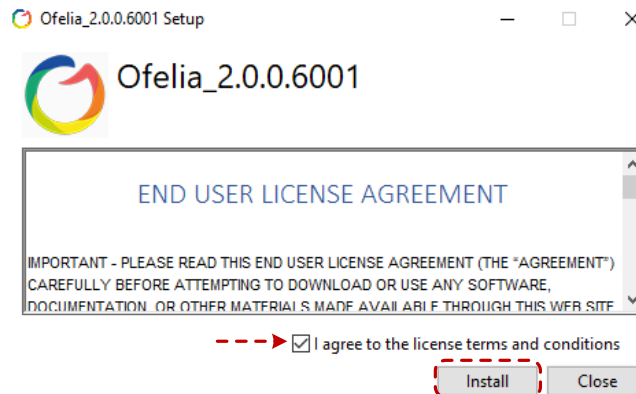


Ofelia's installation program does not create the database on the SQL server.

The database is created:

- Automatically when Ofelia is first launched.
- Or beforehand by the customer's IT department.

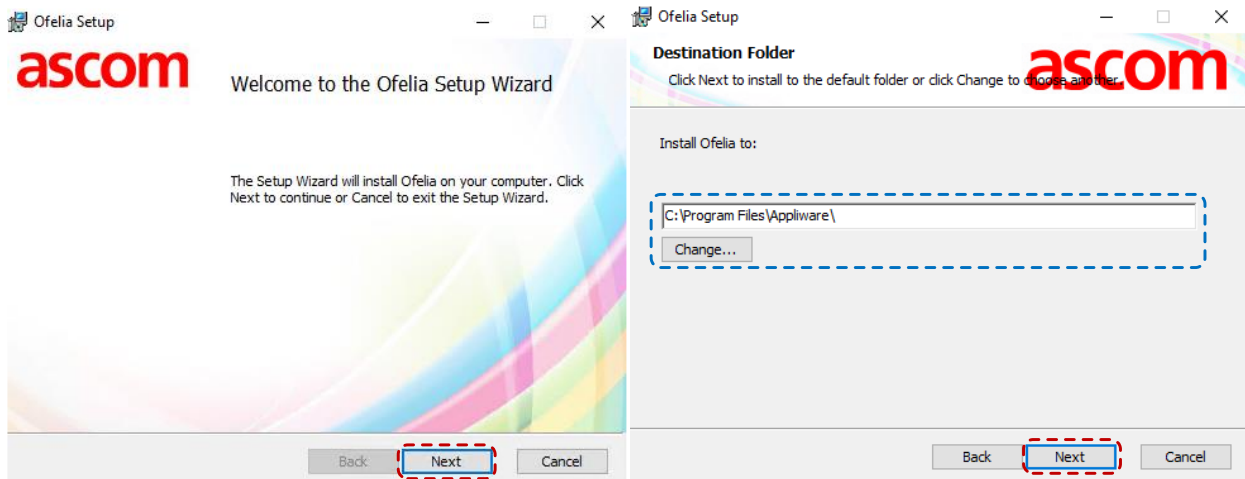
1. Right-click on the setup file and select "Run as administrator".
2. Read through the end user license agreement, check the box to agree to the license terms and conditions, then click on **Install** button.



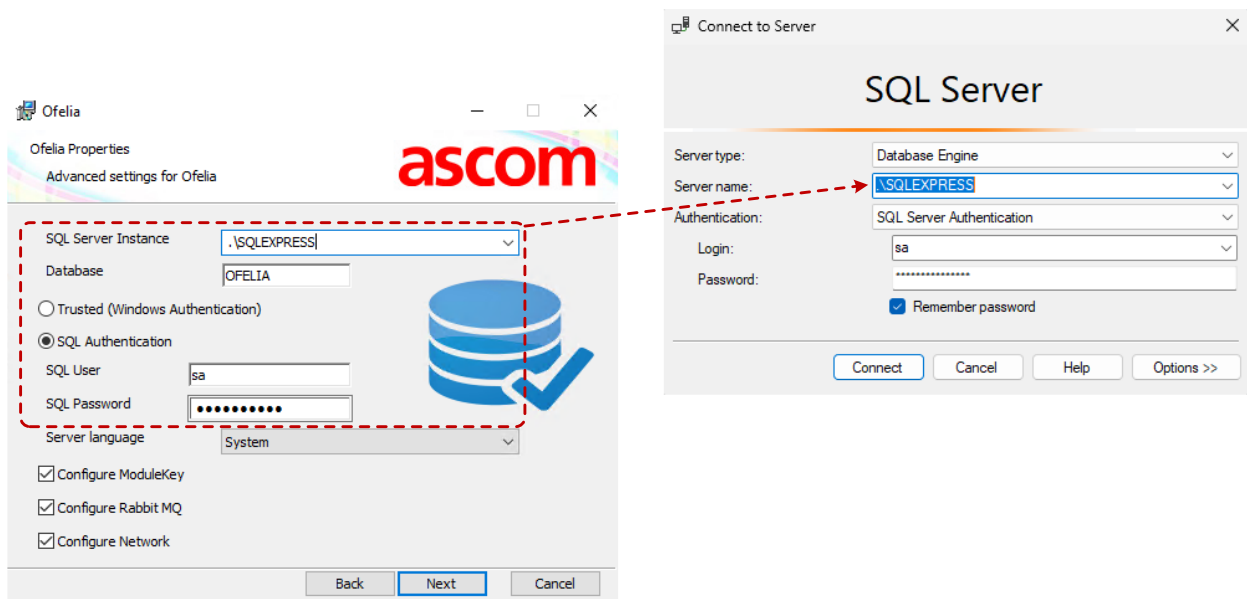
3. Click **Next**, adjust Ofelia installation directory, if necessary, then click **Next** again.

9.1. Ofelia installation directory:

Adjust Ofelia installation directory, if necessary, then click **Next** again.



9.2. Configure the SQL Server connection:

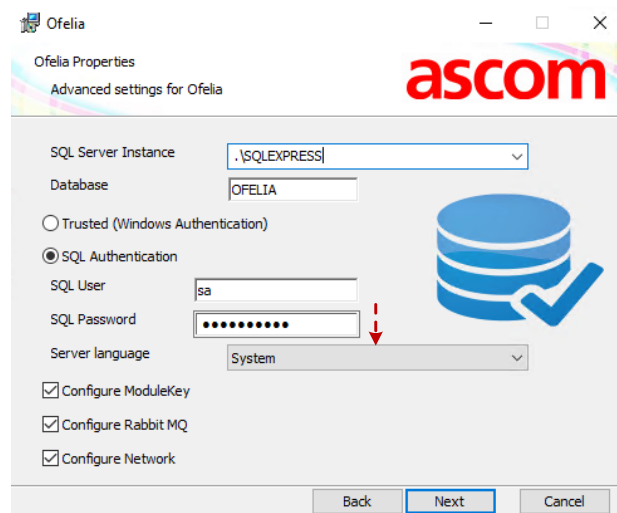


1. To be sure of what you need to enter as the SQL server instance name in the Ofelia installation program, you can take the example of the "Server Name" used in Microsoft's SQL Server Management Studio tool to connect to the SQL server instance.
2. Adjust, if necessary, the database name to be used.
3. Then select the authentication mode on SQL Server instance.



For more details, refer to chapter [Ofelia SQL connection configuration possibilities](#)

9.3. Select the appropriate server language.



WARNING: This parameter is important because it defines the language in which all variables, default messages and event names will be automatically created when Ofelia is first launched. Once the database has been initialized when Ofelia is first launched, all default texts will have to be translated manually....

9.4. Configure the parameters related to the "Elise3 Module Key" licensing method:

By checking the "Configure Module Key" box, an additional page will be added to the installation process.

Module key configuration is required when the Ofelia license is based on an Elise 3 card key module.

➔ **When the Ofelia license is based on an environment code, it is not necessary to fill in this section (it is not necessary to check this box).**



Note:

Virtual Module Key of vNISM and UnitePS are not supported.

The screenshot shows two windows from the Ofelia installation wizard. The first window, titled 'Ofelia Properties', shows the 'Advanced settings for Ofelia' tab. It has fields for 'SQL Server Instance' (set to '.\SQLEXPRESS'), 'Database' (set to 'OFELIA'), and authentication options. The 'SQL Authentication' radio button is selected, with 'SQL User' set to 'sa' and 'SQL Password' masked with dots. The 'Server language' is set to 'System'. At the bottom, three checkboxes are checked: 'Configure ModuleKey', 'Configure Rabbit MQ', and 'Configure Network'. A red dashed box highlights the 'Configure ModuleKey' checkbox, with an arrow pointing to the second window. The second window, titled 'Ofelia', is the 'Module Key security' screen. It prompts the user to 'Please enter one or two ModuleKey'. It has fields for 'Login' (set to 'admin') and 'Password' (masked with dots). Below these are two rows of fields for 'ModuleKey' and 'Address'. The first row has 'ModuleKey' set to '00123456' and 'Address' set to '192.168.0.80'. The second row is empty. At the bottom, there are radio buttons for 'Use Http' (selected) and 'Use Https'. A red dashed box highlights the 'ModuleKey' and 'Address' fields. A large red key icon is on the right side of the window.

This page lets you define:

- The WEB **login** of the Elise3 module on which the license is to be checked.
- The **module key(s)** and their corresponding **IP addresses**.
- The WEB **method** of the Elise3 module for checking the module Key (*same as for accessing the Elise 3 module WEB page - default http*).

9.5. Configure RabbitMQ Connection settings:

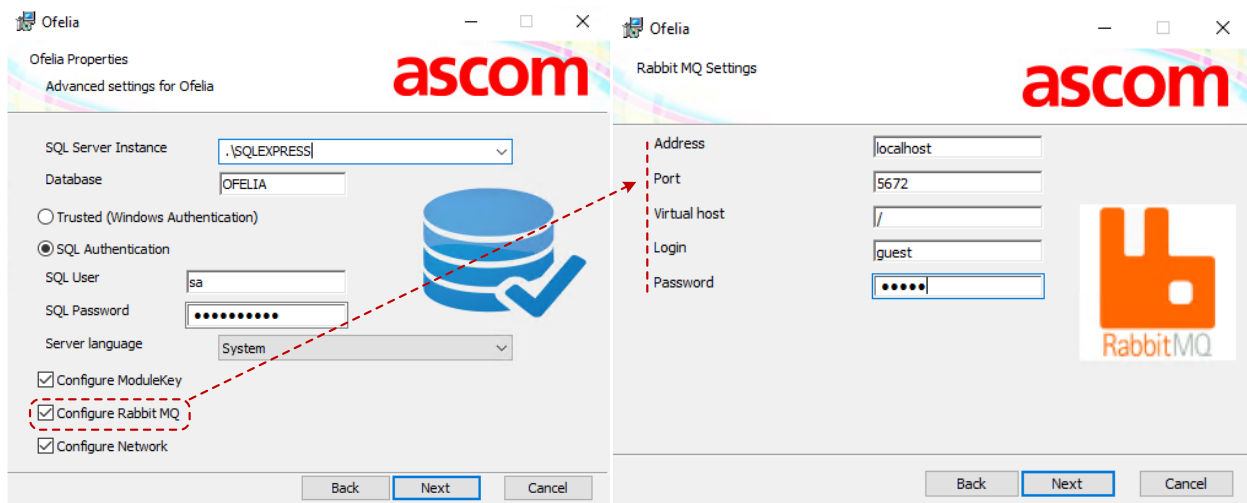
By checking the "Configure RabbitMQ" box, an additional page will be added to the installation process.

➔ For small footprint installations, where RabbitMQ is often installed on the same machine as Ofelia, it is not necessary to tick this box. The settings can be left as default.

