VOSS Insights Platform Guide

Release 25.2

August 27, 2025

Legal Information

DOCUMENT ID: 20250827114400

Contents

1. Introduction to Insights platform functionality

This guide describes the Insights platform commands and functionality.

Refer to the Automate Platform Guide for details on the general commands and tasks that are available via the Command Line Interface (CLI).

Note: For the Insights platform, the following functionality and related commands as found in the *Automate Platform Guide do not apply*:

- Clustering and cluster-related settings, commands and output
- Backup functionality and commands. Insights configuration data is backed up from the web interface
- Self-service commands and functions
- CLI user management

---

2. CLI reporter commands

The Insights CLI has a number of `reporter` commands

```
$ reporter
USAGE:
------
reporter connect                    - Connect to remote VOSS system
reporter http_connection            - Print http_connection for VOSS system
reporter http_connection <IP>|None - Set or remove manual http_connection for
                                        VOSS system
reporter test_connection            - Test remote mongo and VOSS system
                                        connection
```

The following commands are used during install and for system management:

- `reporter connect`
- `reporter test_connection`

reporter connect

The `reporter connect` command takes an IP address parameter, which is the target system primary database server address. To determine the VOSS Automate system primary database server IP address, log in to the VOSS Automate system and run the command, `database primary`. Use this IP address as a *host* parameter and enter the password, for example:

::

$ reporter connect host: 192.77.248.122 pass:

reporter http_connection

The Insights system web interface connects to a VOSS Automate node that runs its web server.

The web proxy nodes on a target Automate system can be shown with the `cluster status` command. For example, for a standalone system, the command output of this command is *None*, since the web proxy nodes has the same address as the application and database. The example output of the command below is on a standalone Automate system:

::

$ cluster status

Data Centre: atlantic
application : voss2-08[192.77.248.122]

webproxy : voss2-08[192.77.248.122]

database : voss2-08[192.77.248.122]

reporter http_connection <IP>|None

The web interface of VOSS Insights can be set to a specified web proxy IP address, fully qualified domain name (FQDN), or it can be reset to *None*.

reporter test_connection

The `reporter test_connection` command is a test command that is also used during the install process.

The example shows command output when connected to a standalone Automate system:

```
$ reporter test_connection
MongoDB Connection established to 192.77.248.122
Primary Connection is 192.77.248.122:27020
HTTP Connection to 192.77.248.122 successful
```

3. Application status

The `app status` command is used to display the status of the system. When the command is executed, it requests an up-to-date status of every process, and therefore may take a few seconds to return.

Note: The `uc-reporter` service status indicates the Insights service state.

A typical `app status` screen from the CLI:

```
platform@analytics123:~$ app status
cluster v1.5.0 (2016-09-07 08:02)
template_runner v1.5.0 (2016-09-07 08:13)
mongodb v1.5.0 (2016-09-07 08:02)
   |-arbiter   running
   |-database  running
support v1.5.0 (2016-09-07 08:13)
snmp v1.5.0 (2016-09-07 08:13)
   |-daemon    running (completed)
   |-traps     running (completed)
platform v1.5.0 (2016-09-07 08:03)
nginx v1.5.0 (2016-09-07 08:03)
   |-proxy     running
uc-reporter v1.3.0 (2016-09-07 07:39)
   |-node      running
services v1.5.0 (2016-09-07 08:11)
   |-wsgi      running
   |-logs      running
   |-firewall  running
   |-mount     running
   |-scheduler running
   |-syslog    running (completed)
   |-time      running (completed)
security v1.5.0 (2016-09-07 08:10)
```

The table describes the states that are defined:

`running`	The process is running correctly.
`completed`	The process completed successfully.
`suspended`	The process is suspended while waiting for another process.
`stopped`	The process is not running. An error message indicates that the process stopped for an unexpected reason.

4. Reporter logs

Insights logs that should be noted, are:

- `process/uc-reporter.node.log`

  Insights log

  Use the commands:

  - `log view process/uc-reporter.node.log` to view the log.
  - `log follow process/uc-reporter.node.log` to follow the log.
- `install/uc-reporter_install.script-YYYMMDD.log`

  Installation logs

Refer to the Platform Guide for details on sending logs to a remote destination.

5. Override default SSHD keys for CUCM

5.1. Overview

Customers with older networking systems and who are using Arbitrator for CUCM collection may wish to override the Insights system *sshd_config* default entries with their own cipher values to allow the KexAlgorithms required for legacy systems.

Legacy algorithms are disabled by default on Insights, which retains only the latest and most secure version of ssh. Older ssh keys have been found to have known flaws.

Some legacy systems (particularly Cisco UCMs) that interact with Insights may be unable to upgrade their sshd version. As a result, the legacy system may lose the ability to communicate with Insights.

Warning: It is recommended that if you choose to override the default values that ship with the system, you must verify, in a separate ssh connection (before ending your current ssh session), that you're still able to use ssh to access the system. If the file is corrupted as a result of performing this procedure, your access to ssh (and therefore your access to the system) may be compromised.

