System Requirements Specification

for

Intel Cloud Integrity Technology 3.0

**Quick Start**

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL’S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

Intel products are not intended for use in medical, life-saving, or life sustaining applications.

Intel may make changes to specifications and product descriptions at any time, without notice.

Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined."

Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

Intel® Cloud Integrity Technology may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Intel and the Intel logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

\*Other names and brands may be claimed as the property of others.

Copyright © 2015 Intel Corporation. All rights reserved.

**Table of Contents**

[Introduction 6](#_Toc442707056)

[Purpose 6](#_Toc442707057)

[Scope 6](#_Toc442707058)

[Definitions, Acronyms, Abbreviations 6](#_Toc442707059)

[Acronyms 6](#_Toc442707060)

[References 7](#_Toc442707061)

[Overview 7](#_Toc442707062)

[Description 7](#_Toc442707063)

[Background and Perspective 7](#_Toc442707064)

[User Stories 8](#_Toc442707065)

[Demonstration 8](#_Toc442707066)

[Production deployment 8](#_Toc442707067)

[Production maintenance 8](#_Toc442707068)

[Integration testing 8](#_Toc442707069)

[Packaging 9](#_Toc442707070)

[Distribution 9](#_Toc442707071)

[Installation 9](#_Toc442707072)

[Uninstallation 9](#_Toc442707073)

[External Interfaces 9](#_Toc442707074)

[Human 9](#_Toc442707075)

[Software 9](#_Toc442707076)

[Hardware 9](#_Toc442707077)

[Communications 10](#_Toc442707078)

[Interoperability 10](#_Toc442707079)

[Features 10](#_Toc442707080)

[Quality Requirements 10](#_Toc442707081)

[User Experience 10](#_Toc442707082)

[Performance 11](#_Toc442707083)

[Scalability 11](#_Toc442707084)

[Capacity 11](#_Toc442707085)

[Availability 12](#_Toc442707086)

[Reliability 12](#_Toc442707087)

[Recoverability 12](#_Toc442707088)

[Maintainability 12](#_Toc442707089)

[Serviceability 13](#_Toc442707090)

[Security 13](#_Toc442707091)

[Regulatory 13](#_Toc442707092)

[Manageability 13](#_Toc442707093)

[Appendix 1 Requirement categories 14](#_Toc442707094)

[Appendix 2 Data model 15](#_Toc442707095)

[Features 15](#_Toc442707096)

[Environment 16](#_Toc442707097)

[Layout 16](#_Toc442707098)

[Credentials 16](#_Toc442707099)

[Settings 16](#_Toc442707100)

[Appendix 3 User interface sketches 17](#_Toc442707101)

[Select environment 17](#_Toc442707102)

[Select features 18](#_Toc442707103)

[Select layout 19](#_Toc442707104)

[Settings 19](#_Toc442707105)

[Enter credentials 20](#_Toc442707106)

[Preview 21](#_Toc442707107)

[Deployment progress 23](#_Toc442707108)

[Network map 24](#_Toc442707109)

[Appendix 4 APIs 25](#_Toc442707110)

[General 25](#_Toc442707111)

[URL 25](#_Toc442707112)

[Headers 25](#_Toc442707113)

[Responses 26](#_Toc442707114)

[SSH 26](#_Toc442707115)

[Connect to host and retrieve host key fingerprint 26](#_Toc442707116)

[Orders 27](#_Toc442707117)

[Create a new order 28](#_Toc442707118)

[Get status of existing order 29](#_Toc442707119)

[Appendix 5 Configuration 34](#_Toc442707120)

[Application Settings 34](#_Toc442707121)

[mtwilson.quickstart.order.dispatch.interval 34](#_Toc442707122)

[mtwilson.quickstart.order.update.interval 35](#_Toc442707123)

[mtwilson.quickstart.order.monitor.interval 35](#_Toc442707124)

[mtwilson.quickstart.order.cleanup.interval 35](#_Toc442707125)

[mtwilson.quickstart.order.cleanup.modified 35](#_Toc442707126)

[mtwilson.quickstart.order.cleanup.accessed 35](#_Toc442707127)

[mtwilson.quickstart.filetransfer.etag 35](#_Toc442707128)

[mtwilson.quickstart.ssh.hostkey.store 35](#_Toc442707129)

[mtwilson.quickstart.ssh.password.store 36](#_Toc442707130)

[General Settings 36](#_Toc442707131)

[jetty.port 36](#_Toc442707132)

[jetty.secure.port 36](#_Toc442707133)

[endpoint.url 36](#_Toc442707134)

# Introduction

This document contains the system requirements specification for the Cloud Integrity Technology 3.0 Quick Start (the Wizard).

## Purpose

The purpose of this document is to articulate the features required to be in the software and document considerations for improvements or additional features that may addressed by future versions of the software.

The intended audience is developers, system engineers, product marketing team, and managers.

## Scope

The Wizard helps a system administrator to deploy the Intel Cloud Integrity Technology 3.0 server components, including Attestation Service, Trust Director, Key Broker, Key Broker Proxy, and OpenStack Extensions for the OpenStack controller.

The Wizard provides an easy-to-use web-based interface to guide the administrator through the available deployment options, monitor progress during deployment, and view the outcome.

The Wizard does not include deployment of the Trust Agent. Automated deployment of the Trust Agent to Linux and Windows hosts may be included in a future version of the Wizard or a future version of the Attestation Service.

The Wizard does not include deployment of OpenStack. However, given an existing OpenStack installation, the Wizard includes deployment of the Intel Cloud Integrity Technology 3.0 OpenStack Extensions for the OpenStack controller.

## Definitions, Acronyms, Abbreviations

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

### Acronyms

CSP - cloud service provider

CSC - cloud service consumer (the enterprise customer that is using the CSP), this term is used only to refer to an enterprise in situations when it is acting specifically as a customer of a CSP

## References

ISO/IEC/IEEE 12207-2008

IEEE 830

RFC 2119 [http://tools.ietf.org/html/rfc2119]

## Overview

The requirements in this specification are organized into packaging, external interfaces, features, and other requirements.

Packaging covers distribution, installation, and uninstallation.

External interfaces include human, software, hardware, and communication interfaces.

Features include all functional requirements, which are things the system “does”.

Quality requirements include all other requirements, commonly known as non-functional requirements, which are qualities or attributes of the system or how it “behaves” either normally or under special or stressful conditions.