RabbitMQ configuration is required when:

- RabbitMQ is not installed on the same machine as Ofelia.
- RabbitMQ has been configured to use a port other than its default port (TCP 5672)
- RabbitMQ must use a different Virtual Host than the default (/)
- RabbitMQ must use a different login than the default one (guest/guest)

Clustered installations of RabbitMQ (*infrastructure with HA*) are therefore inevitably concerned, as there will be several nodes to declare, and a user account other than the default one (*the guest account can only be used on localhost...*).



- **Hostname:** enter the hostname (FQDN) used for communication with the RabbitMQ Server(s):
 - When installing as single node, with Ofelia installed on the same machine, it is possible to enter < **localhost** > as hostname.
 - When installing in a cluster, enter the hostnames of all RabbitMQ nodes in this box. Separate the names with a comma, such as < **Node1,Node2,Node3** >, and so on.

RabbitMQ service must be installed and running on at least one node for this installation step to pass.

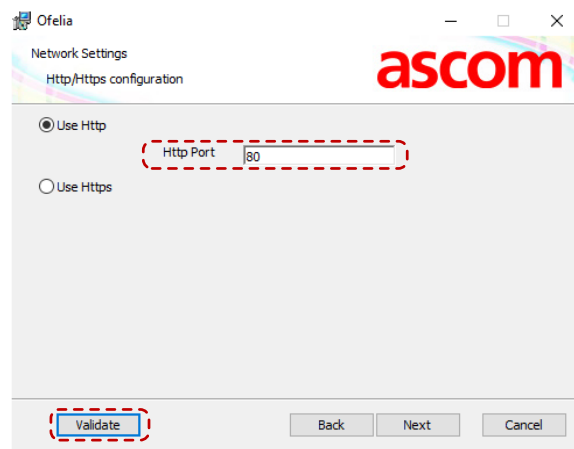
- **Port:** enter the port for communicating with the RabbitMQ servers (default: 5672).
- **Virtual host:** enter the virtual host that Ofelia should use. If it does not exist, Ofelia will try to create it.
- **Login & Password:** Enter the username and password to access the RabbitMQ servers.

9.6. Defining access to Ofelia (Http or Https connection):

By checking the **"Configure Network"** box, an additional page will be added to the installation process.

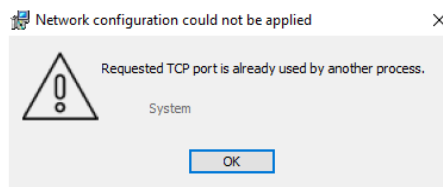
9.6.1. Http connection:

To use Http connection, simply enter the port to be used to connect to Ofelia (*WEB connection, OfeliaBrowser and OfeliaSmart*).



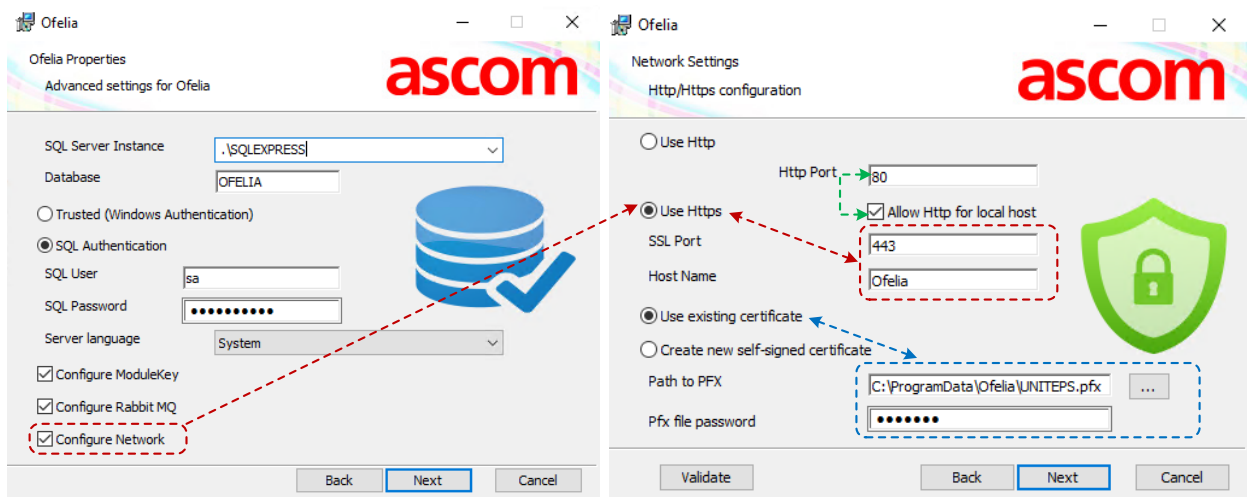
Make sure the port is free by clicking on **"Validate"** button:

- If no message appears, the configured port appears to be free, and the installation can continue.
- If an error message appear is that the port is already being used by another process and is not free for Ofelia.



9.6.2. Https connection:

To be able to activate the Https connection you must have a certificate signed by a trusted authority.



In this example, two types of connection are allowed:

- http (port 80) from the Ofelia server itself. (*localhost*)
- in https (port 443) from any other machine.

"Path to PFX": Defines the path where is stored the pfx file.

"PFX file password": Defines the password for the certificate.



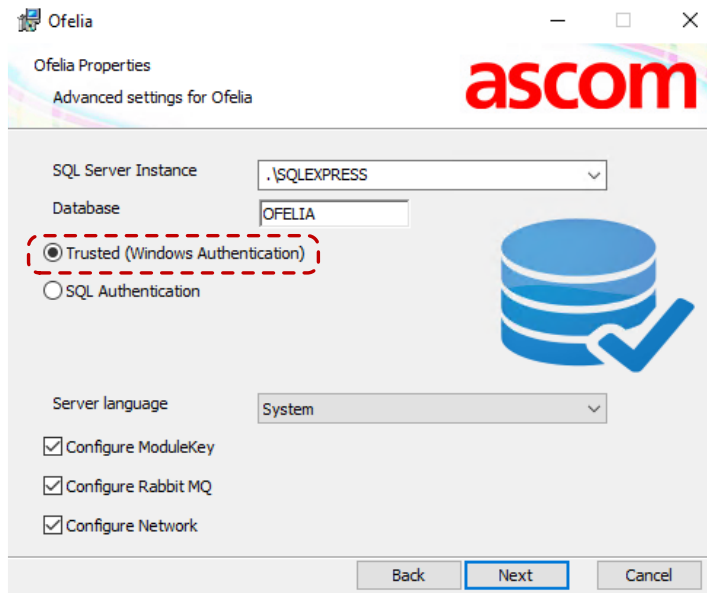
The connection method can also be adjusted after installation, refer to chapter *"Appendix A – Change the WEB connection method to Ofelia after installation"* further in this document.

10.Ofelia SQL connection configuration possibilities:

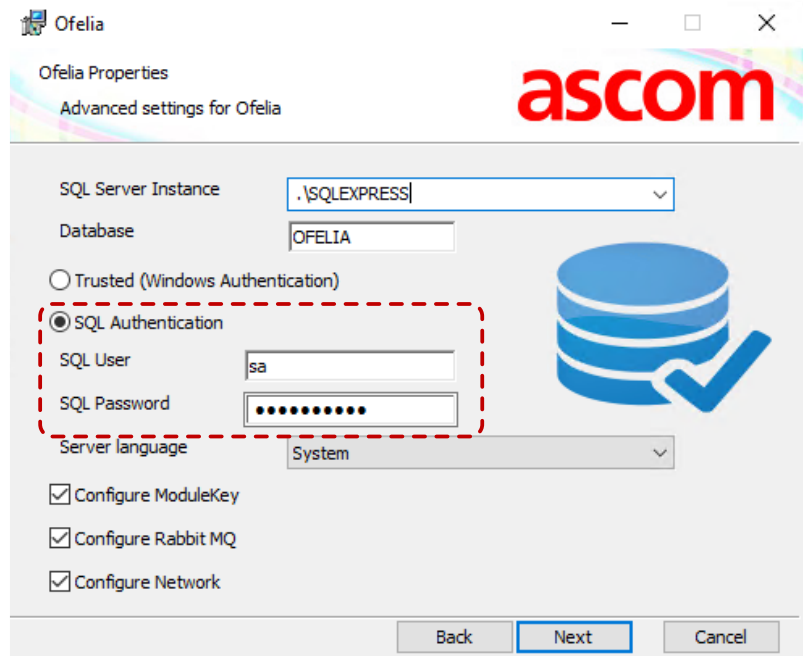
The database access strategy can be modified according to the needs of the solution architecture, according to your preferences,

Three solutions are possible, through:

- Using default Local System account (*Only when MSSQL is installed on the same machine*)
- Using a specific Windows account (***Strongly recommended solution for security reasons***)
- Using a built-in MS-SQL account.



Configuration example in Ofelia installation program with a Windows account (Local System or specific)



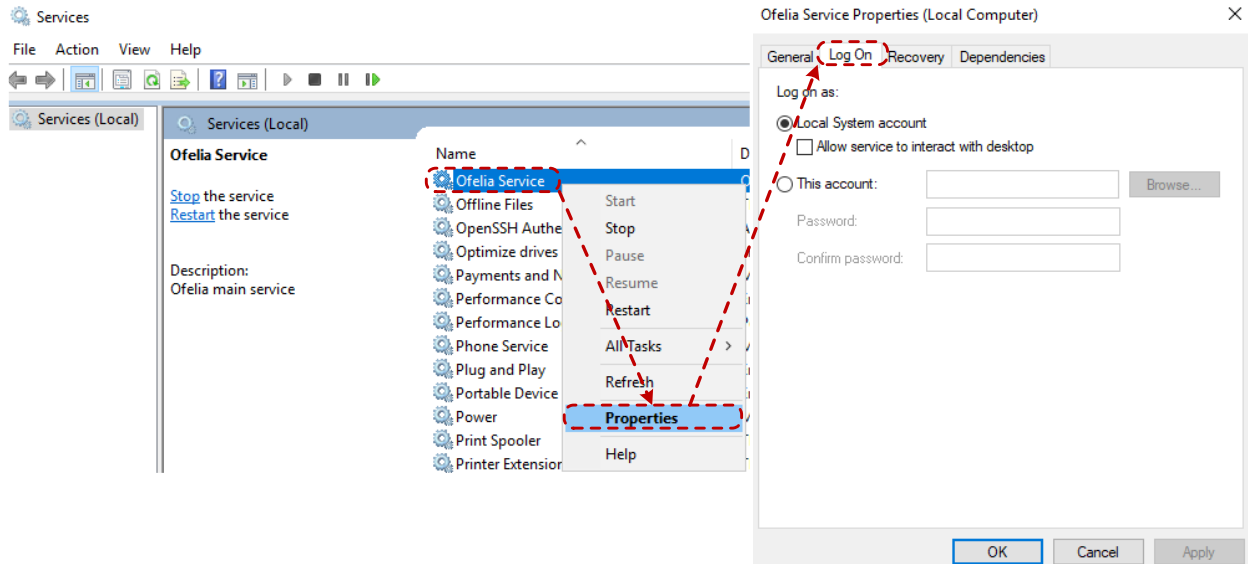
Configuration example in Ofelia installation program with a built-in MS-SQL account.

