hub Release Notes

A CA UIM hub serves as the communication center for a group of robots. A hub binds robots into a logical group with the hub as the central connection point. Hubs are commonly set up based on physical constraints (such as a lab, floor, or building) or by service functions (such as development). A hub can also connect other hubs into a hierarchy of hubs.

To install or upgrade a hub, see Installing on the *CA Unified Infrastructure Management* wiki space.

Contents

Revision History

Date	Description	State	Version
May 2015	**New features**: ▪ Support OpenSSL TLS cipher suites ▪ When using TLS 1.1 or 1.2 cipher suites an alternative fallback to SSLv3 should be included. This allows for backward compatibility with older robots connecting to a new hub, or probes connecting to a robot using SSL. Example: AES128-SHA256:RC4-SHA, where AES128-SHA256 is TLS v1. 2 and RC4-SHA is SSLv3.0 ▪ OpenSSL 1.0.1m implemented ▪ Windows IA64 and RHEL/CentOS-5 are no longer supported ▪ **User tags are automatically copied from robot.cfg to hub.cfg when upgrading to hub 7.80.** If user tags do not exist in the hub.cfg file and if the *os _user_retrieve_from_robot* option is true (default is true), then the user tags will be copied from the robot.cfg file to the hub.cfg file when the hub restarts. After the user tags have been copied the *os_user_retrieve_from_rob ot* option is set to false in the hub.cfg file. This allows a user to intentionally clear the user tags in the hub.cfg file at a later date. ▪ **Removed ability to set SIDs with pu command.** In past releases, the **-S** option of the **pu** command could be used to explicitly set a *session identification* (SID). This capability has been removed to prevent security from being bypassed through SID injection. ▪ **Output character limit extended in pu executable**. Previously, in the pu executable prior to version 7.80, field output from callbacks would be trimmed at around 35 characters. This often made long output strings unusable. The output limit has been extended to 300 characters.	Beta	v7.80

Date	Description	State	Version
April 2015	Hub v7.71 fixes an issue seen with hub v7.70 in assigning ports for tunnel client connections. Prior to v7.70, the tunnel client connections would consistently use the 48xxx port range (based on the controller's default **first_probe_port** setting of 48000). An issue in hub v7.70 caused the tunnel client connections to use a system-assigned port number, which would not reliably fall in the 48xxx range. This caused issues with firewalls where the tunnel ports were explicitly allowed and expected to be in the specific range. With hub v7.71, the default port range for tunnel client connections again falls in the 48xxx range. As in previous versions, the specific ports for tunnel connections can be overridden by enabling **Ignore Controller First Probe Port** (which enables the hub to use its own setting) and by specifying the desired port setting **tunnel/ignore_first_probe_port = 1** and **tunnel/first_tunnel_port = <*port number*>** in the hub configuration file, *hub.cfg:* ▪ In the Admin Console hub configuration GUI, navigate to **Advanced > Tunnel Settings**. Under **Tunnel Advanced Settings**, enable **Ignore Controller First Probe Port**. Specify the desired **First Tunnel Port**. ▪ In the Infrastructure Manager hub configuration GUI, navigate to **Tunnels**. Enable **Ignore first probe port setting from controller**, and specify the desired **First Tunnel Port**.	GA	v7.71
March 2015	**Important:** CA recommends that you connect hubs with tunnels (see Best Practices for Hub-to-Hub Communication). If any hubs are *not* connected by tunnels, communication will be affected between hubs set to **mode 0** (no encryption or use of the OpenSSL transport layer) and those set to **mode 2** (SSL encryption). See Impact of Hub SSL Mode When Upgrading Non-tunneled Hubs for details. **New Features:** ▪ SuSE10 and SLES11 are no longer supported. ▪ **Hubs allow robots to retain the origin of their designated parent hub when failing over to a secondary hub.** The origin is attached to each QoS message generated by a probe and routed through its robot: ▪ By default, the origin is the name of the robot's designated parent hub, and is attached to messages by the hub's spooler. ▪ The default origin used upon hub failover has changed. With hub v7.63 and earlier, when a robot with no specified origin failed over to another hub, the origin became the name of the failover hub. As of hub v7.70, the origin remains the name of the robot's designated parent (specified by the **hub** attribute in robot.cfg). **Note:** UIM administrators can override the default value by defining the origin in the robot configuration. In multi-tenant environments, for example, an admin can specify the origin in order to group data and control user access to the data. If the origin attribute exists in robot.cfg, the robot's spooler attaches it to the message, and the hub's spooler does not alter it. This behavior has not changed.		v7.70

Date	Description	State	Version

- **User tags are propagated by the hub and controller.** User tags are now propagated in alarms and messages sent by both the hub and the controller (version 7.70). Previously, both the hub and controller read user tags *os_user 1* and *os_user2* from robot.cfg. Now, the hub reads user tags from hub.cfg. The **General Configuration** section in the Admin Console hub configuration GUI allows users to specify *os_user1* and *os_user2*. On a hub system, the hub's spooler (which is integrated with the hub probe) adds these values to probe alarms and messages generated by probes. On a robot system, the robot's spooler adds the tags.