There are aspects of the system, such as security, that include both feature requirements and quality requirements.

A list of requirement categories is provided in the appendix because not all categories are covered by this specification.

Possibilities that have been considered but are not currently requirements are listed in the backlog appendix.

# Description

## Background and Perspective

Cloud Integrity Technology software is packaged as a bundle of Linux self-extracting installers, one per service. Each installer can be fully automated through the use of an “environment file” in which the administrator defines variables corresponding to required or optional settings.

In version 2.0, there is an installer for attestation service and an installer for trust agent, and a script to run on OpenStack controller.

In version 3.0, there is an installer for attestation service, trust director, key broker, key broker proxy, trust agent, and OpenStack controller.

Understanding where and how to deploy the components, and which settings are required or optional for each one, can be overwhelming for a first-time user.

The Quick Start (the Wizard) is needed to simplify the Cloud Integrity Technology deployment experience for both testing and operations.

## User Stories

### Demonstration

A potential customer or partner uses the Wizard to quickly deploy all Cloud Integrity Technology components in a minimal or complete laboratory environment and is able to begin using the new installation within minutes.

An engineer should be able to:

* Download a single installer to a host or VM and install it there with a single command.
* Use the Wizard to deploy all Cloud Integrity Technology components (except Trust Agent) to an existing OpenStack controller.
* Use the Wizard to deploy each Cloud Integrity Technology components (except Trust Agent) to a separate host or VM, integrate it with an existing OpenStack controller.

### Production deployment

A customer or partner uses the Wizard to quickly deploy all Cloud Integrity Technology components in a production or sandbox environment with load balancing and high availability configurations and is able to begin using the new installation within minutes.

NOTE: support for load balancing and high availability is not currently a requirement.

### Production maintenance

An existing user of Cloud Integrity Technology 3.0 uses the Wizard to update their current installation of Intel Cloud Integrity Technology to a later version, for example from 3.0 to 3.1.

NOTE: this user story is not currently a requirement.

### Integration testing

The continuous integration server connects to the Wizard and provides an environment layout and links to the recently built installers for each service, and the Wizard automatically deploys the components to the environment. The continuous integration server can then provide the environment layout to an automated testing suite for conducting integration tests.

# Packaging

## Distribution

The Wizard is REQUIRED to be packaged as a Linux self-extracting installer.

## Installation

The Wizard MUST have a “zero-configuration” installation procedure wherein the user is not required to configure anything except a host and root password in order to produce a working installation.

The Wizard MAY have optional configuration settings that the user can set before installation in order to affect the installation itself, such as the location where the software will be extracted.

The Wizard MAY have optional configuration settings that the user can provide to affect the deployment.

## Uninstallation

The Wizard MUST have a “one touch” uninstallation procedure that leaves related data and configuration intact (uninstall).

The Wizard MUST have a “one touch” uninstallation procedure that erases related data and configuration (purge).

# External Interfaces

## Human

The Wizard SHALL provide a visual human user interface using HTML5 and associated technologies accessible via a modern browser.

## Software

The Wizard SHALL execute as a Linux process.

The Wizard MUST provide a non-interactive method for providing input and initiating the installation of services.

## Hardware

The Wizard SHALL operate inside a virtual machine or physical machine.

## Communications

The Wizard SHALL provide an HTTP interface for local or remote clients to interact via a representational state transfer (REST) message style.

## Interoperability

The Wizard is REQUIRED to interoperate with Google Chrome 46, Mozilla Firefox 42, Microsoft Internet Explorer 10 and Microsoft Internet Explorer 11.

# Features

The Wizard is REQUIRED to be able to install all Cloud Integrity Technology services on a specified host or VM that already has OpenStack Nova Scheduler installed.

The Wizard is REQUIRED to be able to install the Trust Agent on one or more remote Linux KVM hosts that already have OpenStack Nova Compute installed.

The Wizard MUST support the security configuration of the installed services in the appropriate order, to the extent that the user has selected a reasonable set of services to install at one time.

The Wizard MUST securely configure TLS between installed services that communicate with each other using HTTPS during operation.

# Quality Requirements

## User Experience

The Wizard MUST provide the easiest, simplest deployment options requiring the least amount of configuration as the default.

The Wizard MUST show some processing indicator during installation of services; calculation and display of actual progress is OPTIONAL and the Wizard MAY simply show an indefinite wait indicator.

The Wizard MUST display any error preventing successful installation of services. The Wizard SHALL NOT perform any automated “roll back” of partially installed services so the system administrator may login interactively to check the state of the system.

The Wizard SHOULD allow a user to complete the setup process (everything prior to the actual deployment) in less than five minutes after it has been fully installed, defaults are used, user can read at least 250 words per minute, and user is not using the browser or operating system accessibility features. Note: this is a usability requirement, not a performance requirement, because it depends on an assumption of a “normal” user.

## Performance

The Wizard self-installation SHOULD complete in less than 10 minutes on a host with the minimum hardware requirements.

## Scalability

The Wizard MUST support multiple users deploying to separate environments concurrently.

## Capacity

The Wizard SHALL operate on a host with at least 4GB memory total and at least 2GB memory free prior to starting the Wizard, at least 32GB disk total and at least 5GB disk available prior to installation, running on an Intel Xeon CPU or 1 vCPU.

Table 1 Capacity Requirements for Quick Start

|  |  |  |
| --- | --- | --- |
| **Resource** | **Minimum** | **Notes** |
| CPU | Intel Xeon or 1 vCPU | Application is not CPU-intensive |
| Memory | 2GB | Recommend total 4GB system memory with 2GB free for CIT |
| Disk | 5GB | Recommend 10GB disk space free prior to installing CIT quick start |
| Network | 1GBit | Installation files for components are large and time required to transfer these to target hosts directly impacts the total time required to deploy CIT |

The following table shows sizes of individual component installers. Capacity requirements for each component when installed are not specified here.

Table 2 Capacity Requirements for CIT Components

|  |  |  |
| --- | --- | --- |
| **Component** | **Disk** | **Notes** |
| Attestation Service | 1GB | Installer size |
| Key Broker | 200MB | Installer size |
| Key Broker Proxy | 200MB | Installer size |
| Trust Director | 225MB | Installer size |
| Trust Agent | 200MB | Installer size |
| OpenStack Extensions | 1MB | Installer size |

## Availability

The Wizard is REQUIRED to respond to 99% of incoming HTTP requests coming at a maximum rate of 120 requests per minute.