10.1. Using default Local System account (Recommended for Small footprint installation):

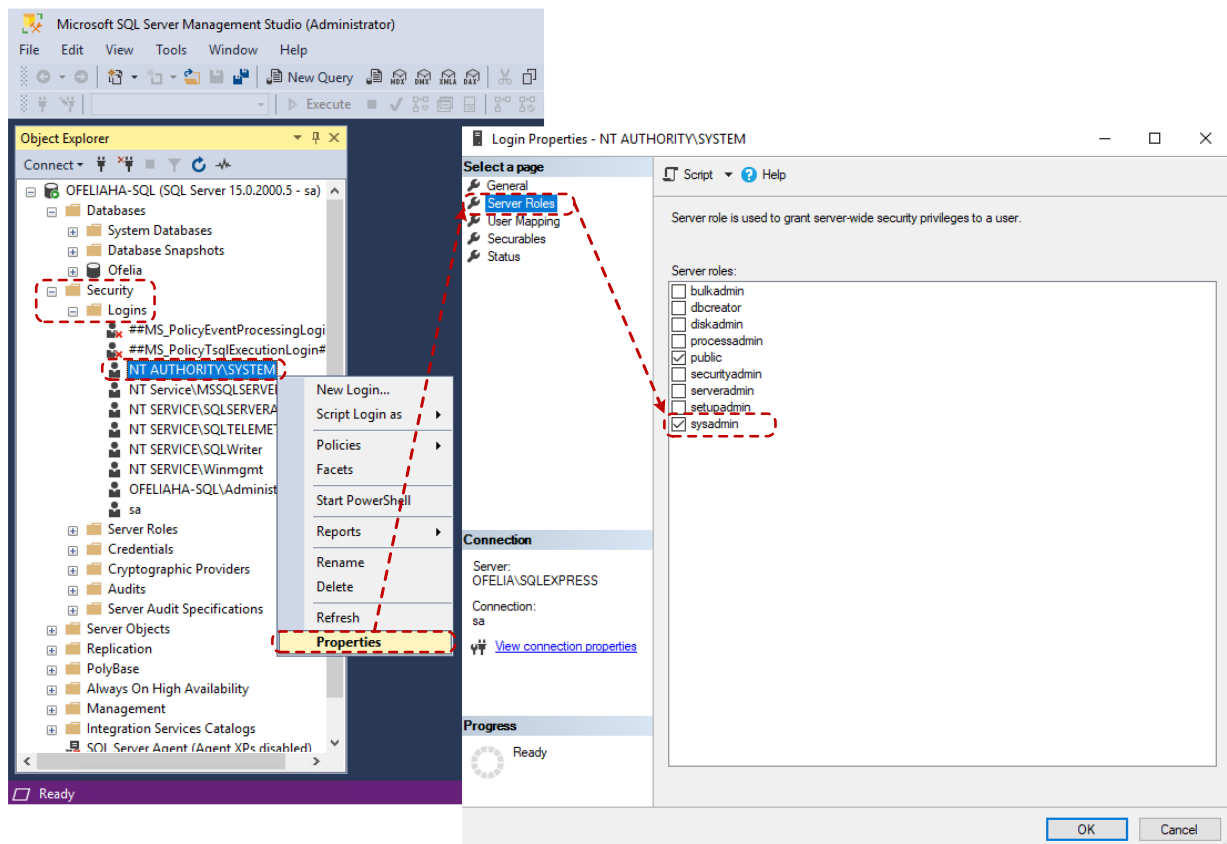
This configuration involves two parts:

- The first step is to add MS SQL rights to the Windows System Local account.
- The second is to configure Ofelia service to use this Windows account. (See example chapter 10.1 Adaptation of the SQL database access method)

By default, Ofelia runs as a Windows service, using the “**System Local**” account.



In this case, it is necessary to add “**sysadmin**” rights on the SQL server for the System Local account **before Ofelia is startup for the first time** (“NT AUTHORITY\SYSTEM”).

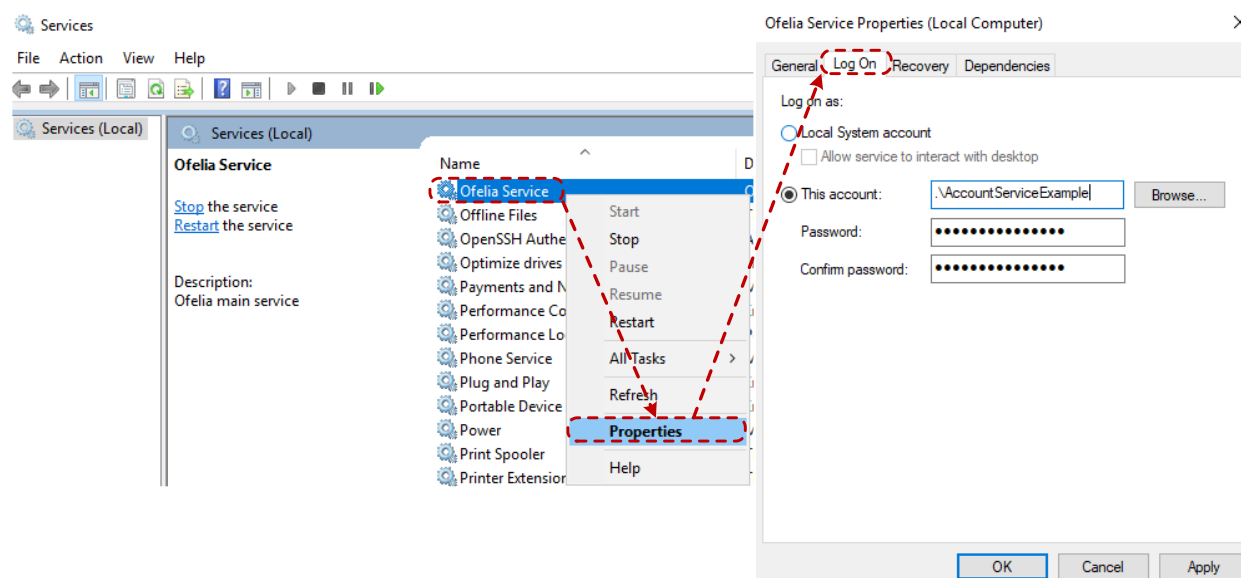


10.2. Using a specific Windows account (Recommended):

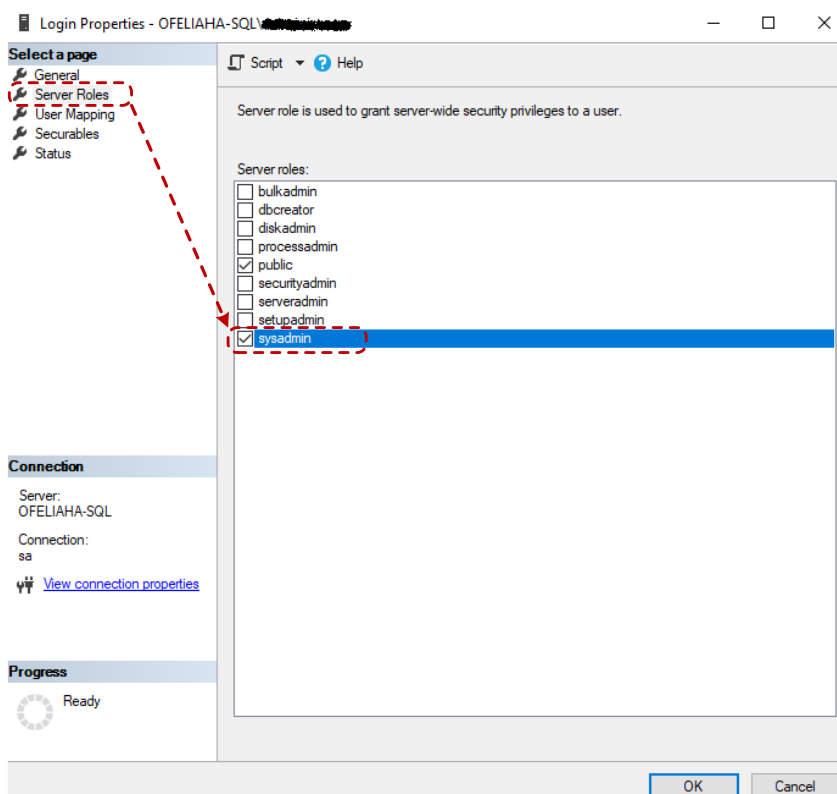
This configuration involves two parts:

- The first step is to add MS SQL rights to the Windows account concerned.
- The second is to configure Ofelia service to use this Windows account. (See example chapter 10.1 *Adaptation of the SQL database access method*)

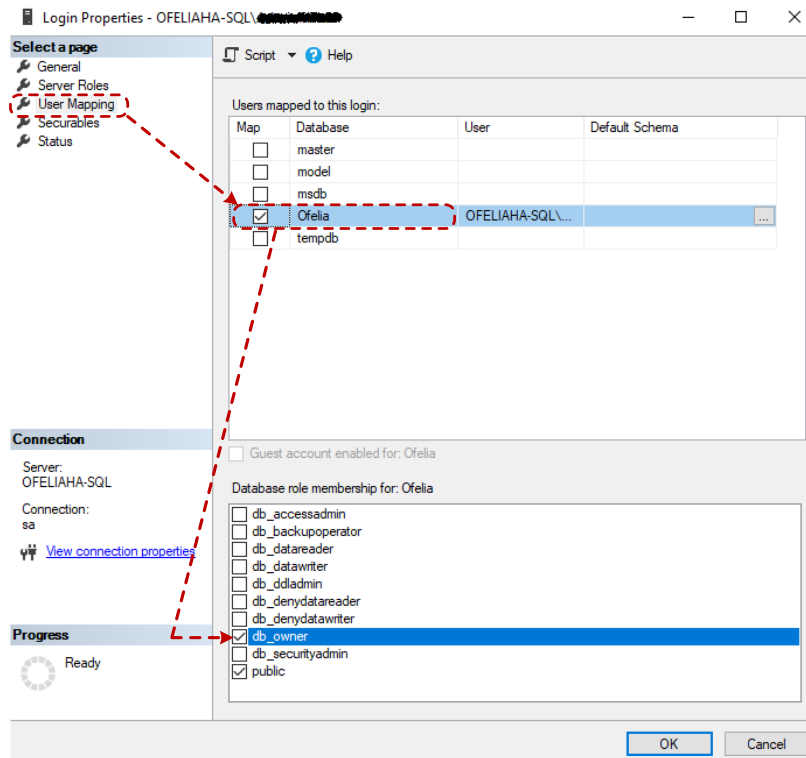
A specific Windows account to interact with the database, can be the user account used when installing MS SQL or any other Windows account with appropriate rights on the SQL server.



Case 1: If Ofelia need create the database on the MS SQL server automatically during the first startup, the Windows account to be used by Ofelia's Service must have at least "**dbcreator**" rights on MS SQL server **before the first Ofelia startup** (Having "**sysadmin**" rights, even temporarily during the first start of Ofelia, might also be a good option).



Case 2: If the database is created manually by the IT department, the Windows account to be used by Ofelia's Service must have "**db_owner**" rights on this database **before the first Ofelia startup** (and must keep this right permanently).



In both cases, it is strongly recommended that the Windows account used **has a non-expiring password**. (Using a "service account" may be a more appropriate choice, as a "standard user account" may be required to change its password regularly).

