Horcrux with Namada Blockchain for High Availability and Security

This guide is intended for Namada Validator operators. It outlines the use of Horcrux as a remote signing cluster with three Namada nodes to enhance high availability and security. This approach eliminates single points of failure and enhances validator signing key security.

Design:

- Three Horcrux servers serve as the remote signers cluster.
- Three Namada Nodes are used as validators.

Software Requirements:

- Operating System: Ubuntu 22.04.3
- Application: Horcrux v3.2.3

Hardware Requirements for Signers:

- Three Virtual Private Servers (VPS) with 2 CPUs, 2 GB RAM, and 20 GB SSD each.

Firewall Open Ports:

- Ports 19901 for Nodes and 2222 for Signers (You can choose different ports if desired).

DNS Records:

- Create three CNAME records for nodes: node1, node2, and node3.
- Create three CNAME records for signers: signer1, signer2, and signer3.

Run These Steps on All Signer Servers:

1. Create a directory to organize your Horcrux files:

mkdir

HorcruxNamada HORCRUX_PATH = (pwd) /HorcruxNamada cd

HorcruxNamada 1. Download the horcrux binary v3.2.3

wget

https://github.com/strangelove-ventures/horcrux/releases/download/v3.2.3/horcrux_linux-amd64 1. Rename horcrux_linux-amd64 to "horcrux" and copy it to /usr/bin/ and /usr/local/sbin/:

mv

horcrux_linux-amd64

horcrux sudo

cp

horcrux

/usr/bin/ sudo

cp

horcrux

/usr/local/sbin/horcrux 1. Create a horcrux service:

sudo

nano

/etc/systemd/system/hornamada.service 1. Paste the following content:

[Unit] Description =

horcrux

Signer

For

Namada After = network.target

[Service] Type = simple User = YOUR_LINUX_USER WorkingDirectory = HORCRUX_PATH

but use the string value

ExecStart

/usr/bin/horcrux

start

--home HORCRUX_PATH

but use the string value

Restart

on-failure RestartSec = 3 LimitNOFILE = 4096

[Install] WantedBy = multi-user.target 1. Enable the service using:

sudo

systemctl

enable

hornamada.service

Activating the first signer (Signer 1):

1. Copy your Namadapriv_validator_key.json
2. from thecometbft/config
3. directory (located inBASE_DIR
4.) to the newly createdHorcruxNamada
5. folder.

(Optional) This guide will be using FQDN instead of IP, port 19901 for Nodes and port 2222 for signers. Feel free to adjust based on your preferences.

1. Run the following commands to initialize the horcrux cluster:

YOURDOMAIN

"" horcrux

config

init

--node

"tcp://node1.YOURDOMAIN:19901"

--node

"tcp://node2.YOURDOMAIN:19901"

--node

"tcp://node3.YOURDOMAIN:19901"

--cosigner

"tcp://signer1.YOURDOMAIN.net:2222"

--cosigner

"tcp://signer2.YOURDOMAIN:2222"

--cosigner

"tcp://signer3.YOURDOMAIN:2222"

--threshold

2

--grpc-timeout

1500 ms

--raft-timeout

1500 ms

--home HORCRUX_PATH horcrux

create-ecies-shards

--shards

3

--home HORCRUX_PATH NAMADA_CHAIN_ID = '" horcrux

create-ed25519-shards

--chain-id NAMADA_CHAIN_ID --key-file HORCRUX_PATH /priv_validator_key.json

--threshold

2

--shards

3

--home HORCRUX_PATH The above steps will generate cosigner communication encryption keys. Expect to find new files and new folders insideHorcruxNamada :

priv_validator_key.json config.yaml cosigner_1/ecies_keys.json cosigner_2/ecies_keys.json cosigner_3/ecies_keys.json state 1. Move yourpriv_validator_key.json 2. to a secure location, as it will not be needed any more. 3. Create a new file inside the state folder named{NAMADA_CHAIN_ID}_priv_validator_state.json

This file will hold the signing state for the cluster.

1. Paste the below content into it:

{ "height" :

"0" , "round" :

"0" , "step" :

3 } 1. Copy theHorcruxNamada 2. directory to the other signers usingscp 3. .

After copying yourHorcruxNamada folder to the second signer, you will need to delete both folders named cosigner_1 and cosigner_3, along with their content from the HorcruxNamada folder inside signer 2. This will leave only two folders:cosigner_2 andstate .

1. Copycosigner_2
2. content (ecies_keys.json
3.) to theHorcruxNamada
4. Folder:

config.yaml ecies_keys.json cosigner_2 state\NAMADA-CHAIN-ID_priv_validator_state.json state 1. Repeat same steps for signer 3. Thus, theHorcruxNamada 2. folder content in the third server should be:

config.yaml ecies_keys.json cosigner_3 state\NAMADA-CHAIN-ID_priv_validator_state.json state

Apply these steps to the first Namada node

1. Configure Namada to start using the Horcrux cluster for signing blocks by editing theconfig.toml
2. located in Namada config folder.

Search for

priv_validator_laddr

"" Replace it with

priv_validator_laddr

"0.0.0.0:19901" 1. Remove the priv_validator_key.json from the node and store it in secure location as we donât need it anymore 2. Stop the Namada node. ONLY after it stopped, open the filepriv_validator_state.json 3. inside thecometbft/data 4. directory and check the âheightâ number. 5. Go to each signer and edit theNAMADA-CHAIN-ID_priv_validator_state.json 6. file inside theHorcruxNamada/state 7. directory with the âheightâ number from step 3.

It should now look like this:

{ "height" :

"" , "round" :

"0" , "step" :

3 } 1. Start your firsthorcrux 2. signer process insideSigner_One 3. and check the logs:

sudo

systemctl

restart

hornamada.service

&&

sudo

journalctl

-u

hornamada.service

-f

--output

cat 1. Start the signer 2 and signer 3horcrux signer 2. processes and watch the logs 3. Start your Namada process on your first node and check the logs.

If everything is working fine, your node should start signing blocks.

1. Install 2 Namada Nodes in different servers and edit their config file as we did with node 1

WARNING :

FOR ALL RUNNING NODES IN THE CLUSTER BE SURE YOU ARE USINGpriv_validator_laddr = "0.0.0.0:19901" AND REMOVE THE ORIGINALpriv_validator_key.json FROM ALL NODES PLEASE NOTE THAT USING REMOTE SIGNING COULD LEAD TO DOUBLE SIGNING AND SLASHING IF YOUR NODE SIGNED SAME BLOCK TWICE, SO BE SURE THAT NEVER USE LOCAL AND REMOTE SIGNING SAME TIME.

TROUBLESHOOTING :

- check FW ports
- check dns for signers and node cnames
- check files and folder paths for horcrux
- check same horcrux version on all signers
- PING RTT time between nodes and signers (more delay more issues)

[Base directory](#) [Validators](#)