Solana Validator Operations Best Practices

After you have successfully setup and started avalidator on testnet (or another cluster of your choice), you will want to become familiar with how to operate your validator on a day-to-day basis. During daily operations, you will bemonitoring your server , updating software regularly (both the Solana validator software and operating system packages), and managing your vote account and identity account.

All of these skills are critical to practice. Maximizing your validator uptime is an important part of being a good operator.

Educational Workshops

The Solana validator community holds regular educational workshops. You can watch past workshops through theSolana validator educational workshops playlist .

Help with the validator command line

From within the Solana CLI, you can execute thesolana-validator command with the--help flag to get a better understanding of the flags and sub commands available.

```
solana-validator --help
```

Restarting your validator

There are many operational reasons you may want to restart your validator. As a best practice, you should avoid a restart during a leader slot. Aleader slot is the time when your validator is expected to produce blocks. For the health of the cluster and also for your validator's ability to earn transaction fee rewards, you do not want your validator to be offline during an opportunity to produce blocks.

To see the full leader schedule for an epoch, use the following command:

solana leader-schedule Based on the current slot and the leader schedule, you can calculate open time windows where your validator is not expected to produce blocks.

Assuming you are ready to restart, you may use thesolana-validator exit command. The command exits your validator process when an appropriate idle time window is reached. Assuming that you have systemd implemented for your validator process, the validator should restart automatically after the exit. See the below help command for details:

```
solana-validator exit --help
```

Upgrading

There are many ways to upgrade theSolana CLI software . As an operator, you will need to upgrade often, so it is important to get comfortable with this process.

Note validator nodes do not need to be offline while the newest version is being downloaded or built from source. All methods below can be done before the validator process is restarted.

Building From Source

It is a best practice to always build your Solana binaries from source. If you build from source, you are certain that the code you are building has not been tampered with before the binary was created. You may also be able to optimize yoursolana-validator binary to your specific hardware.

If you build from source on the validator machine (or a machine with the same CPU), you can target your specific architecture using the-march flag. Refer to the following doc forinstructions on building from source .

solana-install

If you are not comfortable building from source, or you need to quickly install a new version to test something out, you could instead try using thesolana-install command.

Assuming you want to install Solana version1.14.17 , you would execute the following:

solana-install init 1.14.17 This command downloads the executable for1.14.17 and installs it into a.local directory. You can also look atsolana-install --help for more options.

Note this command only works if you already have the solana cli installed. If you do not have the cli installed, refer to install solana cli tools

Restart

For all install methods, the validator process will need to be restarted before the newly installed version is in use. Usesolana-validator exit to restart your validator process.

Verifying version

The best way to verify that your validator process has changed to the desired version is to grep the logs after a restart. The following grep command should show you the version that your validator restarted with:

```
grep -B1 'Starting validator with'
```

Snapshots

Validators operators who have not experienced significant downtime (multiple hours of downtime), should avoid downloading snapshots. It is important for the health of the cluster as well as your validator history to maintain the local ledger. Therefore, you should not download a new snapshot any time your validator is offline or experiences an issue. Downloading a snapshot should only be reserved for occasions when you do not have local state. Prolonged downtime or the first install of a new validator are examples of times when you may not have state locally. In other cases such as restarts for upgrades, a snapshot download should be avoided.

To avoid downloading a snapshot on restart, add the following flag to thesolana-validator command:

--no-snapshot-fetch If you use this flag with thesolana-validator command, make sure that you runsolana catchupafter your validator starts to make sure that the validator is catching up in a reasonable time. After some time (potentially a few hours), if it appears that your validator continues to fall behind, then you may have to download a new snapshot.

Downloading Snapshots

If you are starting a validator for the first time, or your validator has fallen too far behind after a restart, then you may have to download a snapshot.

To download a snapshot, you mustNOT use the--no-snapshot-fetch flag. Without the flag, your validator will automatically download a snapshot from your known validators that you specified with the--known-validator flag.

If one of the known validators is downloading slowly, you can try adding the--minimal-snapshot-download-speed flag to your validator. This flag will switch to another known validator if the initial download speed is below the threshold that you set.

Manually Downloading Snapshots

In the case that there are network troubles with one or more of your known validators, then you may have to manually download the snapshot. To manually download a snapshot from one of your known validators, first, find the IP address of the validator in using thesolana gossip command. In the example below,5D1fNXzvv5NjV1ysLjirC4WY92RNsVH18vjmcszZd8on is the pubkey of one of my known validators:

solana gossip | grep 5D1fNXzvv5NjV1ysLjirC4WY92RNsVH18vjmcszZd8on The IP address of the validators is139.178.68.207 and the open port on this validator is80 . You can see the IP address and port in the fifth column in the gossip output:

139.178.68.207 | 5D1fNXzvv5NjV1ysLjirC4WY92RNsVH18vjmcszZd8on | 8001 | 8004 | 139.178.68.207:80 | 1.10.27 | 1425680972 Now that the IP and port are known, you can download a full snapshot or an incremental snapshot:

wget --trust-server-names http://139.178.68.207:80/snapshot.tar.bz2 wget --trust-server-names http://139.178.68.207:80/incremental-snapshot.tar.bz2 Now move those files into your snapshot directory. If you have not specified a snapshot directory, then you should put the files in your ledger directory.

Once you have a local snapshot, you can restart your validator with the--no-snapshot-fetch flag.

Regularly Check Account Balances

It is important that you do not accidentally run out of funds in your identity account, as your node will stop voting. It is also important to note that this account keypair is the most vulnerable of the three keypairs in a vote account because the keypair for the identity account is stored on your validator when running thesolana-validator software. How much SOL you should store there is up to you. As a best practice, make sure to check the account regularly and refill or deduct from it as needed. To check the account balance do:

solana balance validator-keypair.json Note solana-watchtower can monitor for a minimum validator identity balance.

Seemonitoring best practices for details.

Withdrawing From The Vote Account

As a reminder, your withdrawer's keypair shouldNEVER be stored on your server. It should be stored on a hardware wallet, paper wallet, or multisig mitigates the risk of hacking and theft of funds.

To withdraw your funds from your vote account, you will need to runsolana withdraw-from-vote-account on a trusted computer. For example, on a trusted computer, you could withdraw all of the funds from your vote account (excluding the rent exempt minimum). The below example assumes you have a separate keypair to store your funds calledperson-keypair.json

solana withdraw-from-vote-account \ vote-account-keypair.json \ person-keypair.json ALL \ --authorized-withdrawer authorized-withdrawer-keypair.json To get more information on the command, usesolana withdraw-from-vote-account --help .

For a more detailed explanation of the different keypairs and other related operations refer tovote account management .

Previous Setup an RPC Node Next Best Practices: Validator Monitoring