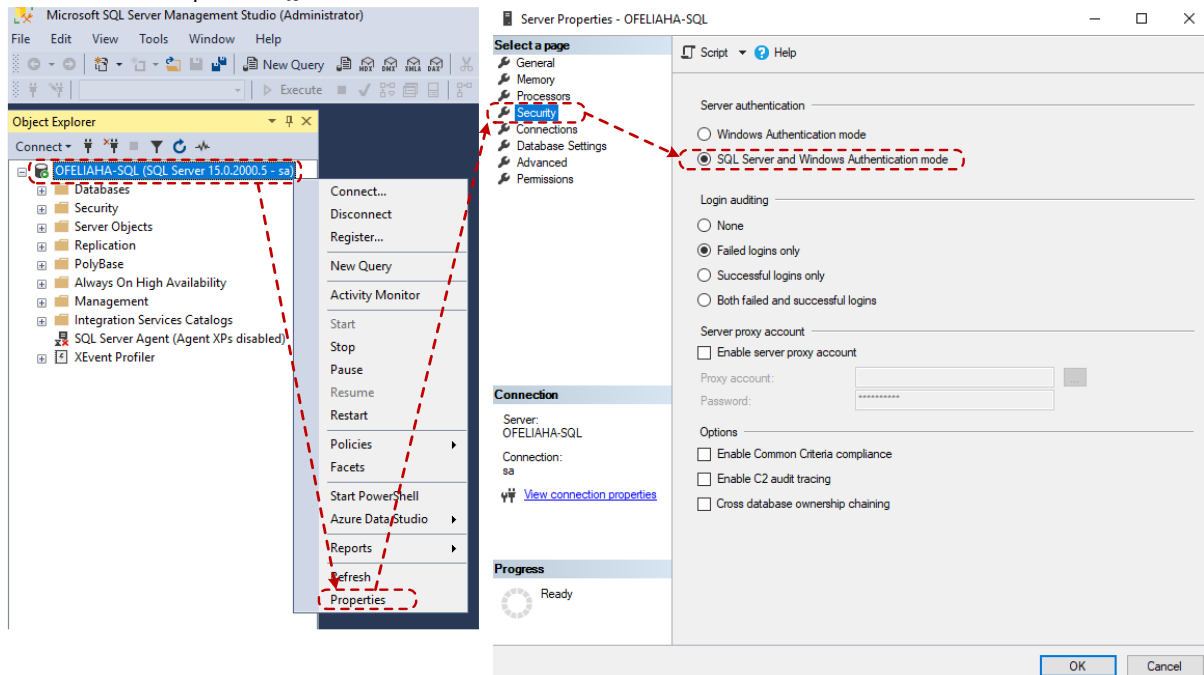
10.3. Using a built-in MS-SQL account:

This configuration involves two steps:

- The first step is to create a built-in SQL account.
- The second is to configure Ofelia to use this built-in SQL account. (See example chapter 10.1 Adaptation of the SQL database access method)

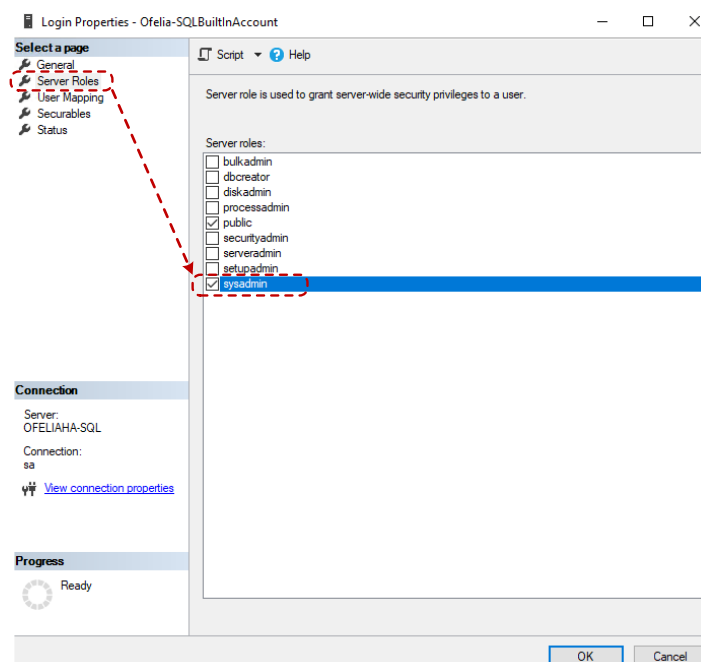
1. Make sure your SQL server supports both SQL server and Windows authentication modes:

To do this, with SQL Management studio, access the server properties, then under the "**Security**" menu, check that the option "**SQL server and Windows authentication mode**" is selected.

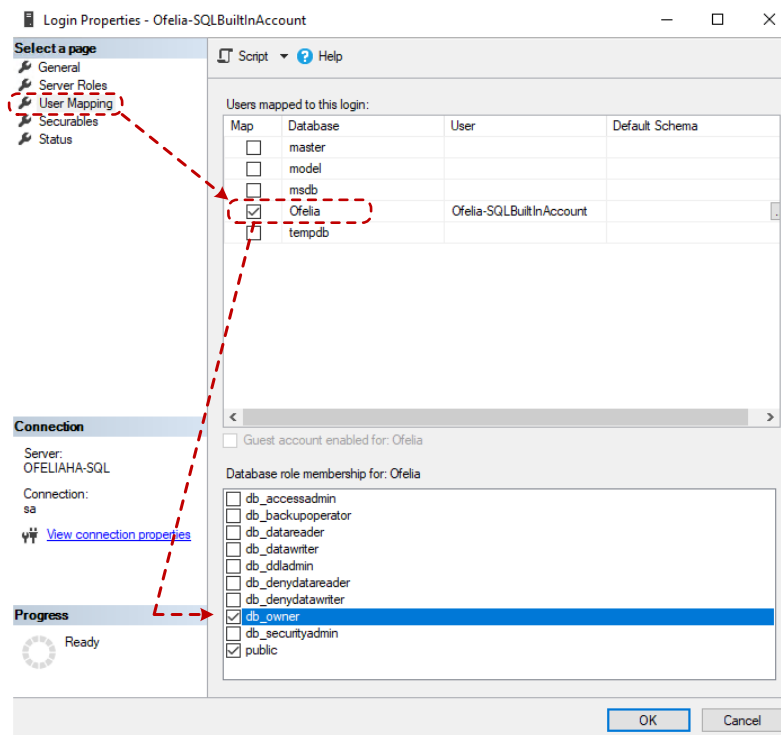


2. Make sure that the MS SL integrated account created for Ofelia has the good rights.

If Ofelia need create the database on the MS SQL server automatically during the first startup, the built-in MS SQL account to be used by Ofelia's Service must have at least "**dbcreator**" rights on MS SQL server **before the first Ofelia startup** (Having "**sysadmin**" rights, even temporarily during the first start of Ofelia, might also be a good option).



If the database is created manually by the IT department, the built-in MS SQL account to be used by Ofelia must have "**db_owner**" rights on this database **before the first Ofelia startup** (and must keep this right permanently).



11. Appendix A - Adjust Ofelia's configuration files outside the installer:

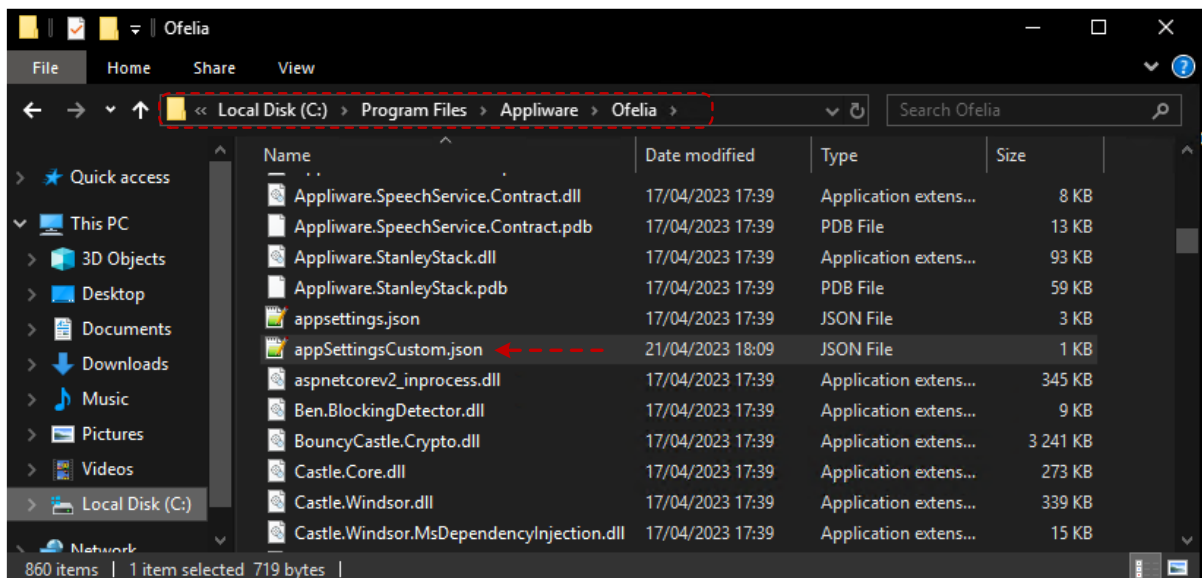
These settings are usually defined in the Ofelia installation program, so it is not necessary to modify these configuration files manually.

However, for troubleshooting or modification purposes, it may be useful to know how to modify these settings without having to uninstall and reinstall everything.

However, it is also possible to adjust these parameters by editing the Ofelia configuration file.

11.1. Adjust Ofelia's SQL connection settings:

1. Stop the Ofelia application completely.
2. With Windows explorer, **open** the Ofelia installation directory, by default:
C:\Program Files\Appliware\Ofelia
3. **Find** the file **<appSettingsCustom.json>**



4. **Edit** this file with a text editor and locate the line with “Ofelia” connection string.

```
{
  "Kestrel": {
    "Endpoints": {
      "Http": {
        "Url": "http://*:80/"
      }
    }
  },
  "AppSettings": {
    "ConnectionStrings": {
      "Ofelia": "Data Source=.\SQLEXPRESS;Initial Catalog=OFELIA;Integrated Security=true;",
      "FileLog": "tcp(reconnect=true, reconnect.interval=5s), file(filename=C:\\ProgramData\\Appliware\\Ofelia\\FileLog.txt)",
      "RabbitMQ": "host=localhost;virtualHost=Ofelia;timeout=120"
    },
    "ModuleKey": {},
    "NoOuranosCom": false,
    "MultiTenants": true,
    "MultiTenantsDisableAutoRefresh": false,
    "DisableAutoBackup": false,
    "CreateRabbitMQVirtualHost": true
  }
}
```

- Adjust the "Ofelia" connection information with the right information.

"Ofelia": "Data Source=<SvrName>; Initial Catalog=<DBName>;Integrated Security=true;",
 "Ofelia": "Data Source=<SvrName>; Initial Catalog=<DBName>;user id=<User>;password=<password>;",

Where, each parameters separated semicolons (";");

- <SvrName> is the name of SQL Server used to connect to SQL Cluster.
- <DBName> is the Database's name define for Ofelia in SQL.

Using Windows account:

The part of the connection string no longer contains the parameters user id and password. This indicates that integrated security is in use.

"Ofelia": "Data Source=OFELIAHA-SQL;Initial Catalog=Ofelia;Integrated Security=true;",

```
{
  "Kestrel": {
    "Endpoints": {
      "Http": {
        "Url": "http://*:80/"
      }
    }
  },
  "AppSettings": {
    "ConnectionStrings": {
      "Ofelia": "Data Source=OFELIAHA-SQL;Initial Catalog=Ofelia;Integrated Security=true;",
      "FileLog": "tcp(reconnect=true, reconnect.interval=5s), file(filename=C:\\ProgramData\\Appliware\\",
      "RabbitMQ": "host=localhost;virtualHost=Ofelia;timeout=120"
    },
    "ModuleKey": {},
    "NoOuranosCom": false,
    "MultiTenants": true,
    "MultiTenantsDisableAutoRefresh": false,
    "DisableAutoBackup": false,
    "CreateRabbitMQVirtualHost": true
  }
}
```

Using a built-in MS-SQL account:

- <User> is the built-in MS SQL user account.
- <password> is the built-in MS SQL user password.

