Zero Touch Deployment of Open Dell Switches with SONiC

April 2023

H19513

White Paper

Abstract

This white paper helps implement a zero touch deployment with a turnkey network staging experience. Digital transformations require a fast time to market with little room for error using streamlined processes with a high degree of automation.

Dell Technologies Solutions

DELL Technologies

Copyright

Contents

Introduction

Executive summary

IT workforces are under great pressure in our digital era. They must deliver their services in a short time, and high quality is expected. Infrastructure deployment and life cycle management should be fast and feasible on a large scale. Cloud application development, deployment, and change management has shifted from manual control to an Infrastructure as Code (IaC) way of working. This change demands optimizations in the underlying infrastructure components, specifically life cycle management of network switches.

Zero touch provisioning (ZTP) allows for the secure deployment of network switching infrastructure from a central management point using a fully automated workflow, eliminating human error, and reducing manual labor. The following are examples of use cases that ZTP supports:

- Turnkey infrastructure deployment and configuration without user intervention
- Upgrade your SONiC infrastructure with a simple ZTP run task
- Replace a broken switch
- Upgrade switches from another third-party NOS to SONiC
- Reinstall an unresponsive SONiC switch
- Reinstall a SONiC switch with lost configuration
- Reset a SONiC switch to clean fabric default status
- Run compliance and quality tasks before letting a switch become operational

This whitepaper guides network staff on how to set up zero touch deployment (ZTD) for a Dell switch with Software for Open Networking in the Cloud (SONiC) step by step. ZTP examples, diagnostics, and monitoring screenshots are all inclusive. After finishing this guide, you will understand ZTD on SONiC and can decide whether this is a valuable addition to your network infrastructure.

Audience

This document is intended for the following Dell-related companies:

- Customers
- Partner
- Internal

This document is intended for the following IT staff roles of the company:

- Network architects
- Network engineers
- Security engineers
- NetOps engineers
- DevOps engineers
- Solution architects

- System engineers
- IT managers
- COO
- CIO
- CISO
- CTO

Solution overview

SONiC has become a standard for open-source networking, similar to how Linux has become an operating system of choice for application serving. SONiC has all the ingredients to fit in a NetDevOps culture. Cloud native architecture, open-sourced, unmodified Debian Linux, purpose build for commodity hardware, and a broad set of API control elements give SONiC the ideal networking operating system to use as a foundation in your infrastructure. All you need is to unlock the Zero touch deployment (ZTD) capabilities of the network switch and the operating system to optimize network staging and life-cycle management.

Dell Technologies has embraced open-source networking on their network switches by delivering an enterprise-grade version of SONiC. With this initiative, customers can benefit from an open-source network operating system with enterprise-grade quality and support.

This whitepaper explains how to install a Dell network switch with the SONiC operating system by using ZTD functionality. You will learn how to get a factory default switch up and running with SONiC in a matter of minutes. You will also be able to equip your environment with SONiC-based network switches and establish an operational environment at scale with central management. Firmware installation, firmware upgrades and downgrades, and base configuration installation is all available with ZTD.

This whitepaper includes zero touch provisioning (ZTP) examples, diagnostics, and monitoring screenshots. After finishing this guide, you will feel confident in your understanding of ZTD on SONiC, and you can decide for yourself that this is a valuable addition to your network infrastructure.

Note: Zero touch deployment (ZTD) and zero touch provisioning (ZTP) are used interchangeably. Both have the same meaning.

Terminology

Acronym	Definition
DHCP	Dynamic host configuration protocol
DNS	Domain name system
HTTP	Hypertext Transfer Protocol
IP	Internet protocol
ONIE	Open network install environment
NOS	Network operating system
OS	Operating system

Acronym	Definition
SONiC	Software for Open Networking in the Cloud
URL	Uniform resource locator
ZTD	Zero touch deployment
ZTP	Zero touch provisioning

Zero Touch Deployment of SONiC

Figure 1 explains the ZTP staging flow of a Dell switch. ZTP capability is a function of the operating system involved, which in this whitepaper is SONiC. However, Dell switches are called open network switches. The switches are factory default equipped with a tiny operating system (OS) called Open Network Install Environment (ONIE). ONIE is installed on the switch and is the first OS that is booted when you turn on the switch. ONIE gives the switch the openness to install a third-party NOS. This whitepaper concentrates on installing SONiC.

