

# Troubleshooting Agent Installation and Configuration Errors

If you have issues installing the Agent and then configuring it to start and run, first try troubleshooting issues by doing the following:

## Use troubleshooting tools

---

### Create Report

To gather the common files such as configurations and logs for troubleshooting, you will run the **createReport.bat** file. Then, run the following command line in the root **Perspectium** directory:

```
# Windows Systems
bin\createReport.bat
Report successfully generated as file: ReplicatorReport.name.date.zip

# OS X and Linux Systems
bash bin/createReport
```

The tool will output a line telling you if it ran successfully and the name of the generated **.zip** file. The file will contain the name of your agent and the current date.

The files included are:

- /conf/agent.xml
- /conf/config.xml
- /conf/log4j2.xml
- /conf/wrapper.conf
- /logs/\* (all the logs)
- /bin/(instance schemas)

**NOTE:** The tool will package the files as they are, so if you have unencrypted sensitive data within them then you should edit the files appropriately before creating the report.

---

### Validate Configuration

The agent.xml **Validate Configuration** tool tests your access to the default and configured message bus connections. It will display other diagnostic information for your current agent name, version, and tasks. The tool verifies if the Agent encryption key matches with the key in ServiceNow and the shared queue with the key is associated to a bulk or dynamic share. The API to perform the check is in DataSync for ServiceNow 6.1.0+. If a lower version is used, the check will be skipped.

For Windows, go to the bin folder of the agent's installed directory and double click on the **validateConfiguration.bat** file. This will automatically run the configuration validation test.

The tool can be run at the command line of the **root Perspectium directory**, not bin, like so:

```
# Windows Systems
bin\validateConfiguration.bat

# OS X and Linux Systems
bash bin/validateConfiguration
```

The following example is a successful validation:

```
Configuration Validation Report: Fri Feb 02 09:13:40 PST 2021
```

Agent Information:

```
Agent version: Iodine_7.0.0
Agent name: 'example_agent'
```

Configured Tasks:

```
Task: example_agent_subscribe Type: subscribe
      example_agent_subscribe instances: 1
```

```
Attempting to connect to the Message Broker Service at: amqps://example.perspectium.net as
user: myUser - SUCCESS
Attempting to connect to the database: psp_repl at host: localhost port: 3306 as user:
myUser - SUCCESS
Attempting to fetch schema from: https://example.service-now.com- SUCCESS
Attempting to validate ServiceNow encryption key with Agent decryption key - SUCCESS
```

```
Validation Completed, results: SUCCESS
```

```
Report has been saved in file: ConfigurationValidationReport.log
```

**NOTE:** With **Iodine 7.0.8+**, the Agent will validate its configurations on start up (results will be shown and saved in the Agent's **perspectium.log** and not the separate **ConfigurationValidationReport.log** as the Validate Configuration tool does).

## Enable debug logging

The initial logging level of the Agent is set at its lowest level of **INFO** so as to not generate extraneous logging and affect performance. However in case of errors and to aid in troubleshooting, you can enable a [higher level of logging](#) such as **DEBUG**. Note that changing to a higher level should only be done during troubleshooting and changed back to **INFO** for every day use because of the performance cost.

## Agent startup issues

### Start the Agent without the wrapper

Starting the Agent without the wrapper that runs it as a service allows you to see if there are issues with running the Agent itself including any issues with the Agent's dependencies such as Java. To run without the wrapper:

- In a terminal window or command prompt, change to the directory where the Agent is installed (e.g. `cd /usr/local/Perspectium_Replicator_Agent`)
- Run the following command:

```
java -Dlog4j.configurationFile=file:conf/log4j2.xml -classpath ".:${CLASSPATH}:bin:jars/*:lib/*" com.perspectium.replicator.Replicator
```

## Verify Java on the machine

For issues with the Agent running as a service, verify Java is setup properly on the machine:

### Verify Java path is defined properly (Windows-specific)

For Windows, first check the Java path (where Java is installed and is defined to run from) is in the [Windows PATH environment variable](#).

Next check the **wrapper.conf** file located in the Agent's **conf** directory has the **set.JAVA\_HOME=<Java\_Path>** parameter defined.

### Verify Java is accessible

If the Agent is having issues starting up, check the **java** system command is working. Normally when **Java** is installed on a machine, the **java** system command is available to executed from any system command prompt in any directory. If you check for Java on your machine by executing the **java -version** and it doesn't work, then the java system command may not be set up properly.