"Ofelia": "Data Source=OFELIAHA-SQL;Initial Catalog=Ofelia;user id=Ofelia-SQLBuiltInAccount;password=AscomOfelia1234;",

```
{
  "Kestrel": {
    "Endpoints": {
      "Http": {
        "Url": "http://*:80/"
      }
    }
  },
  "AppSettings": {
    "ConnectionStrings": {
      "Ofelia": "Data Source=OFELIAHA-SQL;Initial Catalog=Ofelia;user id=Ofelia-SQLBuiltInAccount;password=AscomOfelia1234;",
      "FileLog": "tcp(reconnect=true, reconnect.interval=5s), file(filename=C:\\ProgramData\\Appliware\\Ouranos\\log.sil,level=\"",
      "RabbitMQ": "host=localhost;virtualHost=Ofelia;timeout=120"
    },
    "ModuleKey": {},
    "NoOuranosCom": false,
    "MultiTenants": true,
    "MultiTenantsDisableAutoRefresh": false,
    "DisableAutoBackup": false,
    "CreateRabbitMQVirtualHost": true
  }
}
```

- Save changes.
- Restart Ofelia application.



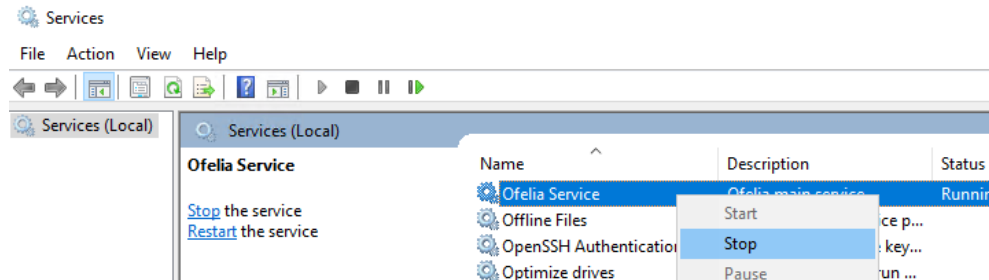
Note: You can check that the content of your Json file is correct (no missing brackets, commas, etc.) using online verification tools, such as <https://jsonlint.com/>

11.2. Adjust Ofelia's RabbitMQ connection settings:

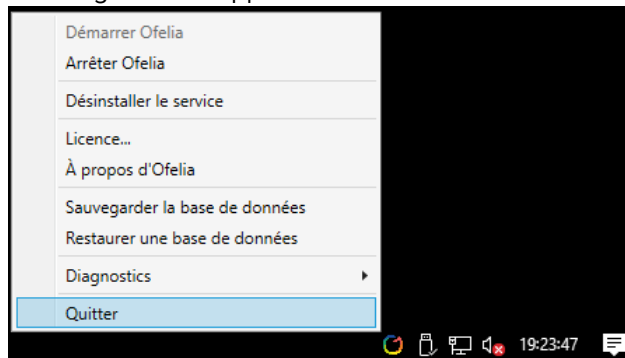


In a clustered architecture, if manual adjustment needs to be done, they must be performed for each Ofelia node.

1. Click the Windows Start button, type "**Services**", and click **Enter**.
 - The Services window appears.
2. Right-clicking on **<Ofelia Service>** service, then click on **Stop**.

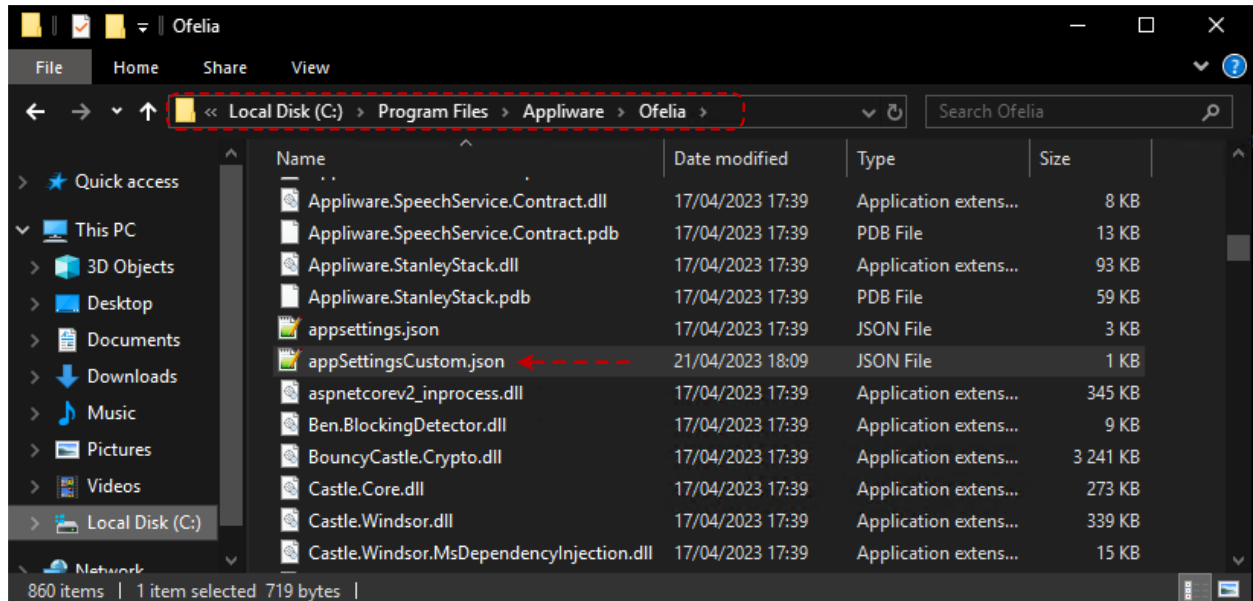


3. **On the System Tray**, if the Ofelia icon is still present, right-click on it and click "close".
 - ➔ Answer "YES" to the dialog box that appears.

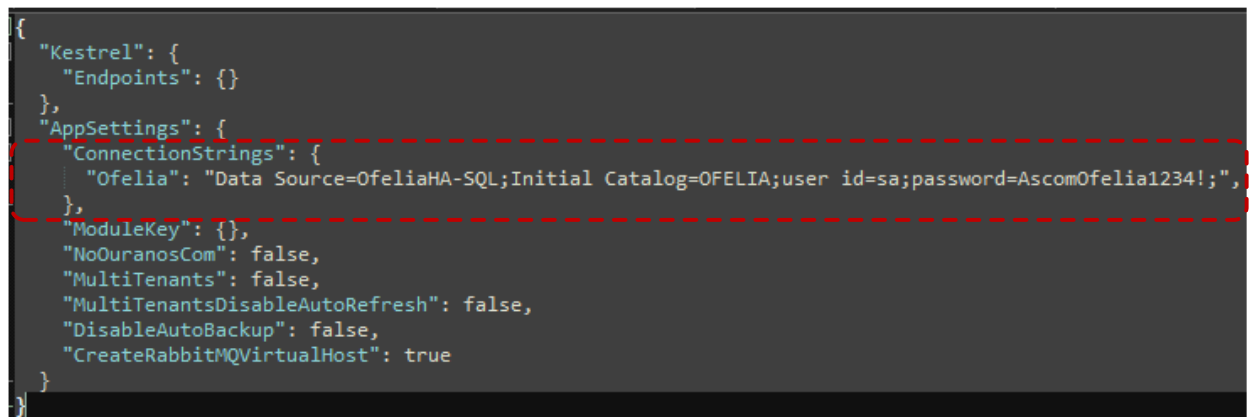


4. With Windows explorer, **open** the Ofelia installation directory, by default:
C:\Program Files\Appliware\Ofelia

5. Find the file <appSettingsCustom.json>



6. Edit this file with a text editor and locate the part about “ConnectionsStrings” settings.



7. **Insert a line** for the RabbitMQ connection information with the right information.

"RabbitMQ": "host=<node1>,<node2>,<node3>;virtualHost=<vhost>;username=<user>;password=<password>;timeout=120"

Where, each parameters separated semicolons (";"):

- <nodeX>** is the name of each node, usually it is the hostname. *(Separated by commas ",")*
- <vhost>** is the vHost's id created in previous step.
- <user>** is the RabbitMQ user's id created in previous step.
- <password>** is the RabbitMQ user's password created in previous step.

Example in context:

```
{
  "Kestrel": {
    "Endpoints": {
      "Http": {
        "Url": "http://*:80/"
      }
    }
  },
  "AppSettings": {
    "ConnectionStrings": {
      "Ofelia": "Data Source=.\SQLEXPRESS;Initial Catalog=OFELIA;Integrated Security=true;",
      "FileLog": "tcp(reconnect=true, reconnect.interval=5s), file(filename=C:\\ProgramData\\Appliware\\Ouranos\\log.sil,level=\"verbose\", maxp",
      "RabbitMQ": "host=OfeliaHA-SVR1,OfeliaHA-SVR2,OfeliaHA-SVR3;virtualHost=Ofelia;username=OfeliaSystemUser;password=OfeliaSystemPassword";
    },
    "ModuleKey": {},
    "NoOuranosCom": false,
    "Multitenants": true,
    "MultiTenantsDisableAutoRefresh": false,
    "DisableAutoBackup": false,
    "CreateRabbitMQVirtualHost": true
  }
}
```

8. **Add a comma (",")** at the end of the line preceding the one you've just added.
9. **Save** changes.
10. **Restart** Ofelia application with the "Ofelia Server" icon on the desktop



Additional notes:

- If the default vHost of rabbitMQ is used ("/"), it is possible to:
 - Use "virtualHost=/"

"RabbitMQ":

"host=<node1>,<node2>,<node3>;virtualHost=/;username=<user>;password=<password>;timeout=120"

- Delete the "virtualHost" part of the connection string:

"RabbitMQ": "host=<node1>,<node2>,<node3>;username=<user>;password=<password>;timeout=120"

- The RabbitMQ connection string should not end with a semicolon (";")
For this connection string, the semicolons (";") are only used as separators between the different types of parameters, but not as an end-of-string marker.



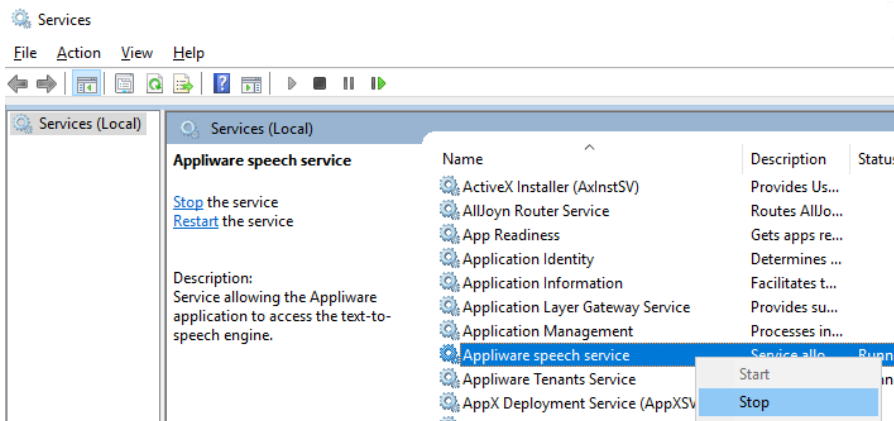
You can check that the content of your Json file is correct (no missing brackets, commas, etc.) using online verification tools, such as <https://jsonlint.com/>

11.3. Adjust AppliwareSpeech's RabbitMQ connection settings:

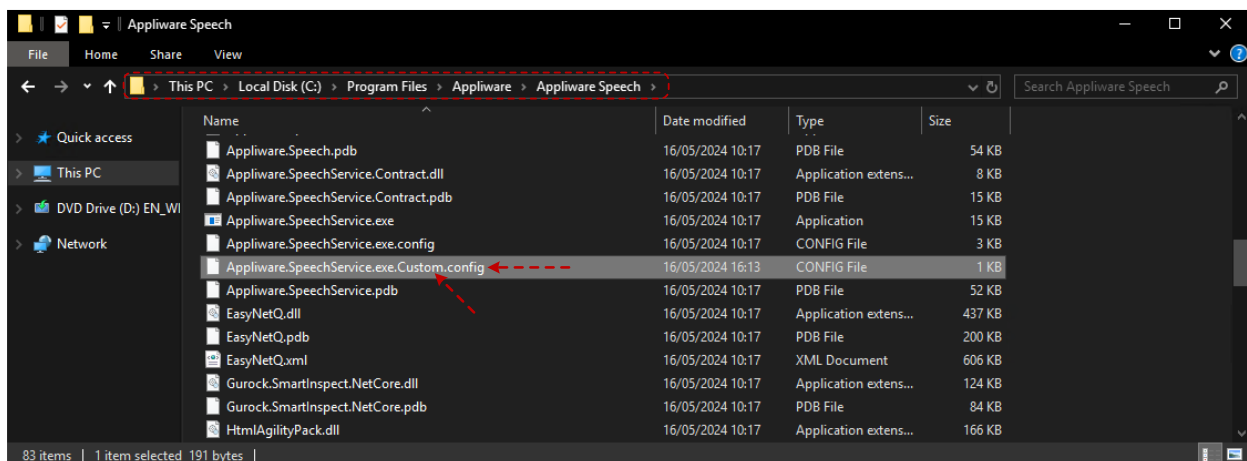
Appliware Speech is an application installed at the same time as Ofelia Server, on the same machine. Ofelia Server relies on it to generate notifications using Microsoft's text-to-speech to compatible media (e.g., SIP Phone)

 In a clustered architecture, if manual adjustment needs to be done, they must be performed for each Ofelia node.

