JobScheduler Universal Agent - HTTPS Agent and Master Authentication

Scope

- The connection to JobScheduler Universal Agent can be secured by HTTPS.
- This article describes the steps required to set up secure HTTPS communication without the need of using a reverse proxy (for this use case see JobScheduler Universal Agent - connecting via HTTPS through a proxy).

Prerequisites

The only prerequisite is to have the Java keytools installed with your Java JRE.

Set up a secure connection to the Agent

Step 1: Create the Java Keystore

- Create the Java Keystore using the Keytools from your Java JRE.
    - Generate the Java Keystore with the private key and certificate for the Agent and export the certificate to a second Keystore that is later on used by the Master or use the attached script keygen.sh to perform this task.
    - If not otherwise configured then JobScheduler Agent and Master by default use the password jobscheduler for the respective Keystore.
    - if you choose an individual password for the Agent Keystore then adjust the following properties in the <agent_data>/config/private/private.conf configuration file:
        - Explanations
            - jobscheduler.agent.webserver.https.keystore.file is used for the path to the Keystore
            - jobscheduler.agent.webserver.https.keystore.password is used for the Keystore password
            - jobscheduler.agent.webserver.https.keystore.key-password is used for the password of your private HTTPS certificate
        - Example

Sample private.conf file

```
jobscheduler.agent.webserver.https.keystore {
  file = "C:/ProgramData/sos-berlin.com/jobscheduler/agent110/config/private/private-https.jks"
  # Backslashes are written twice (as in JSON notation):
  # file = "\\\\other-computer\\share\\my-keystore.jks"
  password = "secret"
  key-password = "secret"
}
```

    - For the Master the Keystore that contains the Agents' public trusted certificate is expected with the password jobscheduler.
- For the Agent store the Keystore with the private key in the directory <agent_data>/config/private
    - File name: private-https.jks
- For the Master store the Keystore with the trusted certificate of the Agent in the directory <master_data>/config
    - File name: agent-https.jks

Step 2: Set up authentication between Master and Agent

- Configure the Master password in a file on the Master in the <master_data>/config/private directory:
    - File name: private.conf
    - The file should contain the following entry that specifies a plain text password myjobscheduler4444 that is used by the Master to authenticate against Agents:

```
jobscheduler.master.credentials.password = "myjobscheduler4444"
```

- Specify the Master password in a file on the respective Agent in the directory `<agent_data>/config/private`
    - File name: `private.conf`
    - Specify the Master that will authenticate with the Agent by its JobScheduler ID and password. For example, for two Masters with JobScheduler ID `scheduler_4444` and `scheduler_5555` this file would look like this assuming that the Master password is `myjobscheduler4444`:

    ```
    jobscheduler.agent.auth.users {
      scheduler_4444 = "plain:myjobscheduler4444"
      scheduler_5555 = "sha512:
    9184ddcaa87eb2f95c32f12741035c1e55cef93f7834905f926c4bc419fbc5613e2e141d39fb05d0ec7c66c9bd9e4c8b95b
    74598e0107f863b7f2bd942a9aea0"
    }
    ```

    - For each entry the JobScheduler ID is used as key, the value (in double quotes) includes the hash algorithm followed by a colon and the hashed password.
        - Using `plain` for the hash alogrithm requires a plain text password to be specified. Use of plain text passwords is not recommended as they could be visible to jobs running on that Agent.
        - Using `sha512` for the hash alogithm requires a password that is hashed with the respective algorithm. A number of command line utilitiies to create a sha512 hash from a plain text password can easily be found.

Step 3: Start the Agent for HTTPS

- Start the Agent with the corresponding parameters:
    - Example (using port 44445 for HTTPS): `<agent_data>/bin/jobscheduler_agent -https-port=44445`
- The HTTP port will always be used, even if the Agent is started for communicating over HTTPS. If no HTTP port is indicated when starting the Agent, then the default port (4445) will be used. The reason for this behavior is the requirement that the Agent can be locally controlled by its start script without further need for authentication.
- HTTPS has to be indicated when starting an Agent by use of the parameter `-https-port`.
- The Agent requires a data directory for configuration files and temporary files The data directory has to be indicated when starting the Agent by using the parameter `-data-directory`.
- The above mentioned parameters can be specified as environment variables with the Agent instance script, see Installation & Operation.

Step 4: Create a Process Class assignment for Agents using HTTPS

- Create a Process Class for a job chain or a job.
- Add the Agent URL to the process class using the HTTPS protocol.
- Assign the process class to the job chain or job.
- Example:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<process_class max_processes="30" remote_scheduler="https://my_agent:44445"/>
```

Caveat

- For releases before FEATURE AVAILABILITY STARTING FROM RELEASE 1.10.7 the problem JS-1675 - Getting issue details... STATUS occurs.

  Consider to apply the workaround as specified from the issue.

Change Management References

Key	Summary	T	Created	Updated	Due	Assignee	Reporter	P	Status	Resolution
JS-1675	Loosing connection to Universal Agent when running with HTTPS		Nov 16, 2016	Mar 20, 2017		Joacim Zschimmer	Uwe Risse		RELEASED	Fixed

JS-1663	Universal Agent should have configurable keystore_path and keystore_passphrase		Sep 09, 2016	Oct 04, 2018		Joacim Zschimmer	Mahendra Patidar		RELEASED	Fixed
JS-1633	Agent gets a data directory for configuration and working files		Jun 20, 2016	Jul 08, 2016		Joacim Zschimmer	Joacim Zschimmer		RELEASED	Fixed
JS-1632	JobScheduler Master authenticates itself to Universal Agent		Jun 14, 2016	Jul 08, 2016		Joacim Zschimmer	Joacim Zschimmer		RELEASED	Fixed
JS-1631	Agent restricts access to authenticated users		Jun 10, 2016	Nov 19, 2016		Joacim Zschimmer	Joacim Zschimmer		RELEASED	Fixed
JS-1628	HTTPS for Universal Agent, command line option -ip-address= replaced		May 20, 2016	Oct 14, 2016		Joacim Zschimmer	Joacim Zschimmer		RELEASED	Fixed
JS-1563	Agent Proxy verifies Master identity by SSL Client Certificate		Dec 22, 2015	Feb 12, 2016		TeamEngine	Andreas Püschel		DISMISSED	Won't Fix

7 issues