  Note: The *os_user_include* option, which enabled the hub to read user tags from robot.cfg, was removed from the hub v7.70. After upgrading hubs and robots to v7.70, the hub will no longer read user tags from robot.cfg. If the hub's robot had user tags defined, they will remain in robot.cfg after upgrade but will be ignored. If you want user tags added to alarms and messages generated by probes running on a hub system, you must specify the user tags in hub.cfg.
- **Hub v7.70 can be configured to send an alarm for dropped messages.** All messages generated by probes have a subject that is used to route the message on the message bus. If the hub does not have an attach or post queue configured for a particular subject, any message with that subject is dropped by the spooler. Hub 7.70 has the ability to send an alarm when this occurs. This behavior is disabled by default. To enable it, specify in hub.cfg:

  `subjects_no_destination_alarm_interval=<seconds>`
- **Consistent enforcement of username and password limitations.** Username must be 1 to 255 characters long and cannot contain control characters, right or left arrows (< or >), or forward slashes (/); password must be 5 to 254 characters long.
- **Improved behavior of SSL mode**. If a controller is managed by a hub that has SSL mode=0, the controller no longer creates the robot.pem file upon startup. Any UIM components installed on that system will not accept SSL connections.
- **Improved DNS lookup during tunnel setup**. DNS lookup, which maps the hostname to an IP address, will retry on DNS failures, making tunnels more tolerant of temporary or intermittent failures.

Defects in hub were fixed to ensure that:

Date	Description	State	Version
	▪ General stability and robustness were improved. ▪ If an attempt is made to save changes to hub configuration but the file system is full, the original configuration file is retained. ▪ Hub support for LDAP over SSL has been improved. ▪ Files and directories under **hub/q** are properly cleaned up when queues are removed. ▪ The **tunnelsetup** command properly imports tunnel client certificates. ▪ The controller does not assign probes to ports used by hub tunnels, and ports are assigned to the correct interfaces based on strict binding mode. ▪ Robots configured for proxy_mode (which allows the controller to act as a proxy so that all callbacks to the probes go through the controller port) return to their designated parent hub after failover.		

December 2014	▪ OpenSSL 1.0.0m implemented.
	▪ Improved reliability of LDAP functionality when an SSL connection to the LDAP service is required.
	▪ Improved stability for long queue names. *Salesforce case 00145363*.
	▪ Package included in UIM Server 8.1 distribution.
	▪ *Fixed defects:* The following rarely-seen issues have been addressed:
	  ▪ Circular message queues occasionally caused memory resource leaks. Salesforce case 00147673.
	  ▪ A file descriptor resource leak could occur if baseline_engine and prediction_engine were deployed to a hub.
	  ▪ A potential crash condition caused by invoking the tunnel_get_info callback has been removed.
	  ▪ A potential crash condition at robot shutdown has been removed
			v7.63

August 2014	▪ Improved tunnel stability. ▪ Package included in UIM Server 8.0 distribution.		v7.61
June 2014	▪ Improved socket management between two hubs connected by a tunnel. ▪ Long-running callbacks over a tunnel connection cause fewer communication errors. ▪ Queue status alarming has been refined to reduce false positives. ▪ LDAP directories with large numbers of groups are handled more efficiently when paired with UMP 7.6. ▪ Package included in NMS 7.6 distribution.		v7.60

Date	Description	State	Version

March 2014	▪ Tunnel and queue connect and disconnect alarms have been retroactively reset to **Information** level to better match their actual meaning and impact.
	▪ The hub detects when the total resources in use threatens tunnel and/or hub viability, and reacts by either resetting the tunnel or restarting the process, without data loss.
	▪ The origin value for probes local to a hub can be set independently from that of the hub.
	▪ Enhanced LDAP and user level security and improved support of LDAP environments with large numbers of groups.
	▪ Improved tunnel stability.
	▪ *Fixed Defects:*
	  ▪ Fixed a defect that caused a core dump on Solaris when *hubup* response contents were malformed
	  ▪ Fixed a defect that led to a significant number of sockets being temporarily left in the *CLOSE_WAIT* state for hubs that communicate with a large number of child robots.
	  ▪ Fixed defects that caused duplicate tunnel connections between the same client and server, and that caused permanent tunnel disconnection due to the exhaustion of session handling threads.
			v7.50

January 2014	▪ The method for determining the status of get-attach queues that carry low volumes of data has been improved. ▪ Tunnel server TCP sockets are more efficient, and a problem that could cause the number of sockets to grow unbounded has been addressed. ▪ *Fixed Defect:* ▪ Addressed issues seen in high-volume deployments when hub was used as an LDAP proxy hub or was used with the snmpcollector probe.		v7.11

Best Practices

While most hubs perform their tasks sufficiently with little or no interaction from the administrator, you can modify various configuration settings for better performance and usability.

