

OpenStack and OpenAttestation Integration User Guide

V 0.1

Contents

OpenStack and OpenAttestation Integration User Guide	1
Introduction	3
What is OpenAttestation?	3
Architecture	3
Installation.....	4
Hardware environment:	4
Software and Packages:.....	5
Environment Setup.....	5

Introduction

The goal of the project is to provide VM level attestation in OpenStack. It leverages the projects including OpenStack, OpenAttestation and Intel TXT. The project is under POC stage and the success criteria is to create instance on trusted host within OpenStack environment. Following sections show the design of the architecture and the environment setup of this POC project.

What is OpenAttestation?

OpenAttestation project is to provide SDK, Software Development Kit, to add cloud management tools with capability of establishing hosts integrity information by remotely retrieving and verifying Hosts' integrity with TPM quote.

For more details, please refer to the following links.

<https://01.org/zh/openattestation?langredirect=1>

<https://github.com/OpenAttestation/OpenAttestation/wiki>

Architecture

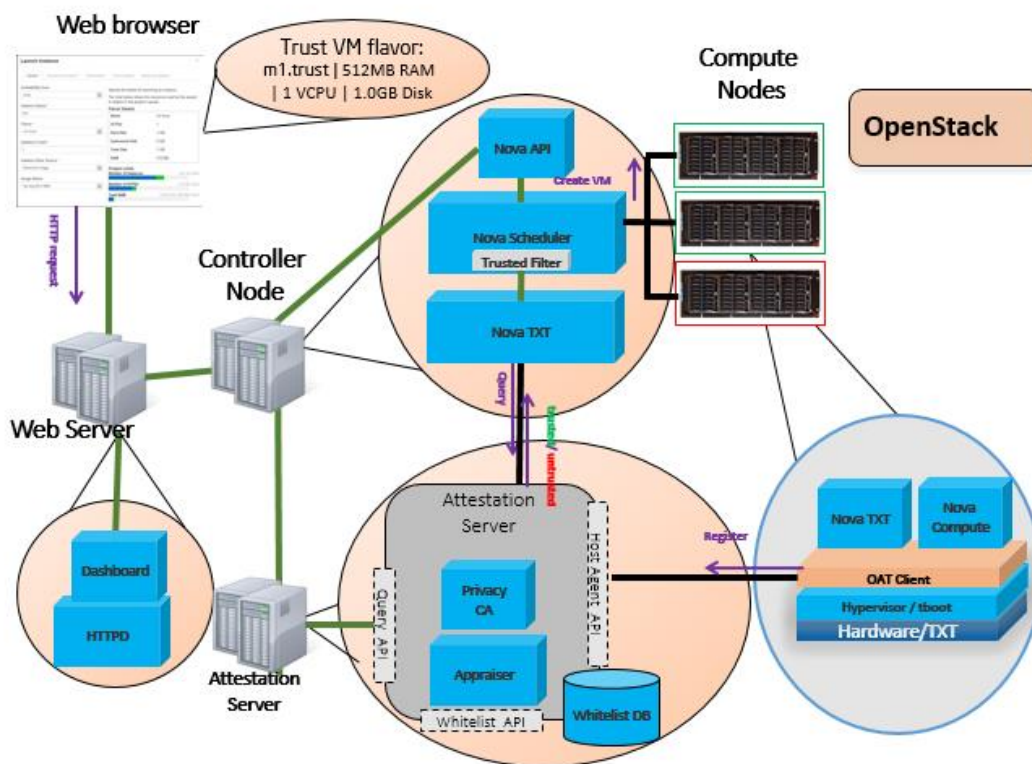


Figure 1 Architecture and Data Flow

Figure 1 illustrates the design of the project and the data flow between OpenStack and OpenAttestation.

OpenAttestation could remotely retrieve and verify hosts' integrity with TPM quote. It restores the PCR registers of physical machine in server side. Accordingly, the machines registers its PCR values to server by OpenAttestation client. The PCR registers store a unique hash code of BIOS or kernel. By verify the consistency of these values, the OpenAttestation server could tell if a machine is trusted or not.