Do not perform this procedure unless you understand the security implications for your system. If you're unsure, please contact VOSS Support before making this change.

5.2. Modify the SSHD_CONFIG file

1. On the Insights system where you want to override values, for example, Arbitrator, Dashboard, or DS9, use `ssh` to log in as admin to the Insights **Administration** configuration screen:

   `ssh` and your admin user account, for example, `ssh admin@123`
2. Select **Network Configuration**.
3. Select **SSHD Config**.
4. On the **Current Customer Overrides** screen, copy and paste the keys for the relevant algorithms (the ones you wish to use). For example, you may wish to add one or more of the following KexAlgorithms:

   Important: None of the examples provided here are supported by or recommended by VOSS. This procedure only provides an alternative for legacy CUCMs.

   - ecdh-sha2-nistp521
   - ecdh-sha2-nistp384

- diffie-hellman-group14-sha1
- diffie-hellman-group1-sha1
- diffie-hellman-group-exchange-sha256
- diffie-hellman-group-exchange-sha1

You can copy these keys into the screen in a comma separated list (without spaces), as in the following example, which uses two of these algorithms:

```
KexAlgorithms ecdh-sha2-nistp521,ecdh-sha2-nistp384
```

For a older CUCMs (e.g. CUCM 11.5.1), add the following:

```
KexAlgorithms diffie-hellman-group1-sha1,diffie-hellman-group14-sha1,diffie-hellman-
↪group-exchange-sha1
MACs hmac-md5,hmac-sha1,hmac-sha2-256,hmac-sha1-96,hmac-md5-96
HostKeyAlgorithms ssh-rsa,ssh-dss
```

5. Click **OK**.
6. Verify that your changes are accepted.

Note: If you've introduced errors in the copy/paste operation, a system error displays and reverts the change. If you see an error message warning that ssh is unstable, you may need to contact VOSS Support for assistance, or re-paste the keys into the **Current Customer Overrides** screen and attempt the update again.

---

7. Before disconnecting from your current session, open a new ssh session to verify that you can still connect.

6. Elevated access for debugging

The Insights modules (Dashboard, Arbitrator, and DS9) do not allow direct root access over ssh.

If root access is required for debugging purposes, you can use the **NRS** tool (available from the **Administration** menu).

Selecting the menu and enabling the tool generates a key that can only be deciphered by VOSS. VOSS requires the key to gain root access for debugging purposes.

7. Dashboard and Arbitrator transaction logging and audit

7.1. Overview

The Insights platform provides transaction logging as an audit trail for both the Dashboard (Reporter) and for Arbitrator. This allows you to inspect the logs to investigate actions taken on these modules in the event of a data breach or for troubleshooting.

Insights records the following event types:

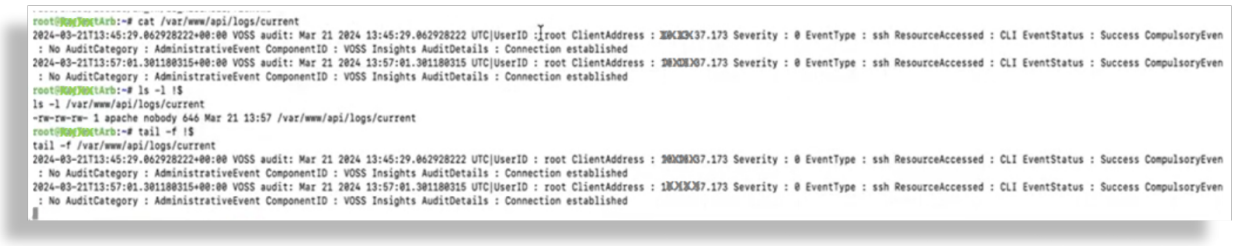
- All logins - including root, CLI, Web, admin ssh, sysadmin
- Logout
- Failed login attempts
- Password changes - including details for which password was changed, for example, admin, ftpuser, or Dropbox
- All user account changes - add, update, and delete
- Export of reports from Dashboard
- Dashboard views, updates, or deletes - including widgets on dashboards
- NRS connections (run as root) - connection established and connection closed

Related topics

- *Elevated access for debugging*

7.2. Transaction logs

Transaction logs for audited events are stored in the following file: */var/www/api/logs/current*



