



## Installing Cisco VIM Insight

---

Cisco VIM 2.2 offers a unified management solution which will be available in the subsequent releases.

Cisco VIM Insight can be installed on two modes:

- Standalone/non-HA mode on a dedicated node to manage multiple VIM pods.
- Standalone/non-HA mode on the management node to manage a single VIM pods.

You can start installing in a Standalone/non-HA mode initially (on the management node of the pod) or a standalone (BOM) server. If VIM Insight is hosted on the node where the VIM management service of a pod is running, ensure that the workspace for Insight is different from that of the installer. Rendition and migration from one install mode to another is easy because the UI interacts to each pod through REST API and very little RBAC information of the admin and user is kept in the database. As the UI interacts with the REST API, it is not necessary that the pod should be managed by Insight from day 0. You can register a pod, with an Insight instance after it is up and running.

Also, the UI has two types of Admin: UI Admin and Pod Admin. UI Admin is for the administrators who can add more folks as UI Admin or Pod admin. Pod Admin has privileges only at the pod level, where as an UI Admin has privileges both at UI and pod level.

Complete the following procedure to install Cisco VIM Insight on the Cisco NFVI management node.

- [Cisco VIM Insight with Internet Access, page 1](#)
- [Installing Cisco VIM Insight without Internet Access , page 6](#)
- [VIM Insight UI Admin Login for Standalone Setup, page 13](#)
- [VIM Insight Pod Admin Login for Standalone Setup, page 13](#)

## Cisco VIM Insight with Internet Access

Complete the following steps to install Cisco VIM Insight on the Cisco NFVI management node. As security is paramount to pod management, the web-service hosting the single pane of glass is protected through TLS. Following are the steps to get the TLS certificate setup going.

You can select one of the following approaches for the TLS certificate configurations:

- 1 Provide your own certificate: You can bring in your certificate on the management node and provide the absolute path of .pem and CA certificate files in the insight\_setup\_data.yaml file. The path must be provided as a value for the key 'PEM\_PATH' in the insight\_setup\_data.yaml file.
- 2 Generate a new certificate on the node: You can create a new certificate on the node by running the following command: `#!/tls_insight_cert_gen.py -f <path_to_insight_setup_data.yaml>/insight_setup_data.yaml` This script will search for the 'PEM\_PATH' inside the insight\_setup\_data.yaml. As the path is not provided it will create a new certificate inside `install-dir/openstack-configs`.

### Before You Begin

You must complete all Cisco NFVI preparation tasks described in [Preparing for Cisco NFVI Installation](#), and also the management node has to be installed as described [Cisco VIM Management Node Networking](#). The procedure to bootstrap the node hosting the Insight is same as installing the `buildnode.iso`.

### Step 1

Enter `ip a` to verify the `br_mgmt` and `br_api` interfaces are up and are bound to `bond1` and `bond0` respectively. For example:

```
$ ip a
br_api: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP link/ether 00:42:68:6f:79:f2
brd ff:ff:ff:ff:ff:ff
inet nnn.nnn.nnn.nnn/25 brd nnn.nnn.nnn.nnn scope global br_api valid_lft forever preferred_lft
forever
inet6 fe80::3c67:7aff:fef9:6035/64 scope link valid_lft forever preferred_lft forever
bond1: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 1500 qdisc noqueue master br_api state UP
link/ether 00:42:68:6f:79:f2 brd ff:ff:ff:ff:ff:ff
br_mgmt: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP link/ether 00:78:88:46:ee:6e
brd ff:ff:ff:ff:ff:ff
inet nnn.nnn.nnn.nnn/24 brd nnn.nnn.nnn.nnn scope global br_mgmt valid_lft forever preferred_lft
forever
inet6 fe80::278:88ff:fe46:ee6e/64 scope link valid_lft forever preferred_lft forever
bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 1500 qdisc noqueue master br_mgmt state UP
link/ether 00:78:88:46:ee:6e brd ff:ff:ff:ff:ff:ff
```

**Note** The `br_mgmt` and `br_api` interfaces are created when you install RHEL on the management node in [Installing the Management Node](#).