First, verify the user you are currently logged in on the machine has access to the Java install (it may have been installed by another user and your user wasn't given access).

If the user does have access to Java, you may need to explicitly point to where the **java** executable is located for the Agent to run it. The executable is generally located in in the **bin** folder where Java is installed i.e.

/usr/local/java/jre1.8.0\_181/bin/java (Linux)

C:\Program Files\Java\jre1.8.0\_181\bin\java.exe (Windows)

Once you have the path of the Java executable, open up the **wrapper.conf** configuration file located in the Agent's **conf** directory and update the **wrapper.java.command** configuration near the top of the file to point to the path.

That is, change it from:

```
wrapper.java.command=java
```

To this (using the example above):

Linux

```
wrapper.java.command=/usr/local/java/jre1.8.0_181/bin/java
```

Windows

**NOTE:** Use **\\** for each folder in order for the path to be read correctly

```
wrapper.java.command=C:\\Program Files\\Java\\jre1.8.0_181\\bin\\java.exe
```

Once you've done the initial steps for troubleshooting issues, review the the logs found in the Agent's **logs** folder to see the issues. Some of the more common issues include:

## Outbound Network Access

You should ensure that your firewall and or proxy allow outbound connectivity to the Perspectium Integration Mesh and your ServiceNow Instance.

Note: Running the configuration validation tool will perform connectivity tests for all elements defined within the agent.xml configuration file and will tell you if the test succeeded or failed. The following is recommended if additional trouble shooting is required.

Depending on the protocol you plan on using you should test the corresponding DNS resolved name and port number. If you plan on operating through a proxy then you should test it as well.

The connection to the destination host and port can be tested by calling *telnet* from the command line as necessary with the ports:

| HTTPS | AMQP | AMQPS | HTTP |
|-------|------|-------|------|
| 443   | 5672 | 5671  | 80   |

If you're running the Agent on a Windows server and need assistance installed the telnet client please reference this link: <https://social.technet.microsoft.com/wiki/contents/articles/22715-how-to-enable-telnet-in-windows-server-2012.aspx>

If you're running the Agent on a Linux distribution that supports yum then you should be able to install the client by issuing the following command which requires sudo privileges.

```
sudo yum install telnet
```

If your system does not leverage yum then ask your system administrator for assistance.

The following reflects a successful test to the host <your\_url>.perspectium.net. The telnet command specifies the destination host's fully qualified domain name followed by the target port which in this case is 5672. The message starting with 'Connected to' states that the connection has been established which is the desired outcome.

```
# Good Connections (the IP address will vary per region)
telnet <your_url>.perspectium.net 5672
Trying 52.8.132.178 ...
Connected to <your_url>.perspectium.net.
Escape character is '^]'.
```

The next example is also successful at establishing a connection to the host <your\_url>.perspectium.net which accepts connections directed to port 443 or HTTPS.

```
telnet <your_url>.perspectium.net 443
Trying 52.8.203.198 ...
Connected to <your_url>.perspectium.net.
Escape character is '^]'.
```

The next example of a successful test is performed to ensure that the agent can connect to your ServiceNow instance which accepts HTTPS connections via port 443.

```
telnet <your_instance>.service-now.com 443
Trying 103.23.66.118 ...
Connected to <your_instance>.service-now.com.
Escape character is '^]'.
```

Now let's take a look at what failed connection attempts look like.

In the following example we attempt to connect to the host my\_domaine.perspectium.net using port 443 or HTTPS. The error message could be due to a typo in the spelling of the fully qualified domain name. Perhaps the name you were assigned is my\_domain but what was actually typed during the test or perhaps entered into the agent.xml configuration file was my\_domaine.perspectium.net (Note the extraneous 'e' at the end of domain):

```
# Failed Connections (you may get a failed response or no response at all)
telnet my_domaine.perspectium.net 443
bad_domain.perspectium.net: nodename nor servname provided, or not known
```

Here are some additional examples or error responses that can happen when connection attempts fail:

```
# Failed Connections (you may get a failed response or no response at all)
telnet blocked_domain.perspectiun.net 443
telnet: connect to address XX.XX.XXX.XX: Connection refused
telnet: Unable to connect to remote host
```

It's also a good idea to see if you have general internet connectivity. For example, you should be able to connect to google via telnet:

```
telnet www.google.com 443
```

All my tests fail, what should I do? If all of your tests fail then it's likely that you have one or more of a few standard issues.