File Format

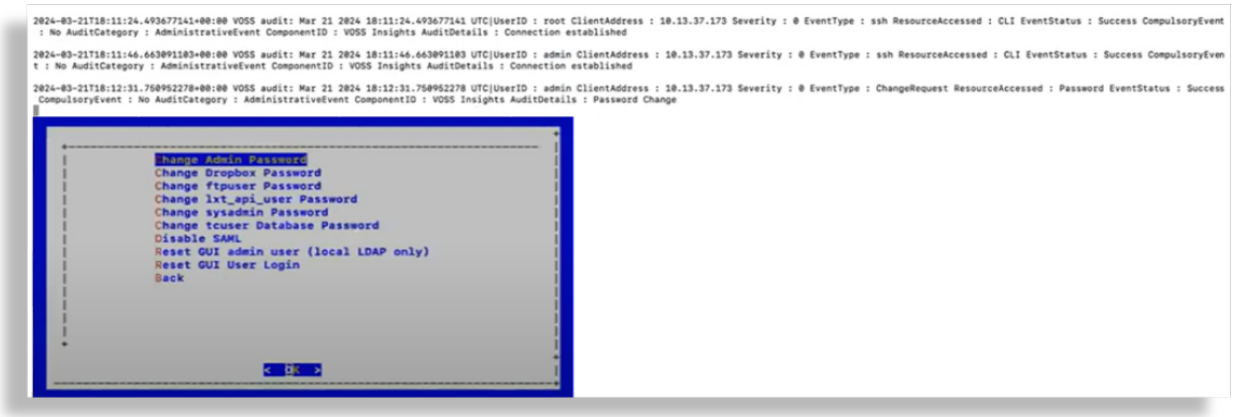
Fields in the file, such as `UserID` (for example, *root* or *admin*), `Severity`, and `EventType`, are separated by space, colon, space, that is, `` : ``

Event types

Event types logged may include, for example, `ssh` (log in event), or `ResourceAccessed` (AccessEvent or ReconnectEvent). The event type (`EventType`) and event value, for example, *AccessEvent*, depends on the action taken in the system.

Note: The transaction logging also records a reconnect event (*ReconnectEvent*) when you're switching tabs or when opening Arbitrator's System Configuration module.

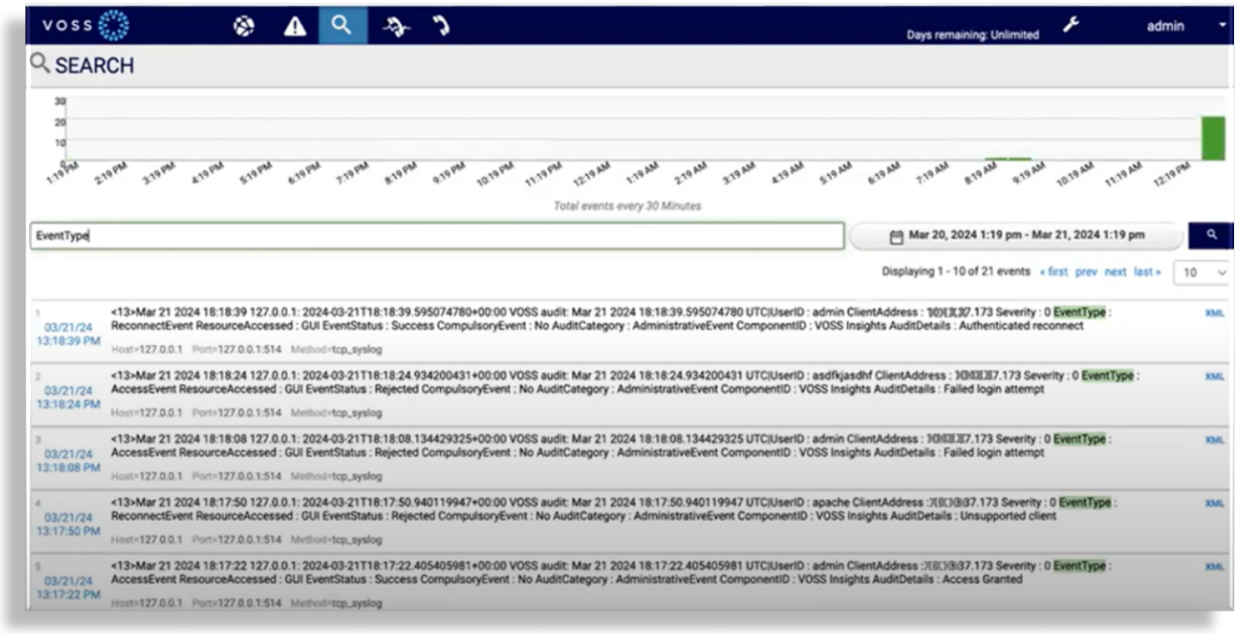
The image displays an example of a log entry showing an admin user log in and password change:



7.3. View audit event logs via the GUI

You can search for and view events through the CLI, either all events, or search for a specific audit event using the `ndx_client` command.

You can also view the audit event logs via the GUI syslogs. For example, using the field `EventType` returns all audit events as this field appears in all audit event logs. The output of this search can be redirected to a different location.



Related topics

- Search the Logs in the Dashboard Administration Guide

7.4. Dashboard event audits

Transaction and audit logging for the Dashboard system records log entries each time you view, edit and save, or delete a dashboard or widget.

Log entries are also recorded when you generate, download, or export reports from the Dashboard.

Dashboard log entries include details such as the user role and username, the date and time of the event, the dashboard or widget name, ID, and directory path, and the user role and username of the relevant user.