OpenAttestation provides restful APIs to the administrator for the status querying, which could be used in OpenStack. As shown in figure1, Nova-scheduler could get the trust information of computing nodes. When request to create trusted instance come, it will get the trusted nodes list by TrustFilter and launch VMs on them. At the same time, a new agent Nova-TXT is been created to query host trust information periodically.

On the OpenStack Dashboard, a new status bar is added to show the trusted status of compute node and a new page is added to display the PCR values of the hosts.

Note:

In the latest OpenStack upstream, TrustFilter has been included. But it has performance issue. It will take minutes to create a trusted instance because it needs to wait for several seconds and get the results from OpenAttestation server side. To resolve it, a new agent, Nova-TXT, is created and query the trust information periodically.

Installation

This section will show how to set up the environment of the POC.

Hardware environment:

Five nodes are used and shown in the table.

Hostname	CPU	Memory	Disk	Ethernet	Role	TPM
Node-1	Intel Core i5-2400 3.10GHz	4G	250G	1	Fuel master	Without TPM
Node-2	Intel Core i7-4790 3.60GHz	16G	500G	2	Controller node	With TPM and enabled
Node-3	Intel Core i7-4790 3.60GHz	16G	500G	2	Compute node	With TPM and enabled
Node-4	Intel Core i7-4790 3.60GHz	16G	500G	2	Compute/Storage node	With TPM and disabled
Node-5	Intel Core Quad CPU Q9650 3.00GHz	4G	250G	1	OpenAttestation server	Without TPM

Software and Packages:

In this POC, Fuel 6.0 is used to set up Juno OpenStack and the OS is Ubuntu 12.04. The neutron is enabled with GRE mode. Ubuntu 14.04 is be used to setup OpenAttestation server.

Environment Setup

Prepare:

Enable Intel® TXT in BIOS before installation OpenStack. Client system must have TPM 1.2 compliant device with driver installed. Below is an example for HP8300 system:

- 1) Power on, ESC key -> Startup Menu -> Computer Setup(F10)
- 2) Security->Setup Password, set setup password as xxxxxx then F10 save it.
- 3) Security->System Security, enable vtx/vtd/Embedded Security Device/Trusted Execution Technology, F10 save it.
- 4) File->Save Changes and Exit

Note:

Please clear the TPM in BIOS if it has been enabled before.

OpenStack installation:

Three nodes are used to set up the OpenStack environment. One controller, one compute node and the third node is used for both storage node and compute node. Please follow the standard procedure to set up OpenStack using Fuel.

OpenAttestation installation:

For now, the latest version of OpenAttestation is 2.2. But it doesn't support packages installation on Ubuntu, which means user have to build from source code.

Please refer to the OAT wiki about how to compile and install server/client of OpenAttestation.

<https://github.com/OpenAttestation/OpenAttestation/wiki/OAT-v2.2-Ubuntu14.04.1>

Manual Configuration:

After the OpenAttestation installation done, the next step is to configure OpenStack. Please make sure you clone the repo of patches.

```
patches_repo/
├── 0001-OpenAttestation-intergration-with-Horizon-v1.patch
├── 0001-OpenAttestation-intergration-with-Nova-v1.patch
├── create_soft_link.sh
├── create_trust_flavor.sh
├── nova-txt
├── nova-txt.conf
├── oat-register-node.sh
└── restart-nova.sh
```

- 1) OpenAttestation client register

Run the command:

```
$ ./oat-register-node.sh --oatserver=$OAT_SERVER_IP --myip=$HOST_IP
```

You need to run this script on every node you want to register on server side.

If succeed, you can login the web portal of OpenAttestation server via this link, [https://\\$OAT_SERVER_IP:8080/TrustDashBoard/home.html](https://$OAT_SERVER_IP:8080/TrustDashBoard/home.html). It shows the information of registered nodes.

