SALT – HOWTO

Installation

MASTER
https://repo.saltproject.io/#ubuntu

sudo curl -fsSL -o /usr/share/keyrings/salt-archive-keyring.gpg
https://repo.saltproject.io/py3/ubuntu/20.04/amd64/latest/salt-archive-keyring.gpg
echo "deb [signed-by=/usr/share/keyrings/salt-archive-keyring.gpg arch=amd64]
https://repo.saltproject.io/py3/ubuntu/20.04/amd64/latest focal main" | sudo tee
/etc/apt/sources.list.d/salt.list
sudo apt-get update
sudo apt-get install salt-master -y
ss -lntp
sudo apt-get install -y --force-yes libffi-dev libssl-dev python-dev python-cffi libxslt1-dev
python3-pip

sudo pip install ncclient==0.6.9
sudo pip install napalm

MINION
https://repo.saltproject.io/#ubuntu

sudo curl -fsSL -o /usr/share/keyrings/salt-archive-keyring.gpg
https://repo.saltproject.io/py3/ubuntu/20.04/amd64/latest/salt-archive-keyring.gpg
echo "deb [signed-by=/usr/share/keyrings/salt-archive-keyring.gpg arch=amd64]
https://repo.saltproject.io/py3/ubuntu/20.04/amd64/latest focal main" | sudo tee
/etc/apt/sources.list.d/salt.list
sudo apt-get update
sudo apt-get install salt-minion -y
ss -lntp

CONFIGURATION
LOCATION
MASTER - /etc/salt/master
MINION - /etc/salt/minion

MASTER
`salt-key -F master`
Copy the `master.pub` fingerprint from the *Local Keys* section, and then set this value as the `master_finger` in the minion configuration file. Save the configuration file and then restart the Salt minion.

```
MINION
Add the host IP addresses of master node as
<IP ADDRESS> salt

Or add master: <IP ADDRESS> in minion configuration
```
sudo systemctl restart salt-minion
```
salt-call --local key.finger
```

Go back to MASTER
```
[root@master ~]# salt-key -L
Unaccepted Keys:
minion
Accepted Keys:

Either use salt-key -A or use salt-key -a minion to add individually as
this should be done for all minions.


Add Network Devices:
ios in /srv/pillar/ios.sls
proxy:
  proxytype: napalm
  driver: ios
  host: 192.168.30.232
  username: apnic
  password: apricot2022
  dest_file_system: flash


junos in /srv/pillar/junos.sls

proxy:
  proxytype: napalm
  driver: junos
  host: 192.168.30.137
  username: apnic
  password: apricot2022

Add those devices in /srv/pillar/top.sls

base:
  ios:
    - ios
    - r1_data
  junos:
    - junos
    - r2_data
  junos2:
    - junos2
    - r2_data
  '*':
    - common_data

After every change in top.sls restart salt master with

Sudo systemctl restart salt-master
```

Add those devices to salt master through proxy
sudo salt-proxy --proxyid=ios -d
sudo salt-proxy --proxyid=junos -d

Go back to MASTER

```
[root@master ~]# salt-key -L
Unaccepted Keys:
ios
junos
Accepted Keys:

Either use salt-key -A or use salt-key -a ios/junos to add individually as
this should be done for all minions.
```

Add some varibles for ios as r1_data, junos as r2_data and common in common_data
In file /srv/pillar/r1_data.sls

```
asn: 65500
router_id: "100.100.100.0"
interfaces:
 - interface_name: "Ethernet1/1"
   description: "### Peering with Customer01 ###"
   ipv4_address: "100.100.100.0"
   ipv4_netmask: "255.255.255.254"
   ipv6_address: "2001:DB8:100::"
   ipv6_netmask: 127
bgp_peers:
 - v4_peers:
   - peer_address: "100.100.100.1"
     peer_as: 63932
     peer_description: "IPv4 eBGP with Customer01"
     peer_password: "apricot2022"
 - v6_peers:
   - peer_address: "2001:DB8:100::1"
     peer_as: 63932
     peer_description: "IPv6 eBGP with Customer01"
     peer_password: "apricot2022"
```

In file /srv/pillar/r2_data.sls

```
asn: 65500
router_id: "100.100.200.0"
hostname: "R2"
interfaces:
 - interface_name: ge-0/0/0
   description: "Peering-with-Customer01"
   ipv4_address: "100.100.200.0"
   ipv4_netmask: 31
   ipv6_address: "2001:DB8:200::"
   ipv6_netmask: 127
```

```
bgp_peers:
  - v4_peers:
    - peer_address: "100.100.100.1"
      peer_as: 63932
      peer_description: "IPv4 eBGP with Customer01"
      peer_password: "apricot2022"
  - v6_peers:
    - peer_address: "2001:DB8:100::1"
      peer_as: 63932
      peer_description: "IPv6 eBGP with Customer01"
      peer_password: "apricot2022"
```

In file /srv/pillar/common_data.sls

```
ntp_servers:
- 192.168.0.250
- 192.168.0.251
dns_servers:
- 192.168.0.253
- 192.168.0.254
```

Create the template file /srv/pillar/templates/router.jinja

```
{%- set router_vendor = grains.vendor -%}
{%- if router_vendor|lower == 'juniper' %}
system {
  replace: name-server {
{%- for dns_server in pillar.dns_servers %}
  {{ dns_server }};
{%- endfor %}
  }
  replace: ntp {
{%- for ntp_server in pillar.ntp_servers %}
    server {{ ntp_server }};
{%- endfor %}
  }
}
{%- for interface in pillar.interfaces %}
interfaces {
  {{ interface.interface_name }}
  description {{ interface.description }}
  unit 0 {
    family inet {
      address {{ interface.ipv4_address }}/{{ interface.ipv4_netmask }}
    }
    family inet6 {
      address {{ interface.ipv6_address }}/{{ interface.ipv6_netmask }}
    }
```

```
 }
}
{%- endfor %}
{%- elif router_vendor|lower in ['cisco'] %}
{%- for dns_server in pillar.dns_servers %}
ip name-server {{ dns_server }}
{%- endfor %}
{%- for ntp_server in pillar.ntp_servers %}
ntp server {{ ntp_server }}
{%- endfor %}
{%- for interface in pillar.interfaces %}
interface {{ interface.interface_name }}
no shutdown
description {{ interface.description }}
ip address {{ interface.ipv4_address }} {{ interface.ipv4_netmask }}
ipv6 enable
ipv6 address {{ interface.ipv6_address }}/{{ interface.ipv6_netmask }}
{%- endfor %}
{%- endif %}
```

Refresh pillar changes:

```
Sudo salt '*' saltutil.refresh_pillar

Create a provision file /srv/salt/provision_router.sls
Install the infrastructure services config:
  netconfig.managed:
    - template_name: salt://templates/router.jinja
    - timeout: 100


Apply the changes
sudo salt '*os' state.apply provision_router

Check if the bgp is up:

sudo salt-run bgp.neighbors up=True
```