Take a closer look into what happens when you power on a factory default switch which has ONIE installed. The following numbers correspond to the numbers in figure 1.

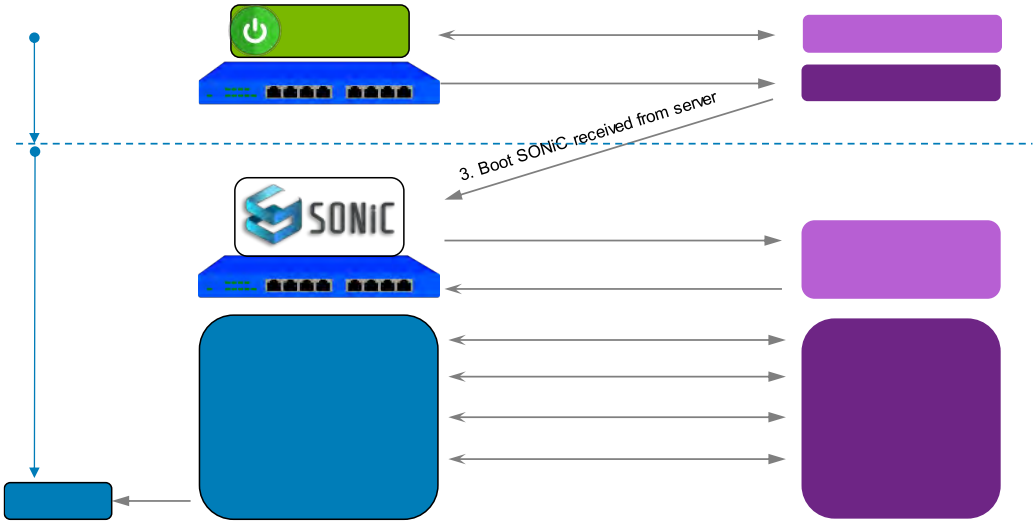

Figure 1. Zero Touch Deployment flow explained

Staging flow overview

1. ONIE loads and requests an IP address using DHCP.
2. Based on the platform details ONIE discovers, it tries to fetch a NOS based on various URL names. If a NOS image is offered by the web server, it proceeds to step 3. When all URL names ONIE tries do not find a match on the webserver, the discovery process starts over.
3. ONIE downloads the NOS (in this case SONiC) offered by the webserver.
4. The switch reboots, loads SONiC, and requests an IP address using DHCP.

5. The DHCP server offers an IP address and option 67, which contains a ZTP URL pointing to a json file for SONiC.
6. SONiC requests the ZTP json file, which contains details for subsequent optional ZTP tasks.
7. If firmware install is defined in the json file, ONIE requests an upgrade/downgrade of the SONiC firmware image, install, and reboot.
8. If config-db is defined in the json file, ONIE requests the SONiC configfile and configures SONiC.
9. If provisioning-script is defined in json file, ONIE requests and runs the script.
10. When the provisioning is completed, the ZTP flow is also completed. The switch is now operational with SONiC.

Note: When ZTP has finished, the ZTP mode is disabled, and normal startup occurs if switch is restarted.

ONIE-based discovery only occurs if no NOS is found on the switch.

M Z y ‘ && ’

Compatible Dell switches

ZTD with SONiC is applicable to all Dell switches that are used with an ONIE bootloader and support the SONiC operating system.

At the time of writing this document, the following are qualified SONiC capable switches:

- 400G — Z9432F-ON, Z9332F-ON, Z9664F-ON (support from SONiC 4.1)
- 100G — Z9100F-ON, Z9264F-ON, S5232F-ON, S5448F-ON (support from SONiC 4.1)
- 25G — S5296F-ON, S5248F-ON, S5224F-ON, S5212F-ON
- 10G — E3248PXE-ON
- 1G — E3248P-ON, N3248TE-ON

ZTP requirements

To build a ZTP environment, the following devices and workloads were used in the lab:

- Linux VM on VMware vSphere
  - VMware vSphere 6.7.0.50000
  - Ubuntu 20.04.4 LTS
    - Docker 20.10.16
    - Docker Compose 1.25.0
  - Docker Hub images
    - HTTPD service: enonicio/apache2
    - DHCP service: rackhd/isc-dhcp-server
- Dell SONiC Enterprise image
  - ‘Enterprise_SONiC_OS_4.0.0_Enterprise_Premium.bin’
- Dell Z9100 switch (or another Z- or S- type switch)