### Step 2

Run the following commands to copy the installer directory and the standalone `insight_setup_data.yaml`.

- a) Copy the installer dir to a another directory in `/root/`. The name of the new directory should start with `Insight-tag_id`.
 

```
# cd /root/
# cp -pr installer-<tag_id> <Insight-tag_id>
```
- b) Copy the Standalone `insight_setup_data.yaml`. Standalone\_EXAMPLE file from the `Insight-dir/openstack-configs` to any other location on the management node or the BOM.
 

```
# cp /root/Insight-<tag_id>/openstack-configs/insight_setup_data.yaml.
Standalone_EXAMPLE /root/insight_setup_data.yaml
```

### Step 3

Modify the insight setup data according to your requirements.

```
#Configuration File:

#####
# User Defined Configuration File.
# Information in this file is specific to the user setup.
```

```

#####

# This file is used as an inventory file to setup Insight Container.

#####

# Registry credentials

#####
REGISTRY_USERNAME: '<username>'
REGISTRY_PASSWORD: '<password>'

# Install Mode: connected/disconnected, Optional parameter; default is connected
INSTALL_MODE: connected

# https_proxy: <Name of the proxy server without https://> ; Optional Parameter for INSTALL_MODE
# Needed for connected install only and not required for disconnected mode.

#####
# Super Admin Username Password
#####

# This User will be the default Super Admin of the System and can Grant Access to all other Users
getting registered to PODs.
# This is a mandatory field and is required to be filled every time.
UI_ADMIN_USERNAME: '<username>'
UI_ADMIN_EMAIL_ID: '<email_id@domain.com>'

# Please define the mail server off which the Insight email alias works;
# For example, outbound.cisco.com
# Mandatory: Valid SMTP Server is required for sending mails to the customers.
INSIGHT_SMTP_SERVER: <smtp.domain.com>
#INSIGHT_SMTP_PORT: <port no.>
#optional, defaults to 25, if undefined

# for Insight UI, customer needs to create a mailer, so that automated mails come from that alias;
# For example, vim-insight-admin@cisco.com
# Mandatory: You need to create a valid email alias that would be responsible for sending email
notification for Users and UI Admin.
INSIGHT_EMAIL_ALIAS: <Email-Alias@domain.com>
# Optional: Insight Email Alias Password is required if log in on a SMTP server requires
authentication.
INSIGHT_EMAIL_ALIAS_PASSWORD: <password> #Optional

#TLS certificate path;
#Absolute TLS certificate path, can also be generated using the script tls_insight_cert_gen.py located
at
# installer-<tagid>/insight/; if generated by: tls_insight_cert_gen.py, then entry of the info is
optional;
# the script will copy the certs to installer-<tagid>/openstack-configs/ dir
PEM_PATH: <abs_location_for_cert_path>

#If using tls_insight_cert_gen.py to create the cert, please define the following:
CERT_IP_ADDR: <br_api of the insight node> # Mandatory

```

```
CERT_HOSTNAME: <Domain name for Cert>      # Optional
```

And then execute:

```
# cd installer-<tagid>/insight
# ./tls_insight_cert_gen.py --file <absolute path of insight_setup_data.yaml>
```

The script will generate the certs at installer-<tagid>/openstack-configs/ dir

If bringing in a 3rd part Cert, skip the above step and define the following

```
CERT_IP_ADDR: <br_api of the insight node> # Mandatory
```

```
CERT_HOSTNAME: <Domain name for Cert> # Optional
```

```
PEM_PATH in insight_setup_data.yaml, and go to step 4 instead of executing # ./tls_insight_cert_gen.py
```

As part of insight bootstrap the script will copy the certs to installer-<tagid>/openstack-configs/ dir

**Step 4** Save the edited insight\_setup\_data.yaml file.

**Step 5** Start the insight installation process.

```
$ cd /root/Insight-<tag_id>/insight/
$ ./bootstrap_insight.py --help
usage: bootstrap_insight.py [-h] --action ACTION
                             [--regenerate_secrets] [--setpassword]
                             [--file INSIGHTSETUPDATA] [--keep] [--verbose]
                             [--backupdir BACKUPDIR] [-y]
```