## Reliability

The Wizard MUST verify remote host login credentials prior to commencing deployment.

The Wizard MAY verify remote credentials provided by user in the settings page prior to commencing deployment.

The Wizard MAY fail if remote credentials provided by user in the settings page are incorrect.

The Wizard MAY fail if network errors occur during deployment.

The Wizard MAY fail if specified remote hosts do not have sufficient memory or disk space for the components to be installed on them.

The Wizard MAY fail if the remote hosts do not have the prerequisite software installed prior to deployment.

## Recoverability

The Wizard MUST restart automatically after a reboot.

The Wizard MAY restart automatically after a crash, if sufficient disk space is available.

## Maintainability

The Wizard MUST allow setting the log level to DEBUG, INFO, WARN, or ERROR.

The Wizard MUST log internal details at the DEBUG level.

The Wizard MUST log user actions and results at the INFO level.

The Wizard MUST log alerts and notices to the administrator at the WARN level.

The Wizard MUST log error conditions at the ERROR level.

The Wizard MAY log an incident identification tag and provide the same tag to the client in its error response.

The Wizard MUST allow an administrator to update or patch the code and restart the server for the patches to take effect.

## Serviceability

The Wizard MUST allow an administrator to update the installers by copying them to a designated location.

The Wizard MUST allow an administrator to view and edit its configuration file and restart the server for the changes to take effect.

The Wizard MUST NOT require any routine service by the administrator beyond ensuring sufficient memory and disk space is available in accordance with the capacity specification.

## Security

The Wizard MUST NOT permanently store any user-provided login credentials such as root password or SSH keys - all such credentials must be discarded after use.

The Wizard SHALL NOT require the user to login to the Wizard itself because there is no stored data to protect - user must have login credentials for deploying components to any host (including root password for localhost deployment).

## Regulatory

The Wizard MUST qualify for export outside the United States.

## Manageability

When deploying the same service across multiple target hosts (for a high availability or load balancing configuration) the Wizard SHALL choose the same port numbers for the service on each host for that deployment.

The Wizard MAY choose a different port number for the service for separate deployments.

Services have default and alternate ports. The alternate ports are used when more than one service is deployed on the same host, in order to avoid port conflicts. The following table summarizes the default and alternate port choices for each service.

Table 3 Default and alternate port numbers

|  |  |  |
| --- | --- | --- |
| **Service** | **Default ports** | **Alternate ports** |
| Trust Agent | http:80  https:1443 | http:17080  https:17443 |
| Attestation | http:8080  https:8443 | http:18080  https:18443  Note: attestation service currently does not support setting alternate ports, so it always installs at 8080/8443 |
| Trust Director | http:80  https:443 | http:19080  https:19443 |
| Key Broker | http:80  https:443 | http:20080  https:20443 |
| Key Broker Proxy | http:80  https:443 | http:21080  https:21443 |
| OpenStack | http:80  https:443 | No alternates; wizard does not reconfigure OpenStack |

# Appendix 1 Requirement categories

The following list shows typical non-functional requirement categories. If the specification includes at least one requirement from a category, it is marked as being included in this specification.

It’s possible to organize the requirements into groups such as reliability (functionality, availability, recoverability, maintainability, and manageability), performance (response time, memory usage, hardware requirements, scalability, and capacity), usability, external interfaces (human, software, hardware, and communication), and security (integrity, confidentiality, availability). Some of these are mentioned below but no effort is made at this time to fully organize the list.

Table 4 Requirement categories

|  |  |
| --- | --- |
| **Category** | **Included** |
| User Experience | Yes |
| Packaging (Distribution, Installation, Uninstallation) | Yes |
| External interfaces (Human, Software, Hardware, Communication, Interoperability) | Yes |
| Performance (Response Time, Memory Usage) | Yes |
| Scalability | Yes |
| Capacity | Yes |
| Availability | Yes |
| Reliability | Yes |
| Recoverability | Yes |
| Maintainability | Yes |
| Serviceability | Yes |
| Security (Privacy, Data Integrity, Confidentiality, Detect, Protect, Recovery) | Yes |
| Regulatory | Yes |
| Manageability | Yes |
| Environmental | No |

# Appendix 2 Data model

NOTE: this section needs to be updated, as of 13 Jan 2016

The Wizard accepts user input for the following data:

## Features

The Wizard provides a list of product features to be included in the deployment, organized into categories, and the user must choose from these. The selected features affect which services will be installed.

**Attestation**

* **attestation-host: Host BIOS, kernel, initrd, hypervisor**; corresponds to the CIT 1.0/2.0 “host attestation” use case and implies that Attestation Service will be installed; every Trust Agent 3.0 installation supports this
* **attestation-host-xm: Host apps, drivers, and configuration**; corresponds to the CIT 3.0 “host attestation with extended measurements” use case and implies that Attestation Service and Trust Director will be installed; every Trust Agent 3.0 installation supports this and includes tboot-xm; this feature is required when choosing attestation-vm or attestation-container
* **attestation-vm: Workload integrity for virtual machines**; corresponds to the CIT 3.0 “workload integrity” use case and implies Trust Director will be installed; this feature is required when choosing enforcement-vm-encryption
* **attestation-container: Docker containers**; corresponds to the CIT 3.0 “workload integrity” use case and implies Trust Director will be installed

**Enforcement**

* **enforcement-vm-encryption: Encrypted workloads run only on trusted servers**; corresponds to the CIT 3.0 “workload confidentiality” use case and implies Key Broker and Key Broker Proxy will be installed

**Integrations**

* **integration-openstack: OpenStack**; corresponds to the CIT 3.0 “workload integrity” and “workload confidentiality” use cases and implies OpenStack Extensions will be installed on an existing OpenStack controller

## Environment

The Wizard provides a list of deployment environments from which the user must select one. The selected environment affects which services will be installed.

* **PRIVATE: Private network**; corresponds to a private Enterprise datacenter or the management network of a CSP. When choosing a private network, all necessary components will be installed in the same environment.
* **PROVIDER: CSP**; corresponds to the cloud service provider in the CIT 3.0 architecture diagram. Implies installation of attestation service, trust director, key broker proxy, and openstack extensions depending on selected features, but not the key broker.
* **SUBSCRIBER: CSC**; corresponds to the enterprise cloud service consumer in the CIT 3.0 architecture diagram. Implies installation of trust director and key broker depending on selected features, but not the key broker proxy.