- Your company has a security policy in place such that outbound connections to certain destinations such as Perspectium must be configured. Discuss your issue with your network support team.
- Your company leverages an outbound proxy. An outbound proxy server is a server (a computer system or an application) that acts as an intermediary for requests from clients seeking resources from other servers, typically those in the public internet. Please contact your proxy support team, or possibly network support for assistance.

---

## Failed Initial Connection

If you see the following error with **Connection refused**:

```
ERROR AMQP - new connection attempt failed: Connection refused
ERROR SubscriberTask - Subscriber Error: Failed to get connection to URI: amqps://<your_url>.perspectium.net
VHost: / User: your_instance
ERROR SubscriberTask - Can't report status - MessageBus is not set or not open
```

This is a result of an incorrect protocol/DNS relation. This is solved by correctly configuring the **<message\_connection>** tag which depends on the assigned DNS given to you by Perspectium Support. Your connection will likely be either:

- amqps://<your\_url>.perspectium.net
- https://<your\_url>.perspectium.net

---

## Incorrect Login Credentials

### Incorrect Subscriber - Perspectium Credentials

```
ERROR SubscriberTask - Subscriber Error: Failed to authenticate to URI: https://<your_url>.perspectium.net
VHost: / User: your_instance
ERROR SubscriberTask - Can't report status - MessageBus is not set or not open!
```

If your subscriber task has incorrect login credentials you cannot authenticate to our servers and you may see these error messages. Double check that your credentials are correct.

### Incorrect Subscriber - ServiceNow Credentials

```
ERROR ServiceNowSchema - Error: Send Error: Unexpected response status: 401
WARN SchemaDocument - com.perspectium.replicator.sql.subscriber.SchemaAccessException: Send Error:
Unexpected response status: 401
WARN SchemaDocument - retrying ... 0
ERROR ServiceNowSchema - Error: Send Error: Unexpected response status: 401
WARN SchemaDocument - com.perspectium.replicator.sql.subscriber.SchemaAccessException: Send Error:
Unexpected response status: 401
WARN SchemaDocument - retrying ... 2
WARN SchemaDocument - Failed to fetch schema for table: incident trying cached version
ERROR SQLTableSubscriber - Unable to establish table meta data.
```

If your subscriber task has incorrect credentials for the instance connection to ServiceNow you may see these errors upon pulling messages from the queue. Double check that your credentials are correct.

### Incorrect Share - Perspectium Credentials

```
ERROR SharerTaskMessagePublisher - post error: Send Error: Send Error: Unexpected response status: 401
ERROR SharerTaskMessagePublisher - closeQueue Error: Send Error: Send Error: Unexpected response status: 401
```

If your share task has incorrect credentials than you may see this error upon connecting to our server.

### Incorrect SQL Credentials

```
WARN TaskDatabase - Access denied for user 'root'@'localhost' (using password: YES)
ERROR SQLDataSource - Error: Access denied for user 'root'@'localhost' (using password: YES)
```

If your SQL credentials are incorrect than you may see that your access is denied and receive the following error messages.

---

## Schema Changes and Corruption

This is something that occurs more often during installation but can occur other times. When your agent is Subscribed it will store schemas from ServiceNow within the **bin** folder. When you receive a message the agent will check this schema and if this schema is corrupted, doesn't exist, is out of date, etc. the agent will check ServiceNow and update the schema. If your ServiceNow credentials are out of date/incorrect than you may see the following errors.

```
ERROR ServiceNowSchema - Error: Send Error: Unexpected response status: 403
WARN SchemaDocument - com.perspectium.replicator.sql.subscriber.SchemaAccessException: Send Error:
Unexpected response status: 403
WARN SchemaDocument - retrying ... 0
...
WARN SchemaDocument - Failed to fetch schema for table: //table_in_message// trying cached version
ERROR SubscriberTask - .....
    com.perspectium.replicator.SubscribeException: Error: SQLSubscriber processMessage error against table:
//table_in_message// error: Unable to obtain or locate the schema for table: //table_in_message//
```

To fix this you can remove the stored schema for this table, do this by going to the **bin/<your\_instance>.service-now.com.schemas\_directory** folder and finding the tables' XML and delete it. Then double check that your ServiceNow credentials are correct and restart the agent. It should then re-acquire the correct schema from ServiceNow.