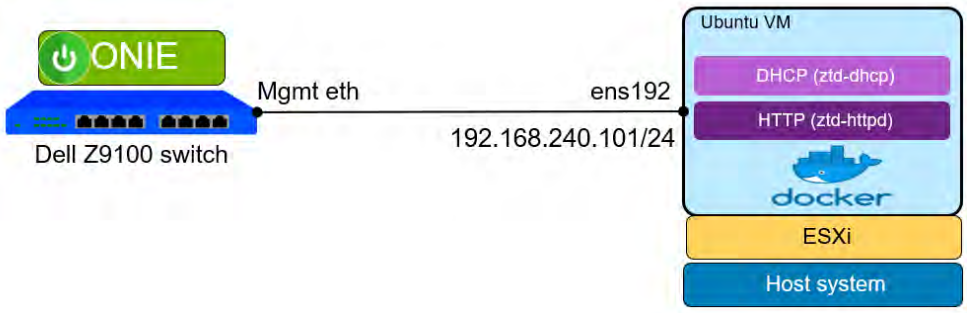

Figure 2. ZTD setup for SONiC

Note: It is the reader's responsibility to use other resources or equipment than those described in this whitepaper. Other than Ubuntu, there are numerous Linux distributions.

While a VM was used on VMware, it could also be used on KVM or bare metal.

Container setup for the web and DHCP service is not mandatory but was used in this environment. The webserver and DHCP servers could also be used as: Bare metal servers, VM-based, or bundled on one host or dedicated per service.
Following the exact steps with same resources and versioning will yield correct end results.
ZTP is also successfully tested on a GNS3 environment; however, firmware upgrade is not available on a GNS3 image. Config install and provisioning script execution will work as expected.

Assumptions

Before starting the setup of the ZTP environment, it is assumed you have set up a fresh Ubuntu Linux server, including:

- Configured IP address 192.168.240.101/24 on the Ethernet interface
- Internet connectivity
- Layer3 link between the server and the Dell switch

Set up essential Linux packages

If Docker and Docker Compose are already available on the Linux VM, proceed to the next chapter; otherwise, perform the following steps to put the Linux VM in the correct state:

```
sudo apt update
sudo apt install net-tools -y
sudo apt install -y apt-transport-https ca-certificates curl
software-properties-common
sudo add-apt-repository "deb [arch=amd64]
https://download.docker.com/linux/ubuntu focal stable"
sudo apt update
sudo apt-cache policy docker-ce
sudo apt install -y docker-ce docker-compose systemctl status
docker -n0
sudo usermod -aG docker ${USER}
```

Create /tftpboot root folder for webserver

Perform the following steps to create /tftpboot folder structure:

```
sudo mkdir /tftpboot && cd tftpboot
sudo mkdir sonic && cd sonic
sudo mkdir config firmware postscript ztp
sudo chown -R www-data.www-data /tftpboot
```

DHCP server configfile setup

Create a /etc/dhcpd.conf file and ensure the following items are included:

```
default-lease-time 600;
      max-lease-time 1800;
      ddns-update-style none;
      authoritative;
      option domain-name-servers 8.8.8.8; #Put in the name-servers
you would like
      option ztp_json_url code 67 = text; #SONiC option for ztp
offering
```

```
      #This is the subnet where the switch management interfaces
connect to
      subnet 192.168.240.0 netmask 255.255.255.0 {
            option routers 192.168.240.254;
      }

      group {
            host ztd-SONiC-Z9100 {
                hardware ethernet 4c:76:25:e8:29:c0; #<<< PUT mac
addresss of your switch mgmt eth
                fixed-address 192.168.240.111;
                option ztp_json_url
"http://192.168.240.101/tftpboot/sonic/ztp/ztp_data_with_firmware.
json ";
            }
      }
```