Insight install setup helper.

optional arguments:

```
-h, --help            show this help message and exit
--action ACTION, -a ACTION
                        install - Install Insight UI
                        install-status - Display Insight Install Status
                        reconfigure - reconfigure - Reconfigure Insight DB password,
                        TLS Certificate, INSIGHT_SMTP_SERVER,
                        INSIGHT_EMAIL_ALIAS_PASSWORD,
                        INSIGHT_EMAIL_ALIAS, INSIGHT_SMTP_PORT
                        update - Update Insight UI
                        update-status - Display Insight Update Status
                        rollback - Rollback Insight UI update
                        commit - Commit Insight UI update
                        backup - Backup Insight UI
                        uninstall - Uninstall Insight UI
--regenerate_secrets, -r
                        System generated INSIGHT_DB_PASSWORD
--setpassword, -s     User supplied INSIGHT_DB_PASSWORD,
--file INSIGHTSETUPDATA, -f INSIGHTSETUPDATA
                        Location of insight_setup_data.yaml
--keep, -k            Preserve Insight artifacts during uninstall
--verbose, -v         Verbose on/off
--backupdir BACKUPDIR, -b BACKUPDIR
                        Path to backup Insight
-y, --yes             Option to skip reconfigure or uninstall steps without prompt
```

```
$ ./bootstrap_insight.py -a install -f </root/insight_setup_data.yaml>
```

VIM Insight install logs are at: /var/log/insight/bootstrap\_insight/bootstrap\_insight\_<date>\_<time>.log

Management Node Validations!

```
+-----+-----+-----+
| Rule | Status | Error |
+-----+-----+-----+
| Check Kernel Version | PASS | None |
| Check Ansible Version | PASS | None |
| Check Docker Version | PASS | None |
| Check Management Node Tag | PASS | None |
| Check Bond Intf. Settings | PASS | None |
| Root Password Check | PASS | None |
| Check Boot Partition Settings | PASS | None |
| Check LV Swap Settings | PASS | None |
| Check Docker Pool Settings | PASS | None |
| Check Home Dir Partition | PASS | None |
| Check Root Dir Partition | PASS | None |
| Check /var Partition | PASS | None |
| Check LVM partition | PASS | None |
| Check RHEL Pkgs Install State | PASS | None |
+-----+-----+-----+
```

Insight standalone Input Validations!

```
+-----+-----+-----+
| Rule | Status | Error |
+-----+-----+-----+
| Insight standalone Schema Validation | PASS | None |
| Valid Key Check in Insight Setup Data | PASS | None |
| Duplicate Key Check In Insight Setup Data | PASS | None |
| Test Email Server for Insight | PASS | None |
+-----+-----+-----+
```

Downloading VIM Insight Artifacts, will take time!!!

Cisco VIM Insight Installed successfully!

```
+-----+-----+-----+
| Description | Status | Details |
+-----+-----+-----+
| VIM Insight UI URL | PASS | https://<br_api:9000> |
| VIM UI Admin Email ID | PASS | Check for info @: <abs path of insight_setup_data.yaml> |
| | | |
| VIM UI Admin Password | PASS | Check for info @ /opt/cisco/insight/secrets.yaml |
| VIM Insight Workspace | PASS | /root/Insight-<tag_id>/insight/ |
+-----+-----+-----+
```

Cisco VIM Insight backup Info!

```
+-----+-----+-----+
| Description | Status | Details |
+-----+-----+-----+
| Insight backup Status | PASS | Backup done @ |
+-----+-----+-----+
```



```
br_mgmt: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP link/ether 00:78:88:46:ee:6e
  brd ff:ff:ff:ff:ff:ff
inet nnn.nnn.nnn.nnn/24 brd nnn.nnn.nnn.nnn scope global br_mgmt valid_lft forever preferred_lft
forever
inet6 fe80::278:88ff:fe46:ee6e/64 scope link valid_lft forever preferred_lft forever
bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 1500 qdisc noqueue master br_mgmt state UP
link/ether 00:78:88:46:ee:6e brd ff:ff:ff:ff:ff:ff
```

**Note** The `br_mgmt` and `br_api` interfaces are created when you install RHEL on the management node in [Installing the Management Node](#).