- **Hub-to-hub communication**. CA recommends the following:
  - Use tunnels as a means to keep communication connectivity between hubs whenever and wherever possible.
  - Use SSL mode 1 whenever possible for the simplest configuration. This enables encrypted communication between all components wherever possible, and allows components that do not support SSL (such as some legacy probes) to function.

    Note: In the past, CA cautioned that using SSL significantly reduces traffic bandwidth and performance. Because of improvements in technology, the overhead of SSL is no longer deemed significant and is often worth the increase in security.
- **Tunnels**
  - Caching the SSL sessions can significantly speed up the server/client connection time.
  - If a non-functioning tunnel will significantly impact your operations, increase the level of alarm sent if a connection is lost or cannot be made.
- **Queues.** If the size of a get or post queue never shrinks to zero or if it always has many messages, increase the **Bulk Size** on the queue. This allows the hub to transfer multiple messages in one packet.

Known Issues

- The PPM probe (which provides functionality for the Admin Console probe configuration GUIs) does not run on AIX hubs. To configure robots and probes on, or under AIX hubs, use the Raw Configure utility in Admin Console, or use Infrastructure Manager.

- If Unix robot communication fails, ensure that the /etc/hosts file on any system hosting a UIM robot, hub, server, or UMP instance contains a valid entry for the local system. This must be a fully qualified hostname and IP address. If only loopback is defined (for example, localhost 127.0.0.1), then the controller probe on that system will be unaware of its own IP address, resulting in network communication failure.

- ADE robot distribution to Windows targets sometimes fails to activate the hdb and spooler probes. To resolve this issue, go to the affected system and do a **validate security** on the affected probes (hdb and spooler)

- In hub v7.1 through v7.63, when a hub fails, robots connected to that hub do not retain the origin of their parent hub when they connect to their failover hub.

- You cannot use Unified Service Manager (USM) to automatically deploy robots to AIX or zLinux systems that were placed in your inventory through automated discovery. Use one of the following methods:

  - Run the native installer manually or with a third-party tool. See Deploy Robots in Bulk with a Third-Party Tool and Native Installers.

  - Use the Automated Deployment Engine (ADE) probe with an XML file. See Deploy Robots with an XML File and the ADE Probe.

  - Import an XML file in USM. See Deploy Robots with an XML File in USM.

- In high-volume deployments, hub v7.10 may have issues when used as an LDAP proxy hub or when used with the SNMPCollector probe. These problems have been addressed in hub version 7.11 and later.

- Inactive queues potentially impact hub 7.10 when it is used as a tunnel server because the number of resources used for Windows file handles and Unix file descriptors may grow steadily over time. The growth rate increases greatly when the tunnel server is servicing one or more get queues that carry little or no data, and therefore reset regularly. As the number of resources in use becomes large, the hub may automatically restart and return to normal operation. No data loss is expected during these restarts, and the system should automatically return to normal operation. These problems have been addressed in hub v7.11 and later.

Impact of Hub SSL Mode When Upgrading Non-tunneled Hubs

The 7.70 release of the UIM hub and robot have improved the robustness of SSL communication. The hub's SSL mode specifies the communication mode used by components managed by the hub. It is primarily used for robot-to-hub communication. However, when hubs are **not** connected with tunnels, hub-to-hub communication is also controlled by each hub's SSL mode.

Prior to this release, in a non-tunneled domain, hubs that were configured to communicate unencrypted were able to decode and respond to encrypted messages. In a multi-hub domain, upgrading to v7.70 will break this type of communication.

This section explains why this could happen and how to address it.

Note: Any tunnels set up between hubs will remain after you upgrade, and communication will continue. *CA strongly recommends that you connect all UIM hubs with tunnels to help ensure communication integrity.*

SSL Communication Mode

UIM hubs have three communication mode options:

- **SSL mode 0** — No encryption. The OpenSSL transport layer is not used.
- **SSL mode 1** — Compatibility mode, which enables the hub and robot to communicate either without encryption or with OpenSSL encryption.
- **SSL mode 2** — OpenSSL encryption only.

When the hub is set to **mode 1**, its managed components first attempt to communicate via SSL. If a request is not acknowledged, the component sends its requests to the target unencrypted.

SSL communication is enabled through the *<Nimsoft>/robot/robot.pem* certificate, which is created by the controller upon startup. This file contains the key that decodes encrypted UIM messages.