REFERENCE CLOUD PORTAL										
Home Host Management Reports										
Trust Status Dash Board										
Host Name	Location	BIOS Trust Status	VMM Trust Status	Overall Trust Status	Updated On	Trust Status	Trust Assertion	Trust Report	Status	
+ node-45	Ubuntu K V M	✓	✓	✓	Tue Sep 8 22:09:17 CST 2015	Refresh				
+ node-44	Ubuntu K V M	✓	✓	✓	Tue Sep 8 22:09:20 CST 2015	Refresh				

2) Configure Controller Node.

```
$ cd /usr/share/pyshared
$ patch -p1 < 0001-OpenAttestation-intergration-with-Nova-v1.patch
$ ./create_soft_link.sh
$ cp nova-txt /usr/bin
$ cp nova-txt.conf /etc/nova
```

Modify the nova-txt.conf with your OAT_SERVER_IP/CONTROLLER_NODE_NAME and restart Nova services.

```
$ ./restart-nova.sh controller
```

Modify the OpenStack Dashboard by running following command.

```
$ cd /usr/share/openstack-dashboard
$ patch -p1 < 0001-OpenAttestation-intergration-with-Horizon-v1.patch
$ service apache2 restart
```

3) Configure Compute Node.

```
$ cd /usr/share/pyshared
$ patch -p1 < 0001-OpenAttestation-intergration-with-Nova-v1.patch
$ ./create_soft_link.sh
$ cp nova-txt /usr/bin
$ cp nova-txt.conf /etc/nova
```

Modify the nova-txt.conf with your OAT_SERVER_IP/CONTROLLER_NODE_NAME and restart Nova services.

```
$ ./restart-nova.sh compute
```

4) Create trust flavor on controller node

```
$ ./create_trust_flavor.sh
```

5) Login OpenStack DashBoard.

After login the dashboard, please check the host information in Admin->Host Aggregates and the trust flavor in Admin->Flavors.

The screenshot shows the OpenStack Admin dashboard. The left sidebar has a menu with 'Host Aggregates' highlighted. The main content area is titled 'Host Aggregates' and shows a table with columns: Name, Availability Zone, Hosts, Metadata, and Actions. Below this, there is a section for 'Availability Zones' with a table showing columns: Availability Zone Name, Hosts, and Available. The 'nova' availability zone is highlighted, and its hosts are listed as 'node-43 (Services Up, unknown)' and 'node-44 (Services Up, trusted)'. The 'node-44' entry is highlighted with a red box.

Availability Zone Name	Hosts	Available
internal	node-44 (Services Up, trusted) node-45 (Services Up, unknown)	Yes
nova	node-43 (Services Up, unknown) node-44 (Services Up, trusted)	Yes

The screenshot shows the OpenStack Admin dashboard with the 'Flavors' section selected. It displays a table of flavors with columns: Flavor Name, VCPUs, RAM, Root Disk, Ephemeral Disk, Swap Disk, ID, Public, Metadata, and Actions. The 'm1.trust' flavor is highlighted with a red box.

Flavor Name	VCPUs	RAM	Root Disk	Ephemeral Disk	Swap Disk	ID	Public	Metadata	Actions
m1.micro	1	64MB	0GB	0GB	0MB	0217036-0c15-434c-9917-68ac8211b0a5	Yes	No	Edit Flavor
m1.tiny	1	512MB	1GB	0GB	0MB	1	Yes	No	Edit Flavor
m1.small	1	2048MB	20GB	0GB	0MB	2	Yes	No	Edit Flavor
m1.medium	2	4096MB	40GB	0GB	0MB	3	Yes	No	Edit Flavor
m1.trust	2	4096MB	100GB	0GB	0MB	6	Yes	Yes	Edit Flavor
m1.large	4	8192MB	80GB	0GB	0MB	4	Yes	No	Edit Flavor
m1.xlarge	8	16384MB	160GB	0GB	0MB	5	Yes	No	Edit Flavor

6) Create trusted instance

Try to create a trusted instance by choose m1.trust flavor.

The screenshot shows the OpenStack Admin dashboard with the 'Instances' section selected. A 'Launch Instance' dialog is open, showing the configuration for creating a new instance. The 'Flavor' dropdown is set to 'm1.trust', which is highlighted with a red box. The 'Instance Name' is 'test2', and the 'Image Name' is 'TestVM (12.6 MB)'. The 'Launch' button is visible at the bottom right of the dialog.

If everything configured correctly, you will see the instance being launched on a trusted node.