---

## Subscribed to a Non-Existent Queue

```
WARN AMQP - com.perspectium.api.AMQPException: Failed close channel on URI 'amqp://<your_url>.perspectium.
net' com.rabbitmq.client.AlreadyClosedException: channel is already closed due to channel error; protocol
method: #method<channel.close>(reply-code=404, reply-text=NOT_FOUND - no queue 'psp.out.replicator.
<your_queue>' in vhost '/', class-id=60, method-id=70)
```

This is an issue of subscribing to a queue that cannot be found in the Perspectium Integration Mesh. Verify the queue you specified in the message connection tag is the queue you are sharing data to from your ServiceNow instance and that you have started sharing (the queue is automatically created when you first start sharing).

---

## Decryption Error

```
ERROR Encrypt - Error decrypting value [encrypted:v1OtU71Xu8N/EFQJJ5SS+o2+kRzuseHBK+]
```

This is generally a sign that at some point your encrypted field was edited post encryption causing it to fail to decrypt. This will also normally cascade causing some sort of authentication or 401 error. Check your if you have a **Before Share Script** that is modifying an encrypted field.

---

## Password Encryption

Encrypting sensitive password field values is supported in the **config.xml** and **agent.xml** files. The installer by default will tag any password field due for encryption with the prefix **encrypt:**. When the agent starts, any configuration values prefixed with **encrypt:** will be encrypted and the configuration file will be rewritten with the prefixes changed to **encrypted:** and the password encrypted.

If you change the password to any of your credentials then you will have to stop the agent, replace **encrypted:password** with **encrypt:new\_password** where necessary, and restart the agent.

For example:

```

<share>
  <task>
    <task_name>My_Sharer</task_name>
    <handler>com.perspectium.replicator.sql.sharer.SQLSharer</handler>
    <message_connection user="user" password="encrypt:newpassword">https://<your_url>.perspectium.net<
/message_connection>

    <database_type>mysql</database_type>
    <database_server>localhost</database_server>
    <database_port>3306</database_port>
    <database_user>root</database_user>
    <database_password>encrypt:newpassword</database_password>

    <database>repl_db</database>
    <table_name>incident</table_name>
    <topic>replicator</topic>
    <type>agent</type>
    <name>.bulk</name>
  </task>
</share>

```

## Agent Installation Wizard Error

If you are trying to install the agent on Windows in a console window (such as using Putty to SSH into a different server) and get an error about the **DISPLAY variable** or **initDisplay**, this error is due to the Agent installer trying to create a GUI for installing the agent.

In this case, you will want to run the agent with the console switch:

```
java -jar perspectium-replicator-installer.jar -console
```

## Failed Validation Configuration

The following error displays the failure to create connections to both the Perspectium Integration Mesh and the database while the connection to the instance was fine. In this case, the failures were just due to bad credentials and using the wrong port to connect to my database (3305 instead of 3306).

```

Configuration Validation Report: Fri Feb 02 09:18:58 PST 2018

Agent Information:
  Agent version: Argon_3270
  Agent name: 'example_agent'

Configured Tasks:
  Task: example_agent_subscribe Type: subscribe
      example_agent_subscribe instances: 4

      Attempting to connect to the Message Broker Service at: https://example.perspectium.net as
user: badUser - *** FAILED ***
      Verify your firewall has port 443 open for outbound traffic.
      Verify your <message_connection> entry is properly configured.
      Attempting to connect to the database: psp_repl at host: localhost port: 3305 as user:
badDBUser*** FAILED *** Unable to connect to database: Cannot create PoolableConnectionFactory
(Communications link failure
  The last packet sent successfully to the server was 0 milliseconds ago. The driver has not
received any packets from the server.)
      Verify your database parameters are properly configured.
      Attempting to fetch schema from: https://example.service-now.com- SUCCESS

Validation Completed, results: *** FAILED ***

```

Double check your endpoint URL and credentials to avoid this issue. If all the credentials are correct, and you are still experiencing this issue, contact [support@perspectium.com](mailto:support@perspectium.com).

---

## Accessing Stack Trace and Heap Dump

If you need to get a stack trace and/or heap dump for issues such as an unresponsive Agent or if the Agent is consuming too much memory, you will need the following commands:

### Heap Dump:

```
jmap -dump:format=b,file=<filepath to write dump to>/$PID.hprof $PID
```

**NOTE:** \$PID is the java.exe process taking up much of the memory

For example:

```
jmap -dump:file=C:\PerspectiumLogs\12345.hprof 12345
```

For more information about memory map, [click here](#).

