

# EVE metadata service

## Motivation

In order to support network metadata we have to rely on services, installed in the VM of user. The most common tool for instance initialization is [cloud-init](#). There are set of images supporting it out-of-the-box: <https://docs.openstack.org/image-guide/obtain-images.html>. It supports several DataStores (from cloud), and as a variant of [datasource](#) we can use [OpenStack](#) one. It is [open source](#) and [documented](#).

## loud-init OpenStack DataSource requirements

In order to start communication with OpenStack DataSource, cloud-init do some [checks](#) of environment:

- Maybe OpenStack if
  - **non-x86 cpu architecture**: because DMI data is buggy on some arches
- Is OpenStack if **x86 architecture and ANY** of the following
  - **/proc/1/environ**: Nova-lxd contains `product_name=OpenStack Nova`
  - **DMI product\_name**: Either `Openstack Nova` or `OpenStack Compute`
  - **DMI chassis\_asset\_tag** is `OpenTelekomCloud`, `SAP CCloud VM`, `OpenStack Nova` (since 19.2) or `OpenStack Compute` (since 19.2)

We can set `product_name` in `smbios` for our VMs to tell cloud-init to fire communication with endpoints.

Also we should take into account, that there are an order of DataSource observation inside cloud-init. By default NoCloud (drive we use now) has priority (the order is [here](#)).

So, with both DataStores activated:

```
root@1a831fa7-c50b-4693-a16e-fb8171f1b69e:~# grep Datasource /var/log/cloud-init-output.log
Cloud-init v. 20.4-0ubuntu1~20.10.1 finished at Tue, 09 Mar 2021 07:10:44 +0000. Datasource DataSourceNoCloud [seed=/dev/sr0][dsmode=net]. Up 22.97 seconds
```

With manually removed NoCloud drive:

```
ubuntu@niceshamir:~$ grep Datasource /var/log/cloud-init-output.log
Cloud-init v. 20.4-0ubuntu1~20.10.1 finished at Tue, 09 Mar 2021 07:25:26 +0000. Datasource DataSourceOpenStack [net,ver=2]. Up 23.16 seconds
```

## loud-init OpenStack DataSource endpoints

OpenStack metadata serves several endpoints <https://docs.openstack.org/nova/latest/user/metadata.html#metadata-openstack-format>:

- `http://169.254.169.254/openstack/{version}/meta_data.json` - contains (among other fields) `public_keys`, `hostname`, `devices` (`disk`, `nic`)
- `http://169.254.169.254/openstack/{version}/network_data.json` - contains information about networks, dns service and links (which will be configured inside VM)
- `http://169.254.169.254/openstack/{version}/user_data` - contains script to run inside VM
- `http://169.254.169.254/openstack/{version}/vendor_data2.json` - data, which independent from VM deployments (we can omit it now)
- `http://169.254.169.254/openstack` - contains versions of OpenStack metadata

Those endpoints should be accessible from VM and serve separate information for different VMs.

## Cloud-init EC2 DataStore

We can also try to implement EC2-compatible datastore described here: <https://docs.openstack.org/nova/latest/user/metadata.html#ec2-compatible-metadata>. It will be called in case of image has no OpenStack datasource inside and forced to skip [check](#) (Cirros image for example).