## Layout

The Wizard provides a list of service deployment layouts from which the user must select one. The selected layout affects where each service will be installed.

* **LOCALHOST: Localhost**; all services will be installed on the local host, no configuration required
* **SINGLE\_HOST: Remote all-in-one**; all services will be installed on a single remote host that already has OpenStack installed
* **CUSTOM: Custom**; service deployment is customized and user inputs specific hosts to use for each component.

## Credentials

The Wizard provides a list of required credentials to login to any hosts or VMs where services will be installed. This affects the ability to install the software, and is required even if all services will be deployed on localhost.

The only supported credential is the root password.

## Settings

The Wizard provides a list of configurable settings the user may edit after services are installed. The same settings could also be configured in each component individually.

**Attestation**

* **saml.validity.seconds: Cache duration**; how long SAML assertions stay in the cache

**Trust Director**

* **OpenStack integration**; the OpenStack URL (glance.ip and glance.port), username (glance.image.store.username), and password (glance.image.store.password), and tenant name (glance.tenant.name); only shown if user selected the OpenStack integration feature; in director.env the same variables are GLANCE\_IMAGE\_STORE\_IP, GLANCE\_IMAGE\_STORE\_PORT, GLANCE\_IMAGE\_STORE\_USERNAME, GLANCE\_IMAGE\_STORE\_PASSWORD, and TENANT\_NAME.

**Key Broker**

* **Key server**; the type of key server to use (Barbican or KMIP or None) with default KMIP; only shown if user selected the enforcement feature
* **Key server URL**; the URL to the key server, only shown if user selected KMIP or Barbican key server
* **Key server username**; the username for logging in to the key server; only shown if user selected KMIP or Barbican key server
* **Key server password**; the password for logging in to the key server; only shown if user selected KMIP or Barbican key server

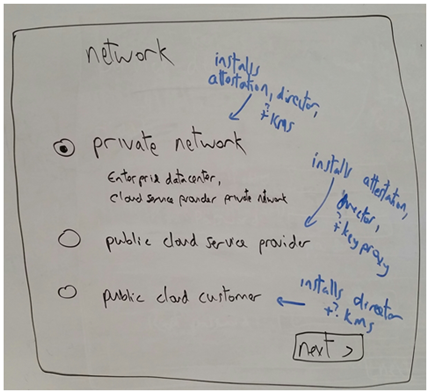
# Appendix 3 User interface sketches

## Select environment

Private cloud needs attestation, director, and if encryption selected also key broker & proxy

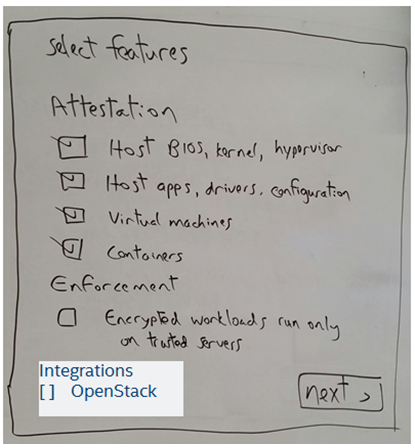
Public cloud provider needs attestation, director for host policy, and if encryption selected also key broker proxy

Public cloud customer needs director for VM policy, and if encryption selected also key broker



## Select features

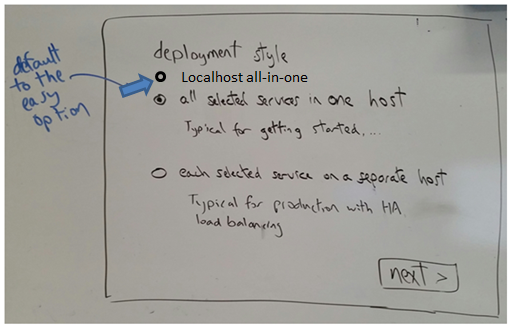
High level selection of features allows customer to declare their intent, which enables us to make decisions later on in the process.



## Select layout

We get the user’s intent first, then we can make decisions based on that and simplify subsequent screens.

NOTE: the “localhost” was added later so default would be “localhost” all-in-one.



## Settings

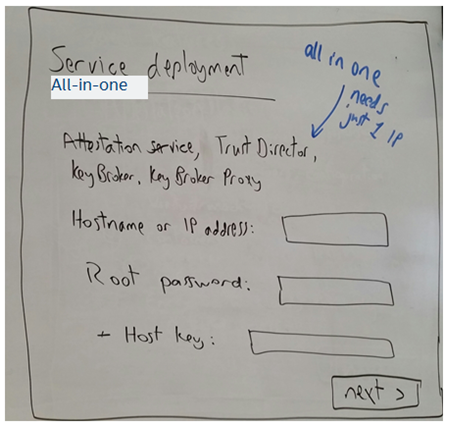
Only sections related to selected features would be shown.

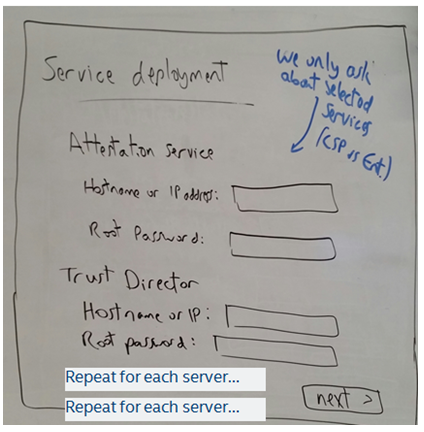
## Enter credentials

For BOTH modes, if openstack feature selected, prompt for openstack host name/ip & root password for installing openstack extensions

need to prompt for Nova , and have checkbox prompt for Horizon to say if it’s same host as Nova, and have a + button after that so user can add more of same host (Nova, or Horizon in the Horizon section) , the box for password/ssh needs to be there but for all hosts after first one, start with disabled and “same as previous” with checkbox (default checked) that user can uncheck to customize password/ssh key for each additional host.

Note: If the localhost layout option was selected, the credentials screen will only ask for localhost root password an not the hostname or host key.