---

### Stack Trace:

```
jstack -l $PID > jstack_$PID.out
```

For example:

```
jstack -l 12345 > jstack_12345.out
```

For more information about stack trace, [click here](#).

---

## Invalid Agent File Format

When the Agent is ran, an additional check to validate the agent.xml file is ran. If the agent.xml file is invalid (i.e. bad file format, missing field characters, etc) then an error message is displayed on the log with a description of the error.

```
Configuration file validation has failed for: agent.xml. File is not a valid format.
```

If the application errors out in this manner, the application will gracefully stop to prevent potentially running the agent with bad/incorrect properties.

In order to correct this issue, fix the agent.xml and run the application again.

When successful, the following message will be in the log:

```
Configuration file validation has passed for: agent.xml. File is a valid format.
```

---

## Java Upgrade on Windows

When you perform a Java upgrade on your machine, this can cause temporary issues with the DataSync Agent. The main issue is the path that the DataSync Agent uses for Java is no longer valid. When the DataSync Agent runs, it is actually running two processes. One for the Java Service Wrapper that allows the Agent to run as a service and another for the actual Agent Java application itself.



When the Agent runs these processes, it will use the system level "java" command to execute this. If the path of this is no longer valid, then the command may not work properly and the Agent will run into issues and not start properly.

The following are suggested troubleshooting techniques for the issues upon upgrading the Java version in your **Windows** machine:

---

## Validate that Java is installed and callable

1

Open a command prompt and execute the following command:

```
java -version
```

2

The version of Java installed should appear:

```
C:\Users\example>java -version
java version "1.8.0_181"
Java(TM) SE Runtime Environment (build 1.8.0_181-b12)
Java Hotspot(TM) 64-Bit Server VM (build 25.181-b12, mixed mode)
```

If not, validate that java is correctly installed and accessible within this environment.

---

## Validate that the installed version matches the path

1

Open a command prompt and execute the following command:

```
where java
```

**i NOTE:** This should output the same version as the previous step.

---

2

The location where Java is located at should appear:

```
C:\Users\example>where java
C:\ProgramData\Oracle\Java\javapath\java.exe
C:\Program Files\Java\jdk1.8.0_181\java.exe
```

You should be able to find these **java.exe** files within your environment and validate that they exists (or link to another). If these do not exist or do not point to the right path, [validate that Java is installed and callable](#) or [validate that the Environment Variables show the right PATH value](#).

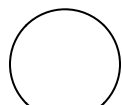
---

## Validate that the Environment Variables show the right PATH value

1

On the Windows taskbar, search **Edit the system environment variables**. Then, under **Advanced** tab, click **Environment Variables**.

---



2

Validate that the **Variables** for the user and the **System Variables** both have a **Path** variable where the **Value** contains the correct path to where the Java bin folder is installed (i.e. C:\Program Files\Java\jdk1.8.0\_181\bin). If not, change the **Value** to the path relative to the correct folder.

## Restart the system

If all of the above suggestions are validated and no changes are needed, reboot the environment. Sometimes during the Java upgrades, the environment requires a restart.

## Force a specific path

As a temporary measure, follow these steps in order to force the Agent to use a specific path for Java rather than the system path for Java:

**⚠ WARNING:** Since you are forcing a Java path, you may run into errors on future Java upgrades.

1

Run the following command as administrator to remove the previous service if it existed:

```
uninstallService.bat
```

2

Verify that the **Perspectium DataSync Agent Service** is removed.

3

Open the **wrapper.conf** file within **conf** folder of the DataSync Agent.

4

Locate the line for **wrapper.conf=java** (should be around line 11).

5

Determine the full path to your java executable, for example:

```
C:\Program Files\Java\jre1.8.0_181\bin\java.exe
```

6

Replace the **wrapper.conf=java** with the escaped \ version of above, for example:

```
wrapper.conf=C:\\Program Files\\Java\\jre1.8.0_181\\bin\\java.exe
```

7

Run the following command for the Agent as administrator:

```
installService.bat
```

8

Verify that the Service appears after this and that when you look at the properties the Path to Executable is showing:

```
C:\Program Files\Java\jre1.8.0_181\bin\java.exe" -classpath .....
```

9

Start the Service.

This is replacing the java call that the Service is performing with the full path of the java call. If this still does not resolve your issue please contact Perspective Support.

### Can't find what you're looking for?

See additional DataSync Agent troubleshooting topics or browse the [Perspective Community Forum](#).