**Step 2** Run the following commands to copy the installer directory and the standalone `insight_setup_data.yaml`.

a) Copy the installer dir to a another directory in `/root/`. The name of the new directory should start with `Insight-tag_id`.

```
# cd /root/
# cp -pr installer-<tag_id>Insight-<tag_id>
```

b) Copy the Standalone `insight_setup_data.yaml.Standalone_EXAMPLE` file from the `Insight-dir/openstack-configs` to any other location on the management node or the BOM.

```
# cp /root/Insight-<tag_id>/openstack-configs/insight_setup_data.yaml.Standalone_EXAMPLE
/root/insight_setup_data.yaml
```

**Step 3** Modify the insight setup data according to the requirements. Refer to the `insight_setup_data.yaml` and cert generation as listed in step 5 of the preceding section.

**Step 4** Save the edited `insight_setup_data.yaml` file.

**Step 5** Run Import Artifacts:

```
$ cd /root/insight-<tag_id>/tools
./import_artifacts.sh
```

This verifies that `/var/cisco/artifacts` on the management node has the following Insight artifacts, along with the other components 'insight-K9.tar', 'mariadb-app-K9.tar'.

**Step 6** Start the insight installation process.

```
$ cd /root/Insight-<tag_id>/insight/
$ ./bootstrap_insight.py --help
usage: bootstrap_insight.py [-h] --action ACTION
```

```
        [--regenerate_secrets] [--setpassword]
        [--file INSIGHTSETUPDATA] [--keep] [--verbose]
        [--backupdir BACKUPDIR] [-y]
```

Insight install setup helper.

optional arguments:

```
-h, --help            show this help message and exit
--action ACTION, -a ACTION
install - Install Insight UI
install-status - Display Insight Install Status
reconfigure - reconfigure - Reconfigure Insight DB password,
TLS Certificate, INSIGHT_SMTP_SERVER,
INSIGHT_EMAIL_ALIAS_PASSWORD,
INSIGHT_EMAIL_ALIAS, INSIGHT_SMTP_PORT
update - Update Insight UI
update-status - Display Insight Update Status
rollback - Rollback Insight UI update
```

```

        commit - Commit Insight UI update
        backup - Backup Insight UI
        uninstall - Uninstall Insight UI
--regenerate_secrets, -r
        System generated INSIGHT_DB_PASSWORD
--setpassword, -s
        User supplied INSIGHT_DB_PASSWORD,
--file INSIGHTSETUPDATA, -f INSIGHTSETUPDATA
        Location of insight_setup_data.yaml
--keep, -k
        Preserve Insight artifacts during uninstall
--verbose, -v
        Verbose on/off
--backupdir BACKUPDIR, -b BACKUPDIR
        Path to backup Insight
-y, --yes
        Option to skip reconfigure or uninstall steps without prompt

```

```
$ ./bootstrap_insight.py -a install -f </root/insight_setup_data.yaml>
```

VIM Insight install logs are at: /var/log/insight/bootstrap\_insight/bootstrap\_insight\_<date>\_<time>.log

Management Node Validations!

```

+-----+-----+-----+
| Rule | Status | Error |
+-----+-----+-----+
| Check Kernel Version | PASS | None |
| Check Ansible Version | PASS | None |
| Check Docker Version | PASS | None |
| Check Management Node Tag | PASS | None |
| Check Bond Intf. Settings | PASS | None |
| Root Password Check | PASS | None |
| Check Boot Partition Settings | PASS | None |
| Check LV Swap Settings | PASS | None |
| Check Docker Pool Settings | PASS | None |
| Check Home Dir Partition | PASS | None |
| Check Root Dir Partition | PASS | None |
| Check /var Partition | PASS | None |
| Check LVM partition | PASS | None |
| Check RHEL Pkgs Install State | PASS | None |
+-----+-----+-----+

```

Insight standalone Input Validations!

```

+-----+-----+-----+
| Rule | Status | Error |
+-----+-----+-----+
| Insight standalone Schema Validation | PASS | None |
| Valid Key Check in Insight Setup Data | PASS | None |
| Duplicate Key Check In Insight Setup Data | PASS | None |
| Test Email Server for Insight | PASS | None |
+-----+-----+-----+

```

Setting up Insight, Kindly wait!!!

