Cấu hình backup Salt-Master với Multimaster Failover.

- Cho phép salt-minion có thể thay thế salt-master khi có sự cố tại salt-master đang kết nối tới.

Yêu cầu:

- Đã triển khai mô hình salt master – minion cơ bản

- Tồn tại 2 Salt-master:

+ Salt-master chính

+ Salt-master backup

+ Salt minion được xác thực trên cả 2 master

- Master backup đã được cấu hình cơ bản.

Bắt đầu:

Tại Salt Master

Bước 1:

- Tại Salt-Master chính (Primary), Cấu hình file “/etc/salt/master”

# vim /etc/salt/master

## Thêm

master\_sign\_pubkey: True

Bước 2:

- Restart lại service Salt master sẽ sinh ra cặp key:

# ls /etc/salt/pki/master/

master\_sign.pem

master\_sign.pub

Bước 3:

- Copy cặp key này đến tất cả các master sẽ được sử dụng cho multimaster. Và thêm dòng “master\_sign\_pubkey: True” tương tự trên các master

# scp [primary salt master] remote: /etc/salt/pki/master/

Bước 4:

- restart salt master

# systemclt start salt-master

Tại Salt minion:

Bước 1:

- copy “file master\_sign/pub” tại salt master primary tới thư mục “/etc/salt/pki/minion/”

Note:

+ Không copy hay sử dụng file master\_sign.pem.

Bước 2:

- Tại file cấu hình minion, tìm hoặc thêm trường:

# vim /etc/salt/minion

###

verify\_master\_pubkey\_sign: True

Bước 3: Trước khi restart, chạy tại chế độ debug để kiểm

# salt-minion -l debug

- Nếu mọi thứ chính xác:

[DEBUG ] Attempting to authenticate with the Salt Master at 172.16.0.10

[DEBUG ] Loaded minion key: /etc/salt/pki/minion/minion.pem

[DEBUG ] salt.crypt.verify\_signature: Loading public key

[DEBUG ] salt.crypt.verify\_signature: Verifying signature

[DEBUG ] Successfully verified signature of master public key with verification public key master\_sign.pub

[INFO ] Received signed and verified master pubkey from master 172.16.0.10

[DEBUG ] Decrypting the current master AES key

- Nếu sai

[DEBUG ] Attempting to authenticate with the Salt Master at 172.16.0.10

[DEBUG ] Loaded minion key: /etc/salt/pki/minion/minion.pem

[DEBUG ] salt.crypt.verify\_signature: Loading public key

[DEBUG ] salt.crypt.verify\_signature: Verifying signature

[DEBUG ] Failed to verify signature of public key

[CRITICAL] The Salt Master server's public key did not authenticate!

- Khi lỗi xảy ra, kiểm tra lại file master\_sign.pub xem minion có trùng với tất cả master hay không.

Bước 4:

- Nếu mọi thứ chính xác, chạy server salt-minion.

# systemclt restart salt-minion

Cấu hình multimaster trên Minion

Bước 1:

- Tại file cấu hình minion tìm và sửa:

# vim /etc/salt/minion

###

master:

- ip-salt master 1 #Điền ip hoặc domain

- ip-salt master 2

- ip-salt master 3

…

master\_type: failover

…

master\_alive\_interval: <seconds> # thời gian kiểm tra lại salt-master

Bước 3:

- chạy debug giống các bước trên để kiểm tra lại kết nối giữa salt minion với các master được định nghĩa

# salt-minion -l debug (chỉ dùng khi stop service salt-minion)

Bước 4:

- Nếu mọi thứ chính xác chạy service salt-minion

# systemclt start salt-minion

Xem thêm:

<https://docs.saltstack.com/en/latest/topics/tutorials/multimaster_pki.html>

TESTING THE SETUP

At least two running masters are needed to test the failover setup.

Both masters should be running and the minion should be running on the command line in debug mode

salt-minion -l debug

The minion will connect to the first master from its master list

[DEBUG ] Attempting to authenticate with the Salt Master at 172.16.0.10

[DEBUG ] Loaded minion key: /etc/salt/pki/minion/minion.pem

[DEBUG ] salt.crypt.verify\_signature: Loading public key

[DEBUG ] salt.crypt.verify\_signature: Verifying signature

[DEBUG ] Successfully verified signature of master public key with verification public key master\_sign.pub

[INFO ] Received signed and verified master pubkey from master 172.16.0.10

[DEBUG ] Decrypting the current master AES key

A test.ping on the master the minion is currently connected to should be run to test connectivity.

If successful, that master should be turned off. A firewall-rule denying the minions packets will also do the trick.

Depending on the configured conf\_minion:master\_alive\_interval, the minion will notice the loss of the connection and log it to its logfile.

[INFO ] Connection to master 172.16.0.10 lost

[INFO ] Trying to tune in to next master from master-list

The minion will then remove the current master from the list and try connecting to the next master

[INFO ] Removing possibly failed master 172.16.0.10 from list of masters

[WARNING ] Master ip address changed from 172.16.0.10 to 172.16.0.11

[DEBUG ] Attempting to authenticate with the Salt Master at 172.16.0.11

If everything is configured correctly, the new masters public key will be verified successfully

[DEBUG ] Loaded minion key: /etc/salt/pki/minion/minion.pem

[DEBUG ] salt.crypt.verify\_signature: Loading public key

[DEBUG ] salt.crypt.verify\_signature: Verifying signature

[DEBUG ] Successfully verified signature of master public key with verification public key master\_sign.pub

the authentication with the new master is successful

[INFO ] Received signed and verified master pubkey from master 172.16.0.11

[DEBUG ] Decrypting the current master AES key

[DEBUG ] Loaded minion key: /etc/salt/pki/minion/minion.pem

[INFO ] Authentication with master successful!

and the minion can be pinged again from its new master.

e