1. Click the Windows Start button, type "**Services**", and click **Enter**.
➤ The Services window appears.
2. Right-clicking on **<Appliware speech service>** service, then click on **Stop**.



3. With Windows explorer, **open** the Appliware Speech installation directory, by default:
C:\Program Files\Appliware\Appliware Speech
4. **Find** the file **<Appliware.SpeechService.exe.Custom.config>**



5. **Edit** this file with a text editor and locate the line with RabbitMQ connection string.

```
1 <configuration>
2 <appSettings>
3 <add key="RabbitMQ.ConnectionString" value="host=localhost:5672;virtualHost=/;username=guest;password=guest;timeout=120" />
4 </appSettings>
5 </configuration>
```

- Adjust the RabbitMQ connection information with the right information.

`value="host=<node1>,<node2>,<node3>;virtualHost=<vhost>;username=<user>;password=<password>;timeout=120"`

Where, each parameters separated semicolons (";"):

- <nodeX>** is the name of each node, usually it is the hostname. *(Separated by commas ",")*
- <vhost>** is the vHost's id created in previous step.
- <user>** is the RabbitMQ user's id created in previous step.
- <password>** is the RabbitMQ user's password created in previous step.

Example in context:

```
<configuration>
<appSettings>
<add key="RabbitMQ.ConnectionString" value="host=OfeliaHA-SVR1:5672,OfeliaHA-SVR2:5672,OfeliaHA-SVR3:5672;virtualHost=Ofelia;username=OfeliaSystemUser;password=OfeliaSystemPassword;timeout=120" />
</appSettings>
</configuration>
```



Additional notes:

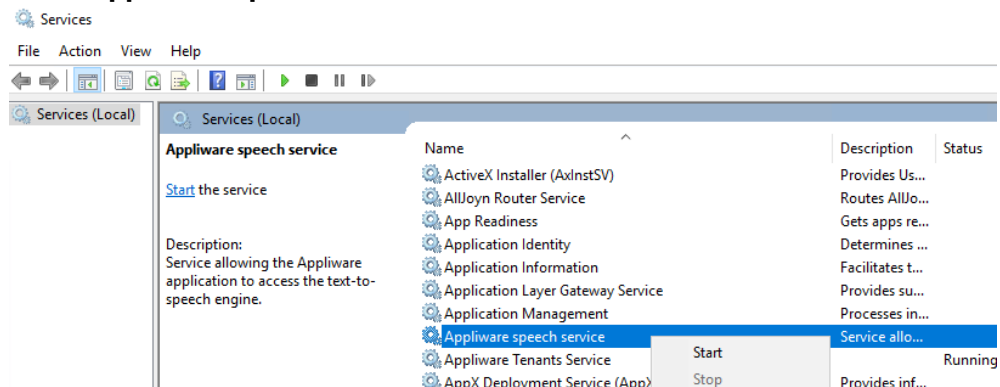
- If the default vHost of rabbitMQ is used ("/"), it is possible to:
 - Use "virtualHost=/"

`value="host=<node1>,<node2>,<node3>;virtualHost=/;username=<user>;password=<password>"`
 - Deleting the "virtualHost" part of the connection string:

`value="host=<node1>,<node2>,<node3>;username=<user>;password=<password>"`
- The RabbitMQ connection string should not end with a semicolon (";")
For this connection string, the semicolons (";") are only used as separators between the different types of parameters, but not as an end-of-string marker.

- Save** changes.

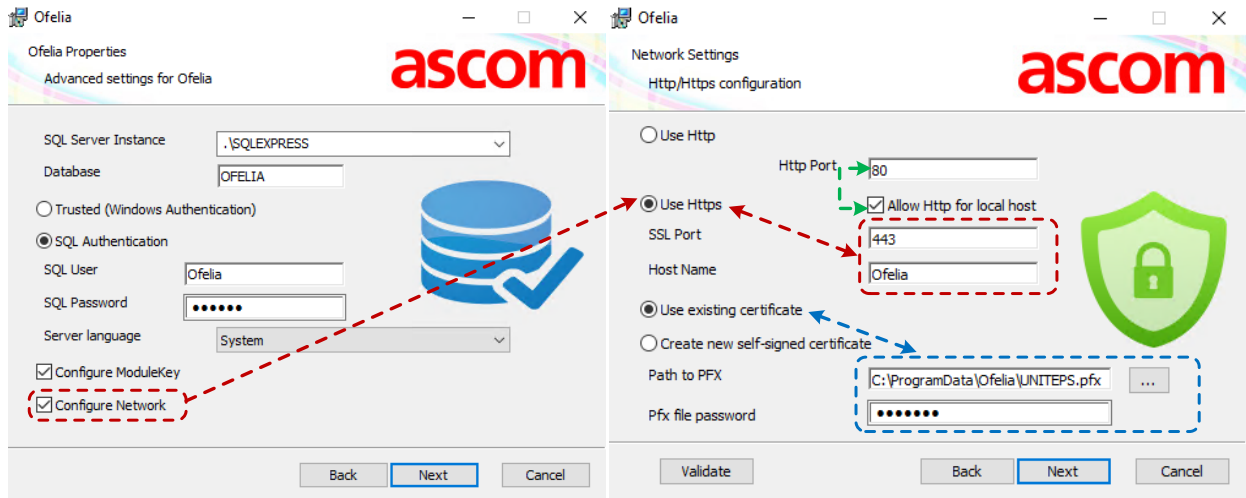
- Restart <Appliware speech service>** service.



11.4. Adjust the Ofelia's http/https connection settings:

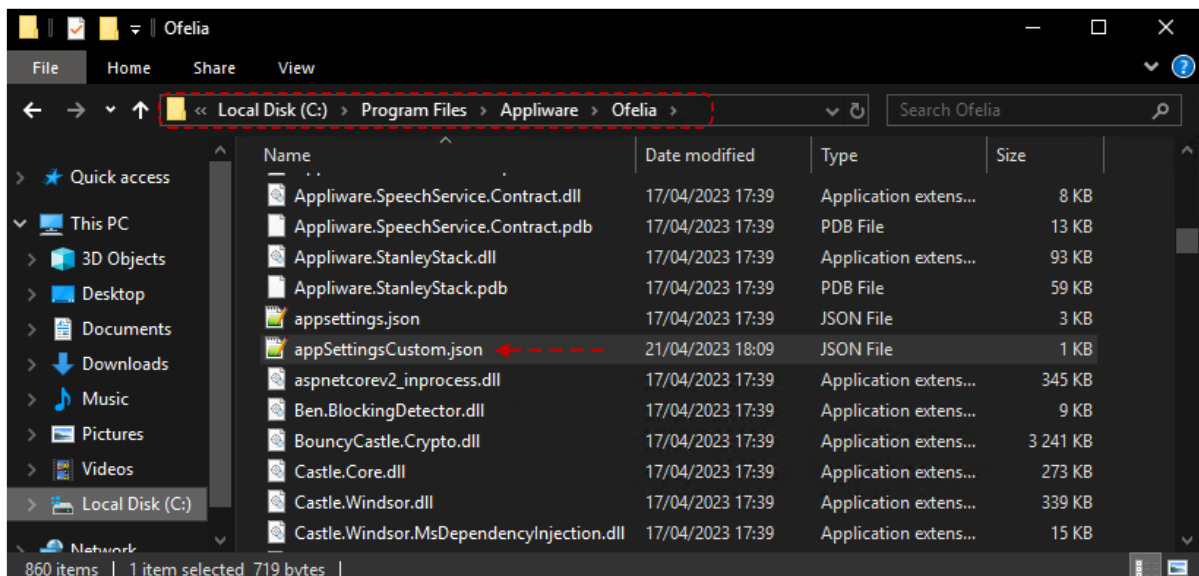
To be able to activate the Https connection you must have a certificate signed by a trusted authority.

The choice of network access type is usually configured in the Ofelia installation program.



However, it is also possible to adjust these parameters by editing the Ofelia configuration file.

- Stop the Ofelia application completely.
- With Windows explorer, **open** the Ofelia installation directory, by default:
C:\Program Files\Appliware\Ofelia
- **Find** the file **<appSettingsCustom.json>**



- **Edit** this file with a text editor and locate the part about “**Kestrel**” settings.



```
{
  "Kestrel": {
    "Endpoints": {
      "Http": {
        "Url": "http://*:80/"
      }
    }
  },
  "AppSettings": {
    "ConnectionStrings": {
      "Ofelia": "Data Source=OFELIAHA-SQL;Initial Cat
      "FileLog": "tcp(reconnect=true, reconnect.inter
      "RabbitMQ": "host=localhost;virtualHost=Ofelia;
```

- Adjust the “**Kestrel**” parameters according to required configuration for https connection.

Using only HTTPS (without http for local connection):



```
"Kestrel": {
  "EndPoints": {
    "Https": {
      "Url": "https://OFELIAHA.mydomain.lo:8080",
      "Certificate": {
        "Path": "C:\\\\OFELIAHA.pfx",
        "Password": "SSLCertPassword"
      }
    }
  }
},
```

<Url> Parameter defines the connection URL to be used to connect to Ofelia and allows you to change the listening port (here the 8080)

"Url": "https://OFELIAHA.mydomain.lo:8080" : users must use this exact URL (in this format) to establish a connection with Ofelia.

<Path> Defines the path of your PFX file, making sure to double the "\\ "

<Password> Defines the password for the certificate.

Using HTTPS for connection, and allow http connection only for local):

```
"Kestrel": {
  "Endpoints": {
    "Http": {
      "Url": "http://127.0.0.1:80/"
    },
    "Https": {
      "Url": "https://OFELIAHA.mydomain.lo:8080",
      "Certificate": {
        "Path": "C:\\\\OFELIAHA.pfx",
        "Password": "SSLCertPassword"
      }
    }
  }
},
},
```

In this example, two types of connection are allowed:

- http from the Ofelia server itself.
- in https from any other machine.

Under **<http>** section:

<Url> Parameter defines the connection URL to be used to connect to Ofelia and allows you to change the listening port (here the 80)

"Url": "https://127.0.0.1:80" : users must use URL with 127.0.0.1 as address to establish a connection with Ofelia through port 80.

"Url": "https://localhost:80" : users must use localhost URL to establish a connection with Ofelia through port 80.