Changes to SSL Communication in Controller v7.70

SSL communication modes are more meaningful with the release of controller 7.70 because of changes to the treatment of the robot.pem certificates.

Note: The following information about controller v7.63 also applies to prior versions.

- Controller v7.63 always creates robot.pem and always acknowledges receipt of encrypted communication, regardless of the parent hub's mode. The result:
  - The first SSL encrypted request from a v7.63 controller in mode 1 to a v7.63 hub in mode 0 would succeed, because the hub would use the robot.pem file to decode the message.
  - An unencrypted request from the hub to the controller would also succeed because the compatibility controller would accept it.

This means components that are configured to communicate unencrypted and without using the OpenSSLtransport layer are able to decode encrypted messages and encrypt their response.

- Controller v7.70 creates robot.pem only when the hub's communication mode is **1** or **2**. The result:
  - An encrypted request from a v7.70 controller in mode 1 to a v7.70 hub in mode 0 would fail, because the hub cannot decode the message. The controller would then send requests unencrypted.
  - An unencrypted request from the hub to the controller would succeed.

Controller v7.70 honors the meaning of no encryption and of SSL only by ensuring that all communication for mode 0 components is unencrypted, and by ensuring that messages sent by mode 2 (SSL only) components are not decoded by mode 0 components.

Communication Issues Between v7.70 and v7.63 (and prior)

If the parent hub is set to mode=0 (unencrypyted), controller v7.70:

- Does not create robot.pem
- Deletes robot.pem if it exists

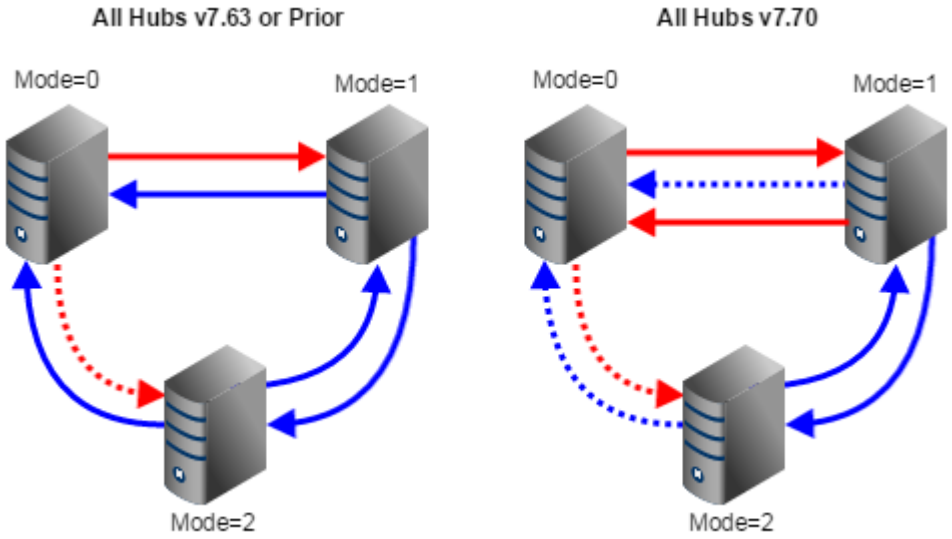
When hubs that have been upgraded to v7.70 communicate with previous versions, the modes have this effect when both hubs are set to the same mode:

- Hubs set to **mode 0** communicate unencrypted.
- Hubs set to **mode 1** use SSL encryption.
- Hubs set to **mode 2** also use SSL encryption.

The following diagram illustrates communication when all hubs are ≤v7.63 and when all hubs are v7.70. In the diagram:

- Blue lines indicate SSL communication; red lines are unencrypted communication.
- Solid lines indicate successful communication; dashed lines are unacknowledged.

- Arrow direction indicates initiator/receiver relationship. Note that although a v7.63 hub in mode 0 cannot initiate communication with a mode 2 hub, two-way communication is enabled once the relationship is established.



The behavior when the modes are mixed is as follows.

- **Version 7.63 and prior:**
  - All encoded messages sent by hubs in mode 1 and 2 are decoded.
  - Non-encoded messages sent by hubs in mode 0 are acknowledged by hubs in mode 1, and discarded unread by hubs in mode 2.
- **As of version 7.70:**
  - Encoded messages sent by a hub in mode 2 to a hub in mode 1 are decoded.
  - All requests between hubs in mode 0 and mode 2 are not acknowledged and are discarded.
  - Encrypted requests from a mode 1 hub to a mode 0 hub are ignored. Unencrypted requests between the hubs are acknowledged.

If any part of a domain based on pre-7.70 hubs and robots relies on communication between hubs in mode 2 and hubs in mode 0, you will need to resolve this after upgrading the components.