Docker-compose configfile setup

To smoothly create and destroy the containers for the webserver and the DHCP server, a docker-compose.yml file is required. Create this file in your preferred location. For example, you could use your /home folder. In this case, we have placed the file in /home/ztd/ztdserver/.

```
version: '3'
services:
      ztd-httpd:
            restart: always
            image: "enonicio/apache2"
            ports:
                  - "80:80"
                  - "443:443"
            volumes:
                  - /etc/dhcp:/etc/dhcp
                  - /tftpboot:/var/www/html
      ztd-dhcpd:
            restart: always
            image: rackhd/isc-dhcp-server
            volumes:
                  - /var/lib/dhcp:/var/lib/dhcp
                  - /etc/dhcp:/etc/dhcp
            network_mode: "host"
```

The command used to start services (from where docker-compose.yml file resides):

```
cd /home/ztd/ztdserver/ && docker-compose up &
```

The command used to stop services (from where docker-compose.yml file resides):

```
cd /home/ztd/ztdserver/ && docker-compose down &
```

Note: Use spaces instead of tabs when you create a YAML file; otherwise, you will get errors when the file is used by docker-compose.

Always start the services from within the location where the docker-compose.yml file resides.

The first time you start docker-compose, the images are pulled from dockerhub. Subsequent starts will use a local copy of the image to create a container.

Store SONiC images in web root

The Dell Enterprise SONiC image needs to be uploaded to the /tftpboot folder. Dell uses two images in this lab:

- The default image for the factory default installs and when all NOS have been removed:
- Enterprise_SONiC_OS_3.5.2_Enterprise_Premium.bin
- Desired upgrade image for the production network:
- Enterprise_SONiC_OS_4.0.0_Enterprise_Premium.bin

Upload these files into the web server firmware directory. You can do this with SCP from command line:

```
scp ./Enterprise_SONiC_OS_3.5.2_Enterprise_Premium.bin
ztd@192.168.240.101:
mv /home/ztd/Enterprise_SONiC_OS_3.5.2_Enterprise_Premium.bin
/tftpboot/sonic/firmware/

scp ./Enterprise_SONiC_OS_4.0.0_Enterprise_Premium.bin
ztd@192.168.240.101:
mv /home/ztd/Enterprise_SONiC_OS_4.0.0_Enterprise_Premium.bin
/tftpboot/sonic/firmware/
```

When ONIE does the factory default start it will search for discovery image names. You need to have a default image that can be used when ONIE requests such a discovery image. Create a symbolic link for this pointing to the SONiC 3.5.2 image. This can be the default image which is picked for factory new switches that do not have a NOS yet.

In our example setup, we named the link onie-installer-x86_64.

```
cd /tftpboot
ln -s
sonic/firmware/Enterprise_SONiC_OS_3.5.2_Enterprise_Premium.bin
onie-installer-x86_64
```

When ONIE does a discovery and execution, it will fetch http://192.168.204.101/onie-installer-x86_64

SONiC ZTP json file

The SONiC ZTP json datafile is consumed by SONiC to perform desired tasks during ZTP staging. For this setup, create the file `/tftpboot/sonic/ztp/ztp_data_with_firmware.json.`

ZTP staging involves the following tasks:

- Firmware upgrade if needed
- Install the basic config file `first_boot_config.json`
- Run a provisioning script that will hint to the server that the ZTP staging has finished
- Configure a hostname.

```
{
   "ztp":{

      "firmware": {
         "install": {
            "url":
"http://192.168.240.101/tftpboot/sonic/firmware/Enterprise_SONiC_O
S_4.0.0_Enterprise_Premium.bin",
            "set-default": true
         },
         "reboot-on-success": true,
      },

      "configdb-json": {
         "url": {
            "source":
"http://192.168.240.101/tftpboot/sonic/config/first_boot_config.js
on",
            "destination": "/etc/sonic/config_db.json",
            "secure": false
         },
         "clear-config" : false
      },

      "provisioning-script": {
          "plugin": {

"url":"http://192.168.240.101/tftpboot/sonic/postscript/postscript
_simple.sh"
           },
           "reboot-on-success": false
      }

   }
}
```

Note: The firmware stanza should be used when you have SONiC already in production and you would like to upgrade your fleet of switches to another SONiC version. You activate the ZTP process with the `sudo ztp enable && sudo ztp run -y` command line. If the requested

version equals the running version, it will only download the image and silently discard the firmware.

SONiC config_db

SONiC stores its configuration default in `/etc/sonic/config_db.json`. You can merge your settings into the configuration, or you can overwrite the file completely. The settings of this file are in json. The settings in the following example were prepared to be merged into the default configuration. Keys that are already defined in the on-box configuration are replaced by the ones in the ZTP prepared config file. Store this file as: `/tftpboot/sonic/config/first_boot_config.json`.