"Url": "https://*:80" : users can use all URL available to establish a connection with Ofelia through port 80.

Under **<https>** section:

<Url> Parameter defines the connection URL to be used to connect to Ofelia and allows you to change the listening port (here the 8080)

"Url": "https://OFELIAHA.mydomain.lo:8080" : users must use this exact URL (in this format) to establish a connection with Ofelia.

<Path> Defines the path of your pfx file, making sure to double the backslash ("\\") between each folder name constituting the file location.

<Password> Defines the password for the certificate.



Note: You can check that the content of your JSON file is correct (no missing brackets, commas, etc.) using online verification tools, such as <https://jsonlint.com/>

```
1 {
2   "Kestrel": {
3     "Endpoints": {
4       "Http": {
5         "Url": "http://127.0.0.1:80/"
6       },
7       "Https": {
8         "Url": "https://OFELIAHA.mydomain.lo:8080",
9         "Certificate": {
10          "Path": "C:\\OFELIAHA.pfx",
11          "Password": "SSLCertPassword"
12        }
13      }
14    },
15  },
16  "AppSettings": {
17    "ConnectionStrings": {
18      "Ofelia": "Data Source=OFELIAHA-SQL;Initial Catalog=Ofelia;Integrated Security=true;",
19      "FileLog": "tcp(reconnect=true, reconnect.interval=5s), file(filename=C:\\ProgramData\\Appliware\\Ouranos\\log.sil,level=
20      "RabbitMQ": "host=localhost;virtualHost=Ofelia;timeout=120"
21    },
22    "ModuleKey": {},
23    "NoOuranosCom": false,
24    "MultiTenants": true,
25    "MultiTenantsDisableAutoRefresh": false,
26    "DisableAutoBackup": false,
27    "CreateRabbitMQVirtualHost": true
28  }
29 }
```

Validate JSON

Clear

Support JSONLint for \$2/Month

Results

Valid JSON

- **Save** changes.
- **Restart** Ofelia application.

12. Appendix B - Load Balancing:

For multi-cluster installations, a load balancer set to support WEB Socket Protocol is required for connections to Ofelia.

12.1. Health Check Monitor

This section describes how a health check works and why it is required. The load balancer must monitor the nodes in order to distribute traffic properly when needed. For example, if the load balancer notices that an application on a node does not respond to a request, the load balancer must distribute the request to an operational application running on another node. The load balancer must support monitoring of the applications over TCP.

In Ofelia there is an HTTP monitoring service that can be used by the load balancer to monitor the status of the nodes, and to for example turn off a node during the upgrade.

The service is a REST API that is called with <http://<IpAddress>/health> or <http://servername.domain/health>

The top part of the image shows a web browser window with the address bar set to `192.168.0.66/health`. The page displays a JSON response from the health check API. A red arrow points to the `"status": "Healthy"` field in the root object.

The bottom part of the image shows a JSON viewer displaying the same JSON data. A red dashed arrow points from the `"status": "Healthy"` field in the JSON viewer to the text: "Status global of the current node being monitored. This is what needs to be monitored by the LB". A blue dashed arrow points from the `"entries"` array in the JSON viewer to the text: "Status of each component of this node".

```

{
  "status": "Healthy",
  "totalDuration": "00:00:00.0149494",
  "entries": {
    "sqlserver": {
      "data": {},
      "duration": "00:00:00.0052863",
      "status": "Healthy",
      "tags": []
    },
    "RabbitMq [OfeliaHA-SVR3]": {
      "data": {},
      "duration": "00:00:00.0109262",
      "status": "Healthy",
      "tags": []
    },
    "RabbitMq [OfeliaHA-SVR2]": {
      "data": {},
      "duration": "00:00:00.0096439",
      "status": "Healthy",
      "tags": []
    },
    "RabbitMq [OfeliaHA-SVR1]": {
      "data": {},
      "duration": "00:00:00.0147949",
      "status": "Healthy",
      "tags": []
    },
    "RabbitMq": {
      "data": {},
      "duration": "00:00:00.0000054",
      "status": "Healthy",
      "tags": []
    }
  }
}
  
```

An example of a view with Notepad++, after processing with the "JsonTools" addon.

Global status description of the current node being monitored:

- **Healthy** = Everything is working properly
- **Degraded** = One of the (non-critical) components are no longer functional, but the node continues to operate
- **Unhealthy** = The node is no longer able to receive requests

12.2. Load Balancing Methods

A load balancing method controls the distribution of incoming requests to your cluster of servers. Examples of methods are listed below:

- **Round Robin** – distributes requests across a group of servers. The request is forwarded to each server in turn in the “server list”. When the load balancer has reached the end of the server list, it starts from the top of the list and repeats.
- **Weighted Round Robin** – the same principle as Round Robin, but the number of requests that each server receives over time is proportionate to the capacity of each server. Example: Server 1 has more capacity than Server 2 in a cluster, therefore, Server 1 should handle more requests than Server 2.
- **Least Connections (recommended control method)** – distributes requests to the server with the least number of connections. Typically used when servers in the cluster have the same capacity.
- **Weighted Least Connection** – the same principle as Least Connection, except that servers are selected based on capacity, not just availability. Example: Server 1 has more capacity than Server 2 in a cluster, therefore, Server 1 should handle more requests than Server 2.

Health check interval of down services

It is important that the load balancer monitors services that are considered as “down” with an interval that does not exceed 10 seconds. This is to provide a fail-over as fast as possible if the active node goes down and a passive node starts services to take over.

Persistence

Persistence is a process in which a load balancer creates an affinity between a client and a specific server in the cluster for the duration of a session. The load balancer must be configured to distribute requests from a client to the same server in the cluster during a session. The persistence should be configured for all clients that connect to the Ofelia Server via the load balance.

Example:

The load balancer receives an initial request from a specific client and distributes the request to Server 1 in the cluster, according to the load balancing method used. If the load balancer, for some reason, distributes requests from this client to Server 2 in the cluster, the heartbeat and synchronization might not work properly between the client and Server 1.

To solve this, the load balancer must retain the connection to Server 1 for a short period of time (that is, 2 minutes), which is done by configuring “persistence” in the load balancer.

For information about persistence configuration, refer to the vendor’s documentation.

Keep idle connections during a timeout.

If no data is sent to/from the load balancer, it must be configured to keep the connection towards the nodes/clients for a specified period. If no data has been sent or received by the time that the idle timeout period elapses, the load balancer closes the connection.

12.3. Load Balancing of Services

In general, for WEB navigation and OfeliaSmart to work properly, the Load Balancer must be set to support the **WebSocket protocol**.

Then, depending on the type of connection, it may or may not be necessary to use the Load Balancer for each of the drivers available on Ofelia.

Examples:

- teleCARE IP (*SSH connection*) **does not require** Load Balancer. Each driver instance (*one per node*) must be directly configured to connect to the tclP central unit.
- For Unite (*OAP*) **it may be possible to use** certain Load Balancer functions (*like reverse proxy*) so that each driver instance (*one per node*) connects to the Load Balancer, the Load Balancer connects to Unite and redirects the responses to be returned to Ofelia to all nodes simultaneously (*Ofelia will organize itself internally so that the same information received on several nodes is processed only once*).

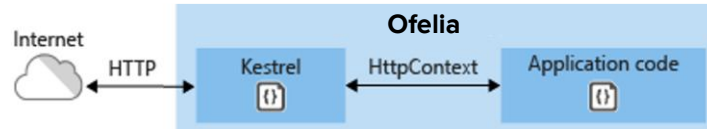
This remains to be defined for each driver on a case-by-case basis. This part of the document is still being drafted and may be completed in the future.

13. Appendix C - Using Ofelia through a reverse proxy:

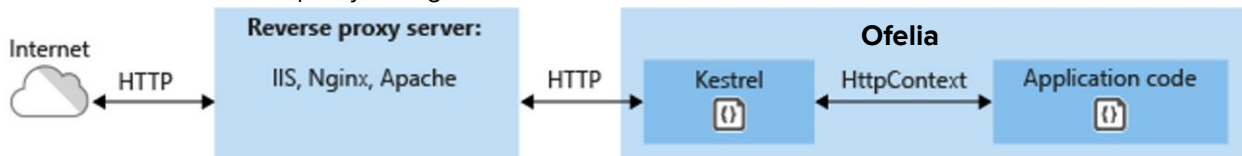
Ofelia does not use the web server provided by Microsoft ("Microsoft IIS"). Ofelia uses the WEB server of the "Kestrel" component.

Kestrel can be used by itself or with a reverse proxy server. A reverse proxy server receives HTTP requests from the network and forwards them to Kestrel.

Kestrel used as an edge (Internet-facing) web server:



Kestrel used in a reverse proxy configuration:



Either configuration, with or without a reverse proxy server, is a supported hosting configuration.

Even if a reverse proxy server isn't required, using a reverse proxy server might be a good choice.

➔ **The reverse proxy must be set to support the WebSocket protocol.**

A reverse proxy:

- Can limit the exposed public surface area of the apps that it hosts.
- Provide an additional layer of configuration and defense.
- Might integrate better with existing infrastructure.
- Simplify load balancing and secure communication (HTTPS) configuration.
- Only the reverse proxy server requires an X.509 certificate, and that server can communicate with the app's servers on the internal network using plain HTTP.



Note: Ofelia does not integrate reverse proxy. It will be necessary to rely on external tools available on the customer's infrastructure.

Implementation is therefore not covered in this document.

14. Appendix D - Flow Matrix:

14.1. General ports used:

Essential ports (for Ofelia and its prerequisites)				
From	To	Port	Protocol	Comments
Ofelia Server	DNS server	53	UDP & TCP DNS	
Ofelia Server	NTP sever	123	UDP NTP	
Ofelia Server	SQL server	1433	TCP	SQL Server default port
Ofelia Server	RabbitMQ nodes	5672 (by default)	TCP	RabbitMQ's AMQP protocol
Ofelia Server	RabbitMQ nodes	15672	TCP	RabbitMQ's WEB management console
RabbitMQ nodes	RabbitMQ nodes	4369	TCP	EPDM (Erlang), a peer discovery service used by RabbitMQ nodes and CLI tools for RabbitMQ
RabbitMQ nodes	RabbitMQ nodes	6000 to 6500	TCP	RabbitMQ, used for stream replication,
RabbitMQ nodes	RabbitMQ nodes	25672	TCP	RabbitMQ, used for communication between nodes and CLI tools (Erlang distribution server port)
RabbitMQ nodes	RabbitMQ nodes	35672-35682	TCP	RabbitMQ, used by CLI tools (Erlang distribution client ports)

License validation (for licenses based on the Elise3 module key)				
From	To	Port	Protocol	Comments
Ofelia Server	Ascom Elise3 module	80	TCP HTTP	Used when the Ofelia license is generated in reference to an Elise3 module key
Ofelia Server	Ascom Elise3 module	443	TCP HTTPS	Used when the Ofelia license is generated in reference to an Elise3 module key

Ofelia Browser (Windows App) // Web Browser				
From	To	Port	Protocol	Comments
Ofelia Browser	Ofelia Server	80 (By default)	TCP HTTP	Connection between a client (Web Browser App or Web browser) to Ofelia Server over HTTP by default. (Same port used for Web Access to Ofelia, that can be customized)
Ofelia Browser	Ofelia Server	443 (By default)	TCP HTTPS	Web connection over HTTPS by default. (Same port used for Web Access, that can be customized)

14.2. Ports used by Ofelia's drivers:

Driver Aeroscout - Stanley Healthcare (http-POST XML - for Technical Event Generators)				
From	To	Port	Protocol	Comments
Stanley server	Ofelia Server	80	TCP HTTP	Connection between Stanley to Ofelia Server over HTTP by default. (Same port used for Web Access to Ofelia, that can be customized)
Stanley server	Ofelia Server	443	TCP HTTPS	Connection over HTTPS by default. (Same port used for Web Access, that can be customized)