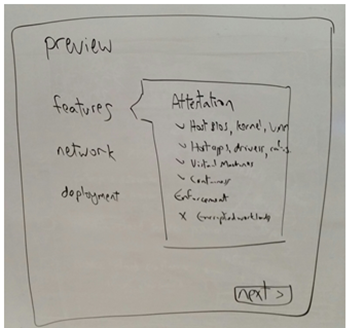


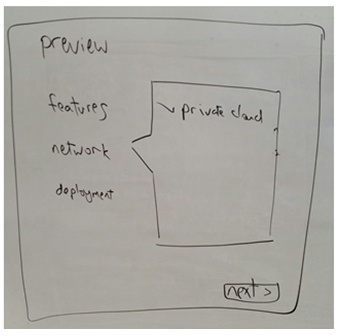


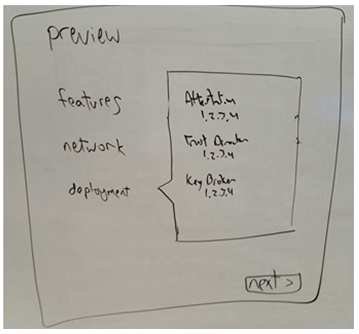
## Preview

The preview allows the user to confirm their selections before deploying the components.

These sketches show three tabs for preview, but these may be combined into one screen.

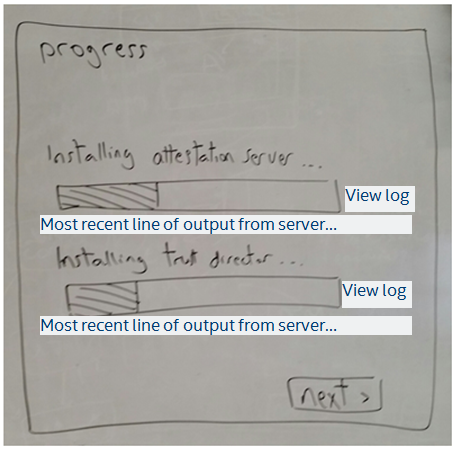






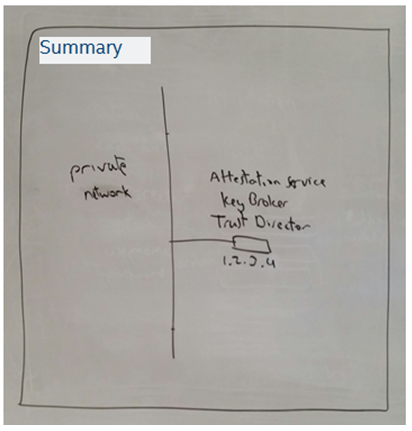
## Deployment progress

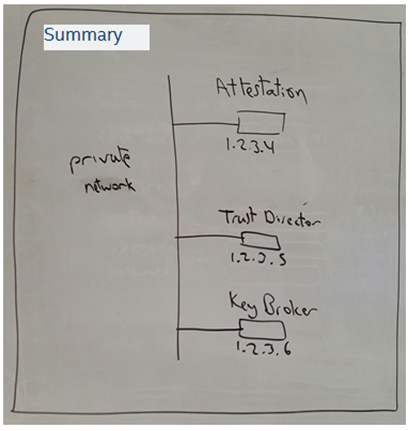
The progress will be an indefinite indicator, but a future version may show a more detailed progress as shown in this sketch.



## Network map

This would be shown after the deployment is complete.





# Appendix 4 APIs

All IP addresses mentioned in this specification are in the documentation example address space 198.51.100.0/24, in accordance with RFC 5737 <https://tools.ietf.org/html/rfc5737>

Certain addresses within the example space will be used consistently to refer to various components as specified in the table below.

Table 5 Example IP addresses

|  |  |
| --- | --- |
| **IP Address** | **Component** |
| 198.51.100.17 | Quickstart server |
| 198.51.100.33 - 198.51.100.62 | Any host (physical or virtualized) |
| 198.51.100.65 - 198.51.100.94 | Compute node (physical) |
| 198.51.100.129 - 198.51.100.158 | Virtual machine |

## General

Unless specified otherwise, clients can incorporate the following specifications for each API:

### URL

The quickstart server base URL is assumed to be prepended to all the API paths. If the quickstart server is installed at 198.51.100.17, the base URL would typically be https://198.51.100.17. For an API with a specified path of /v1/rpc/example the full URL would be https://198.51.100.17/v1/rpc/example.

### Headers

The quickstart application allows anonymous access but this still requires clients to specifically request it. This is done by authenticating as user “anonymous” with an empty password either using HTTP BASIC (for software clients) or by using the token login API to obtain a token (user interface).

In the global request headers below, the HTTP BASIC authorization parameter is the base-64 encoding of username anonymous with an empty password.

Clients should send the following HTTP headers with all API requests unless specified otherwise:

Authorization: Basic YW5vbnltb3VzOg==

Content-Type: application/json

Accept: application/json

### Responses

#### Bad Request

HTTP code will be 400. This indicates the server could not understand (parse) the request. The response may look like this:

Incident-Tag: 7053316b

Content-Type: text/plain

Content-Length: 12

Bad argument

#### Not Found

HTTP code will be 404.

#### Success

HTTP code will be 200.

A success response MAY also include data, an object containing primary data the server is sharing with the client.

A success response MAY also include extra, an object describing related information such as the linked data. For example if data contains blog posts, extra may contain comments for those posts or author information.

#### Failure

HTTP code will be 200. This indicates the server parsed the request successfully, but the request may be invalid (asking for something that isn’t possible) or there may have been an error during processing.

A failure response MUST include faults, a non-empty array with elements describing what went wrong.

A failure response MAY also include extra, an object describing related information such as the request input.

A failure response MUST NOT include data, an object that contains successful response data.

## SSH

### Connect to host and retrieve host key fingerprint

#### Request

POST /v1/rpc/ssh-login

#### Request body

{

"host":"198.51.100.33",

"port":22,

"timeout":2000

}

The host is required can be a hostname, fully qualified DNS name, or IP address.

The port is optional. Default value is 22.

The timeout is optional. Default value is 15000 (ms). In the example above, the timeout is specified to be 2000 (ms), representing 2 seconds.