Cisco VIM Insight Installed successfully!

```

+-----+-----+-----+
| Description | Status | Details |
+-----+-----+-----+

```



```

| VIM Insight UI URL      | PASS  | https://<br_api:9000> |
| VIM UI Admin Email ID | PASS  | Check for info @: <abs path of insight_setup_data.yaml> |
|                         |       |                         |
| VIM UI Admin Password | PASS  | Check for info @ /opt/cisco/insight/secrets.yaml |
| VIM Insight Workspace | PASS  | /root/Insight_<tag_id>/insight/ |
+-----+-----+-----+

```

Cisco VIM Insight backup Info!

```

+-----+-----+-----+
| Description          | Status | Details |
|-----+-----+-----+
| Insight backup Status | PASS  | Backup done @
|                         |       | /var/cisco/insight_backup/insight_backup_<release_tag>_<date_time>|
+-----+-----+-----+

```

Done with VIM Insight install!

VIM Insight install logs are at: /var/log/insight/bootstrap\_insight/

Logs of Insight Bootstrap will be generated at : /var/log/insight/bootstrap\_insight/ on the management node. Log file name for Insight Bootstrap will be in the following format :

bootstrap\_insight\_<date>\_<time>.log. Only ten bootstrap Insight log files are displayed at a time. Once the bootstrap process is completed a summary table preceding provides the information of the UI URL and the corresponding login credentials. After first login, for security reasons, we recommend you to change the Password.

Insight autobackup takes place after an install and is located at default backup location

/var/cisco/insight\_backup;

details of which is provided in the backup summary table.

To add a new UI Admin in a setup that just got created, login to VIM insight and add a new UI admin user from the Manage UI Admin Users menu. Without doing a fresh install (that is un-bootstrap, followed by bootstrap) of the insight application, the UI admin that was bootstrapped with cannot be changed.

Refer Cisco VIM Insight Post Bootstrap Validation Checks section to verify the bootstrap status of Cisco VIM Insight.

## Cisco VIM Insight Post Bootstrap Validation Checks

- 1 After the VIM Insight bootstrap, you can view the status of Insight installation through install-status action using bootstrap.

```

$ cd /root/installer-<tag_id>/insight/
$ ./bootstrap_insight.py -a install-status

```

```

Cisco VIM Insight Install Status!
+-----+-----+-----+
| Description          | Status | Details |
+-----+-----+-----+
| VIM Insight Version  | PASS  | <release_tag> |
| VIM Insight UI URL   | PASS  | https://<br_api:9000> |
| VIM Insight Container | PASS  | insight_<tag_id> |
| VIM Mariadb Container | PASS  | mariadb_<tag_id> |
+-----+-----+-----+