Driver ASCII				
From	To	Port	Protocol	Comments
ASCII sender	Ofelia Server	38000*	TCP	When 'Listen socket' is used, by default (can be customized) *If there are several ASCII drivers, each driver must have its own port. (Communication over IP or using Serial-IP converter)
ASCII sender	Ofelia Server	38000*	UDP	When 'Socket UDP' is used, by default (can be customized) *If there are several ASCII drivers, each driver must have its own port. (Communication over IP or using Serial-IP converter)
Ofelia Server	ASCII sender	38000	TCP	When 'Socket connection' is used, by default (can be customized) (Communication over IP or using Serial-IP converter)

Driver Ascom (ASAP)				
From	To	Port	Protocol	Comments
Ofelia Server	Unite module <i>Messaging Gateway</i>	3217	UDP UNITE	
Unite module <i>Messaging Gateway</i>	Ofelia Server	3217	UDP UNITE	
teleCARE IP Control Equipment <i>NIRC3/4 / NIWC / vNISM / NISM2 /,,,</i>	Ofelia Server	3217	UDP UNITE	

Driver AscomOAP & AscomOAP One Way				
From	To	Port	Protocol	Comments
Ofelia Server	Unite module (OAP server)	1322	TCP	Ascom OAP v4 Protocol
Unite module (OAP server)	Ofelia Server	1323* (By default)	TCP	Ascom OAP v4 Protocol (not used for 'AscomOAP One Way' driver) *If there are several 'OAP v4' drivers, each driver must have its own listen port.
Ofelia Server	Unite module (OAP server)	1321	TCP	Ascom OAP v3 Protocol
Unite module (OAP server)	Ofelia Server	1321	TCP	Ascom OAP v3 Protocol (not used for 'AscomOAP One Way' driver)

Driver BACnet				
From	To	Port	Protocol	Comments
Ofelia Server	BACnet server	47808	TCP	By default (can be customized)

Driver DigitalVirgo				
From	To	Port	Protocol	Comments
Ofelia Server	DigitalVirgo (online)	443	TCP HTTPS	
DigitalVirgo (online)	Ofelia Server	80	TCP HTTP	Connection between DigitalVirgo to Ofelia Server over HTTP by default. (Same port used for Web Access to Ofelia, that can be customized)
DigitalVirgo (online)	Ofelia Server	433	TCP HTTPS	Connection over HTTPS by default. (Same port used for Web Access, that can be customized)

Driver eMail				
From	To	Port	Protocol	Comments
Ofelia Server	Mail server	25	SMTP	By default (can be customized)
Ofelia Server	Mail server	587	SMTPS	Often used to encrypt SMTP messages using STARTTLS - By default (can be customized)
Ofelia Server	Mail server	465	SMTPS	Often used to encrypt SMTP messages using SSL - By default (can be customized)

Driver ESPA <i>ESPA 4.4.4</i>				
From	To	Port	Protocol	Comments
ESPA sender	Ofelia Server	4001*	TCP	When 'Listen socket' is used, by default (can be customized) *If there are several ESPA drivers, each driver must have its own port. (Communication over IP or using Serial-IP converter)
Ofelia Server	ESPA sender	4001	TCP	When 'Socket connection' is used, by default (can be customized) (Communication over IP or using Serial-IP converter)

Driver HighSMS				
From	To	Port	Protocol	Comments
Ofelia Server	HighSMS (online)	443	TCP HTTPS	
HighSMS (online)	Ofelia Server	80	TCP HTTP	Connection between HighSMS to Ofelia Server over HTTP by default. (Same port used for Web Access to Ofelia, that can be customized)
HighSMS (online)	Ofelia Server	433	TCP HTTPS	Connection over HTTPS by default. (Same port used for Web Access, that can be customized)

Driver Http				
From	To	Port	Protocol	Comments
Http third-party App,	Ofelia Server	80	TCP HTTP	Connection between a third-party application to Ofelia Server over HTTP by default. (Same port used for Web Access to Ofelia, that can be customized)
Http third-party App,	Ofelia Server	443	TCP HTTPS	Connection over HTTPS by default. (Same port used for Web Access, that can be customized)

Driver MiniFinder				
From	To	Port	Protocol	Comments
MiniFinder	Ofelia Server	35062	TCP	

Driver Modbus (Master) - TCP				
From	To	Port	Protocol	Comments
Ofelia Server	Modbus server	502 (By default)	TCP	Socket connection - By default (can be customized)
<i>Ofelia Server</i>	<i>Modbus server</i>	<i>80</i>	<i>TCP</i>	<i>For certain modules that are configured via a WEB interface (e.g. WAGO)</i>

Driver Modbus (Slave) - TCP				
From	To	Port	Protocol	Comments
Modbus Server	Ofelia Server	502* (By default)	TCP	Listen socket - By default (can be customized) *If there are several Modbus (Slave) drivers, each driver must have its own port.

Driver MultiTech (SMS Gateway)				
From	To	Port	Protocol	Comments
Ofelia Server	RabbitMQ nodes	443	TCP HTTPS	By default.

Driver Ofelia Messenger (Windows App)				
From	To	Port	Protocol	Comments
Ofelia Messenger	Ofelia Server	34892	TCP	By default (can be customized)

Driver OfeliaSmart (Android App)				
From	To	Port	Protocol	Comments
OfeliaSmart	Ofelia Server	80	TCP HTTP	Connection between Ofelia Smart to Ofelia Server over HTTP by default. (Same port used for Web Access to Ofelia, that can be customized)
OfeliaSmart	Ofelia Server	443	TCP HTTPS	Connection over HTTPS by default. (Same port used for Web Access, that can be customized)
OfeliaSmart	Syslog server	514	TCP	By default (can be customized)
OfeliaSmart	SIP Proxy	5065	UDP SIP	By default (can be customized) - But do not use 5060 on Myco3/4

Driver OPC-UA				
From	To	Port	Protocol	Comments
Ofelia Server	OCP-UA Server	4840	TCP	By default (can be customized) - Depending on the configuration set up on the OPC-UA third-party server.

Driver Opera				
From	To	Port	Protocol	Comments
Ofelia Server	Opera	1412	TCP	By default (can be customized) - Depending on the configuration set up on the Opera third-party server.

Driver Ping				
From	To	Port	Protocol	Comments
Ofelia Server	Device to be pinged		ICMP	<i>No port used for ping. Allow the protocol.</i>
Device to be pinged	Ofelia Server		ICMP	<i>No port used for ping. Allow the protocol.</i>

Driver PopUp (Windows App)				
From	To	Port	Protocol	Comments
PopUp	Ofelia Server	38420	TCP	

Driver SIA				
From	To	Port	Protocol	Comments
Ofelia Server	SIA	9401	TCP/UDP	

Driver SIP				
From	To	Port	Protocol	Comments
Ofelia Server	SIP Proxy (IPBX)	5060 (By default)	UDP SIP	
SIP Proxy (IPBX)	Ofelia Server	5060 (By default)	UDP SIP	
Ofelia Server	Medias (e.g. IP phones/Gateway)	4000 to 4512	UDP RTP	No UDP session timeout.
Medias (e.g. IP phones/Gateway)	Ofelia Server	4000 to 4512	UDP RTP	No UDP session timeout.

Driver SmartSense				
From	To	Port	Protocol	Comments
SmartSense server	Ofelia Server (& All RabbitMQ's nodes)	5672	TCP	RabbitMQ's AMQP protocol - On multi-nodes configuration, SmartSense server should be able to communicate with all RabbitMQ nodes

Driver SMSFactor				
From	To	Port	Protocol	Comments
Ofelia Server	SMSFactor (online)	443	TCP HTTPS	
SMSFactor (online)	Ofelia Server	80	TCP HTTP	Connection between SMSFactor to Ofelia Server over HTTP by default. (Same port used for Web Access to Ofelia, that can be customized)
SMSFactor (online)	Ofelia Server	433	TCP HTTPS	Connection over HTTPS by default. (Same port used for Web Access, that can be customized)

Driver SMTP Server				
From	To	Port	Protocol	Comments
Third-party sender	Ofelia Server	25	SMTP	By default (can be customized)
Third-party sender	Ofelia Server	587	SMTPS	When STARTTLS encryption is required - By default (can be customized)
Third-party sender	Ofelia Server	565	SMTPS	When SSL encryption is required - By default (can be customized)

Driver Stanley Healthcare (http-POST JSON - for Personal alarm)				
From	To	Port	Protocol	Comments
Stanley server	Ofelia Server	80	TCP HTTP	Connection between Stanley to Ofelia Server over HTTP by default. (Same port used for Web Access to Ofelia, that can be customized)
Stanley server	Ofelia Server	443	TCP HTTPS	Connection over HTTPS by default. (Same port used for Web Access, that can be customized)

teleCARE IP (NISM/vNISM over SSH)				
From	To	Port	Protocol	Comments
Ofelia Server	vNISM/NISM2	2222	TCP SSH	
Ofelia Server	MARi	443	TCP HTTPS	When the MARi API must be used

teleCARE M (ISC3 over IP)				
From	To	Port	Protocol	Comments
ascom ISC3 module	Ofelia Server	15001	UDP	By default (can be customized) - Depending on the configuration set up on the ascom ISC3 module

14.3.Ports used by Ofelia's 'External Applications':

External App - LDAPDIRECTORY (For SSO)				
From	To	Port	Protocol	Comments
Ofelia Server	Central directory (LDAP)	389 (or 3268 if MS-AD)	TCP	Communication with external database used for SSO over LDAP.
Ofelia Server	Central directory (LDAPS)	636 (or 3269 if MS-AD)	TCP	Communication with external database used for SSO over LDAPS (LDAP over SSL).
Ofelia Server	Kerberos	88	UDP & TCP	If Kerberos is used

External App - MARI (For teleCARE IP)				
From	To	Port	Protocol	Comments
Ofelia Server	vNISM (or MARI Server)	443	TCP HTTPS	

External App - OIDC (For SSO)				
From	To	Port	Protocol	Comments
Ofelia Server	OIDC Provider	80*	HTTP	By default (can be customized) *Depending on the configuration set up on the OIDC provider
Ofelia Server	OIDC Provider	443*	HTTPS	By default (can be customized) – *Depending on the configuration set up on the OIDC provider
OIDC Provider	Ofelia Server	80	HTTP	Web connection to Ofelia over HTTP by default. (Same port used for Web Access, that can be customized)
OIDC Provider	Ofelia Server	443	HTTPS	Web connection to Ofelia over HTTPS by default. (Same port used for Web Access, that can be customized)

External App - SmartSense				
From	To	Port	Protocol	Comments
Ofelia Server	SmartSense Server	443	TCP HTTPS	SmartSense Web management port (for API)

External App – Stanley Healthcare (For Wi-Fi Hyperlocalisation via MobileView®)				
From	To	Port	Protocol	Comments
Ofelia Server	Stanley Server	80	TCP HTTP	

External App - teleCARE IP				
From	To	Port	Protocol	Comments
Ofelia Server	vNISM or NISM2	5432	TCP	

15. Related Documents

Document	Document Number
Global documentation	1.2
Data Sheet	<i>The last available</i>

16. Document History:

Version	Date	Description
R0	29 Jun. 2023	Initial release.
R1A	03 Jul. 2023	Update document.
R1B	27 Jul. 2023	Update document.
R1C	30 May. 2024	Update document for Ofelia 2.2 rev 1 • RabbitMQ information to the installer screens has been added. • Updated considerations for multi-node installations (clustered - HA) • Reorganization of the chapter on manual modification of configuration files. • Added a chapter on the requirements for text-to-speech functionality. • Flow Matrix has been added.