#### Response on success

{

"data": {

"host": "198.51.100.33",

"port": 22,

"username": "root",

"public\_key\_digest": "22952a72e24194f208200e76fd3900da",

"timeout": 2000

}

}

#### Response on failure

{

"faults": [{

"type": "com.intel.mtwilson.deployment.jaxrs.faults.Connection",

"description": "198.51.100.33"

}],

"extra": {

"host": "198.51.100.33",

"port": 22,

"username": "root",

"timeout": 2000

}

}

## Orders

To deploy Cloud Integrity Technology, submit the deployment “order”, or request.

After an order is successfully submitted, a client can monitor progress by periodically requesting status updates until the order is complete.

The following examples are for deploying only Attestation Service. A deployment with additional components would have a much longer response due to more packages and tasks involved.

Settings are required only when needed by specific features being deployed. When a setting is optional it can be omitted or sent as empty string. Missing, null, or empty settings will cause the server to use a default.

### Create a new order

#### Request

POST /v1/quickstart/orders

#### Request body

{

"features": [

"attestation\_host"

],

"network\_role": "PRIVATE",

"targets": [

{

"host": "10.1.68.34",

"port": 22,

"username": "root",

"password": "\*\*\*\*\*\*\*\*",

"public\_key\_digest": "22952a72e24194f208200e76fd3900da",

"packages": [

"attestation\_service"

]

}

],

"settings": {

"saml.validity.seconds": "900",

"director.port.http": "",

"director.port.https": "",

"director.tls.cert.sha1": "",

"director.glance.host": "",

"director.glance.port": "",

"director.glance.username": "",

"director.glance.password": "",

"director.glance.tenant": "",

"kms.host": "",

"kms.port.http": "",

"kms.port.https": "",

"kms.tls.cert.sha1": "",

"kms.key.provider": "",

"kms.kmip.url": "",

"kms.barbican.url": "",

"kms.barbican.project": "",

"kmsproxy.host": "",

"kmsproxy.port.http": "",

"kmsproxy.port.https": "",

"mtwilson.host": "",

"mtwilson.port.http": "",

"mtwilson.port.https": "",

"mtwilson.tls.cert.sha1": "",

"director.mtwilson.username": "",

"director.mtwilson.password": ""

}

}

#### Response for well-formatted request

This response does NOT indicate that there are no errors with the request; it only means the request was accepted for processing and client can check later for its status. If the request cannot be processed the status will reflect the errors.

{

"id": "fa5c128c-af92-420c-ba4f-d17213210672",

"links": {

"status": "\/v1\/quickstart\/orders\/fa5c128c-af92-420c-ba4f-d17213210672",

"cancel": "\/v1\/quickstart\/orders\/fa5c128c-af92-420c-ba4f-d17213210672\/cancel"

},

"features": [

"attestation\_host"

],

"targets": [

{

"host": "10.1.68.34",

"port": 22,

"packages": [

"attestation\_service"

]

}

],

"network\_role": "PRIVATE",

"status": "PENDING"

}

### Get status of existing order

#### Request

GET /v1/quickstart/orders/fa5c128c-af92-420c-ba4f-d17213210672

#### Response for invalid order

The following example response was sent for a request that included a “non\_existent\_package” instead of a known package name such as “attestation\_service”.

{

"id": "6c1ba765-14bc-42b2-b334-d406abda326b",

"etag": "037b316ac648eb3822dfa5b925e5cb639ca7c070",

"modified\_on": 1451945617793,

"features": [

"attestation\_host"

],

"targets": [

{

"host": "10.1.68.34",

"port": 22,

"packages": [

"non\_existent\_package"

]

}

],

"network\_role": "PRIVATE",

"status": "ERROR",

"progress": 0,

"progress\_max": 0

}

#### Response for order in progress

The following example response was sent for an order after the first three tasks were completed and the fourth task was on step 6 of 25.

{

"id": "fa5c128c-af92-420c-ba4f-d17213210672",

"links": {

"status": "\/v1\/quickstart\/orders\/fa5c128c-af92-420c-ba4f-d17213210672",

"cancel": "\/v1\/quickstart\/orders\/fa5c128c-af92-420c-ba4f-d17213210672\/cancel"

},

"etag": "b69891624428f960e146f7e2ad1f7eb4ad5e1027",

"modified\_on": 1451942437148,

"features": [

"attestation\_host"

],

"targets": [

{

"host": "10.1.68.34",

"port": 22,

"packages": [

"attestation\_service"

]

}

],

"network\_role": "PRIVATE",

"status": "ACTIVE",

"progress": 3,

"progress\_max": 6,

"tasks": [

{

"id": "1517e8df-eb24-49e9-b412-41e3796b49f7",

"links": {

"output": "\/v1\/quickstart\/tasks\/1517e8df-eb24-49e9-b412-41e3796b49f7\/output"

},

"name": "com.intel.mtwilson.deployment.task.SynchronizeSoftwarePackageTargets",

"sequence": 6,

"progress": 0,

"progress\_max": 0,

"data": {

"host\_csv": "10.1.68.34",

"done": false,

"package\_name": "attestation\_service",

"task\_directory": "\/opt\/cit\/repository\/tasks\/1517e8df-eb24-49e9-b412-41e3796b49f7",

"active": false

}

},

{

"id": "034a8ea3-8002-439b-b633-801673d91409",

"links": {

"output": "\/v1\/quickstart\/tasks\/034a8ea3-8002-439b-b633-801673d91409\/output"

},

"name": "com.intel.mtwilson.deployment.task.PostconfigureAttestationService",

"sequence": 5,

"progress": 0,

"progress\_max": 0,

"data": {

"host": "10.1.68.34",

"done": false,

"task\_directory": "\/opt\/cit\/repository\/tasks\/034a8ea3-8002-439b-b633-801673d91409",

"active": false

}

},

{

"id": "2995c160-39a2-4e4e-a4e6-ebac3badcdec",

"links": {

"output": "\/v1\/quickstart\/tasks\/2995c160-39a2-4e4e-a4e6-ebac3badcdec\/output"

},

"name": "com.intel.mtwilson.deployment.task.FileTransfer",

"sequence": 3,

"progress": 2887,

"progress\_max": 2887,

"data": {

"host": "10.1.68.34",

"done": false,

"task\_directory": "\/opt\/cit\/repository\/tasks\/2995c160-39a2-4e4e-a4e6-ebac3badcdec",