```



```

        <!--new dashboard css starts-->
        <link rel="stylesheet" href="../static/css/dashboard.css">
        <!--new dashboard css end-->
    </head>
    <body class="skin-blue sidebar-collapse" ng-controller="DashboardCtrl"
id="ToggleNavbar">
        <div class="wrapper" id="wrapper">

            <div class="content-wrapper" id="contentclass">
                <mi-header></mi-header>
                <mi-left-side-navbar></mi-left-side-navbar>
                <message-box> </message-box>
                <div class=" viewheight" ng-view autoscroll="true"></div>
            </div>

            <mi-footer></mi-footer>
        </div>
        <!--new dashboard js starts-->
        <script src="../static/lib/bootstrap/jquery.min.js"></script>
        <script src="../static/lib/jquery/jquery-ui.js"></script>
        <script src="../static/lib/bootstrap/progressbar.js"></script>
        <!--new dashboard js ends-->
        <script src="../static/lib/chart/Chart.min.js"></script>
        <script src="../static/lib/bootstrap/bootstrap.min.js"></script>
        <script src="../static/lib/angular/angular.js"></script>
        <script src="../static/lib/chart/angular-chart.min.js"></script>
        <script src="../static/lib/uigrad/angular-touch.js"></script>
        <script src="../static/lib/uigrad/angular-animate.js"></script>
        <script src="../static/lib/uigrad/csv.js"></script>
        <script src="../static/lib/uigrad/pdfmake.js"></script>
        <script src="../static/lib/uigrad/vfs_fonts.js"></script>
        <script src="../static/lib/uigrad/ui-grid.js"></script>
        <script src="../static/lib/angular/smart-table.min.js"></script>
        <script src="../static/lib/angular-route/angular-route.js"></script>
        <script src="../static/lib/angular-cookies/angular-cookies.js"></script>
        <script src="../static/lib/angular/angular-translate.js"></script>
        <script
src="../static/lib/angular/angular-translate-loader-static-files.min.js"></script>
        <script
src="../static/lib/angular/angular-translate-storage-cookie.min.js"></script>
        <script
src="../static/lib/angular/angular-translate-storage-local.min.js"></script>
        <script src="../static/lib/yamltojson/yaml.js"></script>
        <script src="../static/lib/yaml/js-yaml.min.js"></script>
        <script src="../static/lib/d3/d3min.js"></script>
        <script src="../static/utility/utility.js"></script>
        <script src="../static/widgets/widgets.js"></script>
        <script src="../static/app.js"></script>
        <script src="../static/layout/layout.js"></script>
        <script src="../static/login/login.js"></script>
        <script src="../static/globals/globals.js"></script>
        <script src="../static/dashboard/dashboard.js"></script>
        <script src="../static/cloudpulse/cloudpulse.js"></script>
        <script src="../static/blueprintsetup/physicalsetupwizard/ucsmcommon.js"></script>

        <script src="../static/blueprintsetup/physicalsetupwizard/cimcommon.js"></script>

        <script src="../static/vmtp/runvmtp.js"></script>

        <script src="../static/blueprintsetup/physicalsetupwizard/networking.js"></script>

        <script
src="../static/blueprintsetup/physicalsetupwizard/serverandroles.js"></script>
        <script src="../static/blueprintsetup/openstacksetupwizard/cephsetup.js"></script>

        <script
src="../static/blueprintsetup/openstacksetupwizard/cindersetup.js"></script>
        <script
src="../static/blueprintsetup/openstacksetupwizard/glancesetup.js"></script>
        <script src="../static/blueprintsetup/openstacksetupwizard/haproxy.js"></script>

```

```

    <script
src="../static/blueprintsetup/openstacksetupwizard/keystonesetup.js"></script>
    <script src="../static/blueprintsetup/openstacksetupwizard/swiftstack.js"></script>

    <script
src="../static/blueprintsetup/openstacksetupwizard/neutronsetup.js"></script>
    <script src="../static/blueprintsetup/openstacksetupwizard/vmtpsetup.js"></script>

    <script
src="../static/blueprintsetup/physicalsetupwizard/physicalsetupwizard.js"></script>
    <script src="../static/blueprintsetup/servicesSetupWizard/systemlog.js"></script>

    <script src="../static/blueprintsetup/servicesSetupWizard/nfvbench.js"></script>

    <script
src="../static/blueprintsetup/servicesSetupWizard/servicesSetupWizard.js"></script>
    <script
src="../static/blueprintsetup/openstacksetupwizard/openstacksetupwizard.js"></script>
    <script src="../static/blueprintsetup/blueprintsetup.js"></script>
    <script src="../static/blueprintmanagement/blueprintmanagement.js"></script>
    <script src="../static/topology/topology.js"></script>
    <script src="../static/monitoring/monitoring.js"></script>
    <script src="../static/horizon/horizon.js"></script>
    <script src="../static/podmanagement/podmanagement.js"></script>
    <script src="../static/blueprintsetup/openstacksetupwizard/tlssupport.js"></script>

    <script src="../static/blueprintsetup/openstacksetupwizard/elksetup.js"></script>

    <script src="../static/systemupdate/systemupdate.js"></script>
    <script
src="../static/blueprintsetup/physicalsetupwizard/registrysetup.js"></script>
    <script src="../static/registerertestbed/registerertestbed.js"></script>
    <script src="../static/registeresaas/registeresaas.js"></script>
    <script src="../static/useradministration/manageusers.js"></script>
    <script src="../static/useradministration/rolemanagement.js"></script>
    <script src="../static/saasadmindashboard/saasadmindashboard.js"></script>
    <script src="../static/saasadmindashboard/buildnodes.js"></script>
    <script src="../static/saasadmindashboard/buildnodeusers.js"></script>
    <script src="../static/saasadmindashboard/managesaasuser.js"></script>
    <script src="../static/saasadminusermanagement/saasadminusermgmt.js"></script>
    <script src="../static/blueprintsetup/physicalsetupwizard/nfvsetup.js"></script>

    <script src="../static/blueprintsetup/physicalsetupwizard/torswitch.js"></script>

    <script src="../static/blueprintsetup/openstacksetupwizard/vtssetup.js"></script>

    <script src="../static/rbacutilities/rbacutility.js"></script>
    <script src="../static/forgotpassword/forgotpassword.js"></script>
    <script src="../static/changepassword/changepassword.js"></script>
    <script src="../static/passwordreconfigure/passwordreconfigure.js"></script>
    <script
src="../static/openstackconfigureconfigure/openstackconfigureconfigure.js"></script>
    <script
src="../static/reconfigureoptionalservices/reconfigureoptionalservices.js"></script>
</body>

