

# Nodectl

- [Nodectl - Node Version details \(sudo nodectl -cv\)](#)
- [nodectl - Refresh Binaries \(sudo nodectl refresh\\_binaries\)](#)
- [nodectl - Install new node via command line with PK12 option](#)
- [nodectl configure -e \(Go into the node control configuration\)](#)
- [Make scripts to make your life easier!](#)
- [nodectl ipv6 disable --ni --sysctl](#)
- [nodectl -csi -p dag-l0](#)
- [Get back to Ready/Ready](#)

# Nodectl - Node Version details

## (sudo nodectl -cv)

```
nodeadmin@Constellation-Node:/scripts$ sudo nodectl -cv
```

```
[sudo] password for nodeadmin:
```

```
A new version of [nodectl] was detected..... v2.17.5
```

```
To upgrade issue: sudo nodectl upgrade_nodectl
```

```
=====
=          CHECK VERSIONS          =
=====
```

PROFILE	CLUSTER	JAR FILE
dag-l0	mainnet	cl-node.jar
TESS INSTALLED	NODECTL INSTALLED	NODECTL CONFIG
<b>v3.0.2</b>	<b>v2.17.3</b>	<b>v2.1.1</b>
TESS LATEST	NODECTL LATEST STABLE	CONFIG LATEST
<b>v3.0.2</b>	<b>v2.17.5</b>	<b>v2.1.1</b>
TESS VERSION MATCH	NODECTL VERSION MATCH	NODECTL CONFIG MATCH
True	False	True
NODECTL CODE NAME	NODECTL PRERELEASE	
Princess Warrior	False	

nodectl installed: Running on node.

nodectl latest stable: Recommended version.

nodectl latest: Newest, may be experimental and not stable.

nodectl config: nodectl's configuration version.

```
=====
=          CHECK VERSIONS          =
=====
```

PROFILE	CLUSTER	JAR FILE
dag-l1	mainnet	cl-dag-l1.jar
TESS INSTALLED	NODECTL INSTALLED	NODECTL CONFIG
<b>v3.0.2</b>	<b>v2.17.3</b>	<b>v2.1.1</b>
TESS LATEST	NODECTL LATEST STABLE	CONFIG LATEST
<b>v3.0.2</b>	<b>v2.17.5</b>	<b>v2.1.1</b>
TESS VERSION MATCH	NODECTL VERSION MATCH	NODECTL CONFIG MATCH
True	False	True
NODECTL CODE NAME	NODECTL PRERELEASE	

Princess Warrior      False

nodectl installed: Running on node.

nodectl latest stable: Recommended version.

nodectl latest: Newest, may be experimental and not stable.

nodectl config: nodectl's configuration version.

This node's restart service does not need to be restarted because pid [782965] was found already.

# nodectl - Refresh Binaries (sudo nodectl refresh\_binaries)

```
sudo nodectl refresh_binaries
```

# nodectl - Install new node via command line with PK12 option

If your P12 password is not working its due to a bug. Run the following command to do the install.

```
sudo nodectl install --quick-install --p12-migration-path /root/your.p12 --p12-passphrase 'AbcXyzab1$$'
```

IMPORTANT!! Once you run this command your passphrase will be in the history file. Run the following to clear history

```
history -c && history -w
```

# nodectl configure -e (Go into the node control configuration)

1. sudo nodectl configure -e
2. Select No unless support asks

```
=====
=  CONSTELLATION NETWORK HYPERGRAPH  =
=          NODECTL                   =
=    CONFIGURATION TOOL              =
=====
Code Name: Princess Warrior

Welcome to the nodectl configuration tool.


This feature of nodectl will help you initialize a new configuration or update/edit an existing configuration file.

nodectl will attempt to migrate/integrate your configurations changes in order to ensure a smooth transition and operations of your Node via
nodectl.

Detailed Mode will walk you through all steps/questions; with detailed explanations of each element of the configuration.
Advanced Mode will be non-verbose, with no walk through explanations, only necessary questions.

The configuration tool does only a limited amount of data type or value verification. After the configuration tool creates a new configuration or edits an
existing configuration, it will attempt to verify the end resulting configuration.

You can also choose the -d option at the command line to enter detailed mode directly.
You can also choose the -a option at the command line to enter advanced mode directly.

Switch to advanced mode? [n]: 
```

3. Press enter on this one... just an FYI.

```
=====
=                      MAIN MENU                      =
=====

Backup [cn-config.yaml] if exists..... complete
A previous cn-config.yaml was found on the system.

In the event the backup directory was not found, a backup was created in the existing directory. The location is shown below.
/var/tessellation/nodectl/backup_cn-config_2025-06-15-15:52:41Z

Press any key to continue
```

4. Select what item you want to configure. The hot ones are in red...especially Auto Restart

----- \* NODECTL EDITOR READY \* -----

nodectl configuration yaml was found, loaded, and validated.