"active": false,

"filename\_csv": "mtwilson.env",

"file\_transfer\_manifest": [

{

"source": "\/opt\/cit\/repository\/tasks\/6ef5317d-8aa2-46b2-9bb2-5d4fcecb9032\/mtwilson.env",

"targetPath": "mtwilson.env",

"permissions": null

}

]

}

},

{

"id": "51c33390-4b09-490f-8812-053057deb660",

"links": {

"output": "\/v1\/quickstart\/tasks\/51c33390-4b09-490f-8812-053057deb660\/output"

},

"name": "com.intel.mtwilson.deployment.task.FileTransfer",

"sequence": 1,

"progress": 804479342,

"progress\_max": 804479342,

"data": {

"host": "10.1.68.34",

"done": false,

"task\_directory": "\/opt\/cit\/repository\/tasks\/51c33390-4b09-490f-8812-053057deb660",

"active": false,

"filename\_csv": "mtwilson-server-3.0-SNAPSHOT-jdk\_glassfish\_monit.bin, mtwilson-server-3.0-SNAPSHOT-jdk\_glassfish\_monit.bin.mark, monitor.sh",

"file\_transfer\_manifest": [

{

"source": "\/opt\/cit\/repository\/packages\/attestation\_service\/mtwilson-server-3.0-SNAPSHOT-jdk\_glassfish\_monit.bin",

"targetPath": "mtwilson-server-3.0-SNAPSHOT-jdk\_glassfish\_monit.bin",

"permissions": null

},

{

"source": "\/opt\/cit\/repository\/packages\/attestation\_service\/mtwilson-server-3.0-SNAPSHOT-jdk\_glassfish\_monit.bin.mark",

"targetPath": "mtwilson-server-3.0-SNAPSHOT-jdk\_glassfish\_monit.bin.mark",

"permissions": null

},

{

"source": "\/opt\/cit\/repository\/scripts\/monitor.sh",

"targetPath": "monitor.sh",

"permissions": null

}

]

}

},

{

"id": "a6913d9d-6420-4362-a523-9493acfb2f5f",

"links": {

"output": "\/v1\/quickstart\/tasks\/a6913d9d-6420-4362-a523-9493acfb2f5f\/output"

},

"name": "com.intel.mtwilson.deployment.task.RemoteInstall",

"sequence": 4,

"progress": 6,

"progress\_max": 25,

"data": {

"host": "10.1.68.34",

"done": false,

"package\_name": "attestation\_service",

"task\_directory": "\/opt\/cit\/repository\/tasks\/a6913d9d-6420-4362-a523-9493acfb2f5f",

"active": false,

"executable\_path": "mtwilson-server-3.0-SNAPSHOT-jdk\_glassfish\_monit.bin"

}

},

{

"id": "6ef5317d-8aa2-46b2-9bb2-5d4fcecb9032",

"links": {

"output": "\/v1\/quickstart\/tasks\/6ef5317d-8aa2-46b2-9bb2-5d4fcecb9032\/output"

},

"name": "com.intel.mtwilson.deployment.task.PreconfigureAttestationService",

"sequence": 2,

"progress": 1,

"progress\_max": 1,

"data": {

"host": "10.1.68.34",

"done": false,

"data": {

},

"package\_name": "attestation\_service",

"task\_directory": "\/opt\/cit\/repository\/tasks\/6ef5317d-8aa2-46b2-9bb2-5d4fcecb9032",

"active": false,

"file\_transfer\_manifest": [

{

"source": "\/opt\/cit\/repository\/tasks\/6ef5317d-8aa2-46b2-9bb2-5d4fcecb9032\/mtwilson.env",

"targetPath": "mtwilson.env",

"permissions": null

}

]

}

}

]

}

#### Response for complete order

The following example response was sent after all six tasks in the deployment order were completed.

{

"id": "fa5c128c-af92-420c-ba4f-d17213210672",

"links": {

"status": "\/v1\/quickstart\/orders\/fa5c128c-af92-420c-ba4f-d17213210672",

"cancel": "\/v1\/quickstart\/orders\/fa5c128c-af92-420c-ba4f-d17213210672\/cancel"

},

"etag": "6b30e47642d9773271da2a4dd70b4f1e1fdcb106",

"modified\_on": 1451942772439,

"features": [

"attestation\_host"

],

"targets": [

{

"host": "10.1.68.34",

"port": 22,

"packages": [

"attestation\_service"

]

}

],

"network\_role": "PRIVATE",

"status": "DONE",

"progress": 6,

"progress\_max": 6,

"tasks": [

{

"id": "1517e8df-eb24-49e9-b412-41e3796b49f7",

"links": {

"output": "\/v1\/quickstart\/tasks\/1517e8df-eb24-49e9-b412-41e3796b49f7\/output"

},

"name": "com.intel.mtwilson.deployment.task.SynchronizeSoftwarePackageTargets",

"sequence": 6,

"progress": 1,

"progress\_max": 1,

"data": {

"host\_csv": "10.1.68.34",

"done": false,

"package\_name": "attestation\_service",

"task\_directory": "\/opt\/cit\/repository\/tasks\/1517e8df-eb24-49e9-b412-41e3796b49f7",

"active": false

}

},

{

"id": "034a8ea3-8002-439b-b633-801673d91409",

"links": {

"output": "\/v1\/quickstart\/tasks\/034a8ea3-8002-439b-b633-801673d91409\/output"

},

"name": "com.intel.mtwilson.deployment.task.PostconfigureAttestationService",

"sequence": 5,

"progress": 1,

"progress\_max": 1,

"data": {

"host": "10.1.68.34",

"done": false,

"task\_directory": "\/opt\/cit\/repository\/tasks\/034a8ea3-8002-439b-b633-801673d91409",

"active": false

}

},

{

"id": "2995c160-39a2-4e4e-a4e6-ebac3badcdec",

"links": {

"output": "\/v1\/quickstart\/tasks\/2995c160-39a2-4e4e-a4e6-ebac3badcdec\/output"

},

"name": "com.intel.mtwilson.deployment.task.FileTransfer",

"sequence": 3,

"progress": 2887,

"progress\_max": 2887,

"data": {

"host": "10.1.68.34",

"done": false,

"task\_directory": "\/opt\/cit\/repository\/tasks\/2995c160-39a2-4e4e-a4e6-ebac3badcdec",