```

- 5 VIM Insight Autobackup: Insight will invoke Insight Autobackup as a daemon process. Autobackup is taken as an incremental backups of database and /opt/cisco/insight/mgmt\_certs dir if there is any change.

You can check the status of Insight Autobackup service:

```

systemctl status insight-autobackup
insight-autobackup.service - Insight Autobackup Service
   Loaded: loaded (/usr/lib/systemd/system/insight-autobackup.service; enabled; vendor
   preset: disabled)
   Active: active (running) since Mon 2017-09-04 05:53:22 PDT; 19h ago
   Process: 21246 ExecStop=/bin/kill ${MAINPID} (code=exited, status=0/SUCCESS)
   Main PID: 21287 (python)
   Memory: 9.2M
   CGroup: /system.slice/insight-autobackup.service
           └─21287 /usr/bin/python
   /var/cisco/insight_backup/insight_backup_2.1.10_2017-08-31_03:02:06/root

```

```
/rohan/installer-10416/insight/playbooks/./insight_autobackup.py
Sep 04 05:53:22 F23-insight-4 systemd[1]: Started Insight Autobackup Service.
Sep 04 05:53:22 F23-insight-4 systemd[1]: Starting Insight Autobackup Service...
```

## VIM Insight UI Admin Login for Standalone Setup

For security reasons, the UI Admin need to login to the UI with which Insight has been bootstrapped, and add other users as UI Admin. UI Admin needs to add new users as Pod Admin.

### Registration of UI Admin to Insight

- 
- Step 1** Enter the following address on the browser: `https://<br_api>:9000`.
  - Step 2** Enter the Email ID and the Password. Email ID should be the one specified as 'UI\_ADMIN\_EMAIL\_ID' in `insight_setup_data.yaml` during bootstrap. Password for UI Admin is generated at `:/opt/cisco/insight/secrets.yaml` and key is 'UI\_ADMIN\_PASSWORD'
  - Step 3** Click **Login as UI Admin User**. You will be redirected to Insight UI Admin Dashboard.
- 

## VIM Insight Pod Admin Login for Standalone Setup

### Registration of Pod Admin to Insight:

- 
- Step 1** Login as Insight UI Admin.
  - Step 2** Navigate to **Manage Pod Admin**.
  - Step 3** Click **Add Pod Admin**.
  - Step 4** Enter a new Email ID in **Add Pod Admin** pop-up.
  - Step 5** Enter the username of the Pod Admin.
  - Step 6** Click **Save**. User Registration Mail would be sent to newly added Pod Admin with a token.
  - Step 7** Click the URL with token and if token is valid then Pod Admin will be redirected to Insight-Update Password page.
  - Step 8** Enter new password and then confirm the same password.
  - Step 9** Click **Submit**.
-