If the configuration found on the node reports a known issue; It is recommended to go through each issue one at a time, revalidating the configuration after each edit, in order to make sure that dependent values, are cleared by each edit made.

If not found, please use the manual setup and consult the Constellation Network Doc Hub for details.

=====

=	OPTIONS MENU	=
---	--------------	---

=====

- E) Edit Individual Profile Sections
- G) Global Pl2 Configuration
- I) Internal Cluster Token Identifier
- T) Token Cluster Token Coin Id
- R) Auto Restart Configuration
- L) Set Log Level
- P) Passphrase Encryption
- D) Setup/Update Delegated Staking
- N) Setup Alerting
- M) Main Menu
- Q) Quit

KEY PRESS an option

# Make scripts to make your life easier!

This will create the folder for you to store your scripts in.

1. Open terminal
2. `sudo mkdir /scripts`
3. `chmod 770 /scripts`
4. `chown root:nodadmin /scripts`

## Create scripts

Scripts are easy to create with nano. They are just the same one line command you would usually run.

1. `cd /scripts`
2. `nano l1.sh` (This will leave and stop L1. Stop means the JAVA process will be stopped)

Your script should contain two lines (two separate commands)

```
root@meeseeks: /scripts
GNU nano 7.2
sudo nodedctl leave -p dag-l1
sudo nodedctl stop -p dag-l1
```

3. `CTRL-X, Y`

```
File Name to Write: l1.sh
G Help
C Cancel
```

4. `chmod +x l1.sh`

5. `run` to see the details

```
-rwxrwx--- 1 nodeadmin nodeadmin 57 Jun 19 14:14 l1.sh
```

`login, cd /scripts`

`./l1.sh`



```
cd /scripts  
./l1.sh ; ./l0.sh ; ./j0.sh
```

[Debug related URLs](#) | [KB-DAG](#)

### Crontab example

To run it every Sunday at, for example, 10:00 AM, add:

bash

Copy

Edit

```
0 10 * * 0 /scripts/myscript.sh
```



#### Explanation:

- 0 → minute
- 10 → hour (10 AM)
- \* → every day of the month
- \* → every month
- 0 → Sunday (can also use 7)

```
0 */3 * * * /scripts/l1.sh
```

# nodectl ipv6 disable --ni --sysctl

Disables IP6! you don't need it as its not really used for most networks. Maybe colleges mainly and the fed's

# nodectl -csi -p dag-l0

Check if you are on the seelist

# Get back to Ready/Ready

Try1:

1. run `sudo nodectl leave -p dag-l1 ; sudo nodectl stop -p dag-l1 ; sudo nodectl leave -p dag-l0 ; sudo nodectl stop -p dag-l0`
2. Shut down your VM from the cloud side. This way it really shuts down, clears out its memory file and the VM executable. Basically a power off if it were a physical machine.
3. Power back on and login
4. run `sudo nodectl execute_starchiver --datetime -p dag-l0`
5. Wait 5 minutes
6. `sudo nodectl upgrade --ni`

Explanation - We are basically stopping L1/L0 cleanly and then shutting down the node. Your nodes are processes on a Virtualization HyperVisor called KVM. If you reboot then that processes never goes away. If you power off you get a clean process and the memory file is normally recreated. After that you Starchiver it, wait 5 minutes and upgrade nodectl. Let me know how it goes as I'm trying to find that as close to perfect procedure.

## 1. Shutdown vs. Reboot Behavior

- **Shutdown:**

- When you shut down a VM, it's like powering off a physical machine. The guest OS inside the VM performs a clean shutdown of all its processes, network services, and file systems. It writes everything to disk, cleans up temporary files, and powers off gracefully.
- **The VM's resources are completely released on the host side**, and the virtual CPUs, RAM, and disk are freed up until the VM is started again.

- **Reboot:**

- When you reboot a VM, the guest OS essentially performs a restart but doesn't fully power off the virtual machine. The VM's state is still there in memory (RAM), and most resources remain allocated. After the VM reboots, it resumes from a fresh boot state.
- It's a bit faster than a full shutdown because it doesn't have to go through the entire cleanup and shutdown process. It just restarts the OS and resets the virtual hardware environment.