"active": false,

"filename\_csv": "mtwilson.env",

"file\_transfer\_manifest": [

{

"source": "\/opt\/cit\/repository\/tasks\/6ef5317d-8aa2-46b2-9bb2-5d4fcecb9032\/mtwilson.env",

"targetPath": "mtwilson.env",

"permissions": null

}

]

}

},

{

"id": "51c33390-4b09-490f-8812-053057deb660",

"links": {

"output": "\/v1\/quickstart\/tasks\/51c33390-4b09-490f-8812-053057deb660\/output"

},

"name": "com.intel.mtwilson.deployment.task.FileTransfer",

"sequence": 1,

"progress": 804479342,

"progress\_max": 804479342,

"data": {

"host": "10.1.68.34",

"done": false,

"task\_directory": "\/opt\/cit\/repository\/tasks\/51c33390-4b09-490f-8812-053057deb660",

"active": false,

"filename\_csv": "mtwilson-server-3.0-SNAPSHOT-jdk\_glassfish\_monit.bin, mtwilson-server-3.0-SNAPSHOT-jdk\_glassfish\_monit.bin.mark, monitor.sh",

"file\_transfer\_manifest": [

{

"source": "\/opt\/cit\/repository\/packages\/attestation\_service\/mtwilson-server-3.0-SNAPSHOT-jdk\_glassfish\_monit.bin",

"targetPath": "mtwilson-server-3.0-SNAPSHOT-jdk\_glassfish\_monit.bin",

"permissions": null

},

{

"source": "\/opt\/cit\/repository\/packages\/attestation\_service\/mtwilson-server-3.0-SNAPSHOT-jdk\_glassfish\_monit.bin.mark",

"targetPath": "mtwilson-server-3.0-SNAPSHOT-jdk\_glassfish\_monit.bin.mark",

"permissions": null

},

{

"source": "\/opt\/cit\/repository\/scripts\/monitor.sh",

"targetPath": "monitor.sh",

"permissions": null

}

]

}

},

{

"id": "a6913d9d-6420-4362-a523-9493acfb2f5f",

"links": {

"output": "\/v1\/quickstart\/tasks\/a6913d9d-6420-4362-a523-9493acfb2f5f\/output"

},

"name": "com.intel.mtwilson.deployment.task.RemoteInstall",

"sequence": 4,

"progress": 25,

"progress\_max": 25,

"data": {

"host": "10.1.68.34",

"done": false,

"package\_name": "attestation\_service",

"task\_directory": "\/opt\/cit\/repository\/tasks\/a6913d9d-6420-4362-a523-9493acfb2f5f",

"active": false,

"executable\_path": "mtwilson-server-3.0-SNAPSHOT-jdk\_glassfish\_monit.bin"

}

},

{

"id": "6ef5317d-8aa2-46b2-9bb2-5d4fcecb9032",

"links": {

"output": "\/v1\/quickstart\/tasks\/6ef5317d-8aa2-46b2-9bb2-5d4fcecb9032\/output"

},

"name": "com.intel.mtwilson.deployment.task.PreconfigureAttestationService",

"sequence": 2,

"progress": 1,

"progress\_max": 1,

"data": {

"host": "10.1.68.34",

"done": false,

"data": {

},

"package\_name": "attestation\_service",

"task\_directory": "\/opt\/cit\/repository\/tasks\/6ef5317d-8aa2-46b2-9bb2-5d4fcecb9032",

"active": false,

"file\_transfer\_manifest": [

{

"source": "\/opt\/cit\/repository\/tasks\/6ef5317d-8aa2-46b2-9bb2-5d4fcecb9032\/mtwilson.env",

"targetPath": "mtwilson.env",

"permissions": null

}

]

}

}

]

}

# Appendix 5 Configuration

The configuration settings described in this section are used to affect the behavior of the quickstart server. The Application Settings section describes settings that are specific to the quickstart server. The General Settings section describes settings that affect other components present in the server.

## Application Settings

### mtwilson.quickstart.order.dispatch.interval

The number of milliseconds to wait between checking if there are new orders to dispatch.

Default value: 200 (0.2 seconds)

Related requirements: performance

### mtwilson.quickstart.order.update.interval

The number of milliseconds to wait between processing updates to order status.

Default value: 200 (0.2 seconds)

Related requirements: performance

### mtwilson.quickstart.order.monitor.interval

The number of milliseconds to wait between monitoring execution of orders.

Default value: 200 (0.2 seconds)

Related requirements: performance

### mtwilson.quickstart.order.cleanup.interval

The number of milliseconds to wait between attempting to cleanup old orders.

Default value: 600000 (10 minutes)

Related requirements: performance, reliability

### mtwilson.quickstart.order.cleanup.modified

Cleanup orders that were modified more than this number of milliseconds ago.

Default value: 3600000 (60 minutes)

### mtwilson.quickstart.order.cleanup.accessed

Cleanup orders that were accessed (read or write) more than this number of milliseconds ago.

NOTE: the access time is checked only if the order was modified recently and is not qualified for cleanup based on the modified time

Default value: 86400000 (24 hours)

### mtwilson.quickstart.filetransfer.etag

Whether to check that a file has already been transferred to a remote host before transferring it again. The comparison is accomplished via SHA-256 sum of the two files.

Default value: “true” (meaning the file should be checked and skip transfer if same)

Related requirements: performance

### mtwilson.quickstart.ssh.hostkey.store

Whether to store SSH host keys. This is disabled by default. Note that the only information released to the client is the SSH host public key.

Default value: “false” (do not store SSH host keys)

Related requirements: user experience, security

### mtwilson.quickstart.ssh.password.store

Whether to store SSH passwords. This is disabled by default. Note that the passwords are NOT stored or released to the client. When this feature is enabled, the SSH password is salted and hashed and only the resulting salted hash is stored by the quickstart server. This is used to quickly compare user input to determine if it’s the right password, without having to open a connection to the remote host.

Default value: “false” (do not store SSH hashed passwords)

Related requirements: user experience, security

## General Settings

### jetty.port

The port on which Jetty listens for HTTP connections.

Default value: 80

Related requirements: interfaces

### jetty.secure.port

The port on which Jetty listens for HTTPS (TLS) connections.

Default value: 443

Related requirements: interfaces

### endpoint.url

The URL to use when generating HTTP links to the server.

Default value: http://ip6-localhost

Related requirements: interfaces