Note: Keep in mind that the initial config is only a staging config. Normally, the use case is to have a secure minimum configuration loaded, which ensures that the switch can be managed remotely. You could set specific usernames or passwords, setup or specific access-lists, or protocol activation as a minimum. Then, finish the final configuration with a configuration controller like Ansible or a REST API client.

```
{
    "MGMT_PORT": {
        "eth0": {
            "admin_status": "up",
            "autoneg": "true",
            "description": "dhcp managed interface",
            "mtu": "1500",
            "speed": "1000"
        }
    },
    "DEVICE_METADATA": {
        "localhost": {
            "hostname": "ztp-config-db-staged"
        }
    },
    "COREDUMP": {
        "config": {
            "enabled": "true"
        }
    },
    "VRF": {
        "default": {
            "enabled": "true"
        }
    },
    "MGMT_VRF_CONFIG": {
        "vrf_global": {
            "mgmtVrfEnabled": "false"
        }
    },
    "SSH_SERVER_VRF": {
        "default": {
            "port": "22"
        }
```

```
    }
}
```

Provisioning script

Save the provisioning in `/tftpboot/sonic/postscript/postscript_simple.sh.`

This is the last script executed in the ZTP cycle. In this lab, the script will perform the following:

- Find the DHCP IP address received.
- Try to run a cgi script on the webserver. Do not be concerned if it is not available; it is optional. (See the Starting Zero Touch Deployment chapter to learn what the script does and whether you would like to attempt it.)
- Fetch the desired hostname for the switch from the webserver.
- Configure the desired hostname using a localhost REST API call on the switch.
- Save the config.

Copy the following contents into the `postscript_simple.sh`

```
#!/bin/bash

ZTD_SERVER_IP="192.168.240.101"
ZTD_PATH=/tftpboot
CALLBACK=/callback
CGI=/cgi-bin/callback.sh
HOSTS=/etc/hosts
USER_NAME=admin
PASSWORD=YourPaSsWoRd
MGMT_IP=localhost
APP=http://
APPLOCAL=https://

## Extract the ip-address that was received from dhcp server
DHCP_IP=`hostname -I | awk '{printf $1}'`

# Request callback script at http server (cgi-script)
# This creates a file with name <ip-address> and hostname in it on
the server end
/usr/bin/curl -s ${APP}${ZTD_SERVER_IP}${CGI}

sleep 2

# Fetch switch hostname
SWITCHNAME=`/usr/bin/curl
${APP}${ZTD_SERVER_IP}${ZTD_PATH}${CALLBACK}/${DHCP_IP}`

sleep 2

# Check if switchname was found on webserver and received
```

```
if [ -z "${SWITCHNNAME}" ]
        then
                SWITCHNAME="ztp-staged-done" #Callback script
malfunctioning
        else
                SWITCHNAME=${SWITCHNAME}
fi

echo "Found desired hostname: $SWITCHNAME"

# current hostname of switch
CURRENTHOSTNAME=`hostname`
echo "current hostname: ${CURRENTHOSTNAME}"

# set hostname via localhost REST call
curl -s -k -X PATCH "https://localhost/restconf/data/openconfig-
system:system/config/hostname" -H "accept: */*" -H "Content-Type:
application/yang-data+json" -u ${USER_NAME}:${PASSWORD} -d
"{\"openconfig-system:hostname\":\"$SWITCHNAME\"}"

# Save config permanent
config save -y
```

Set executable attribute:

```
chmod a+x /tftpboot/sonic/postscript/postscript_simple.sh
```

Create callback cgi-script for webserver

This step is optional.

Note: The use of this script is optional. If you do not want this functionality, omit this step, and the hostname on the switch will not be configured in the same way as the hostname in the dhcpd.conf file.

The following script is used on the webserver to create a file when the script is requested. The output file receives its name from the DHCP IP address that is offered to the switch. Within this file, the desired hostname is written. The desired hostname is extracted from the dhcpd.conf file. The SONiC switch will request the callback file and use the desired hostname to configure it on the switch as a final step. After this step, the ZTP staging has finished. The following figure shows the flow of this functionality.

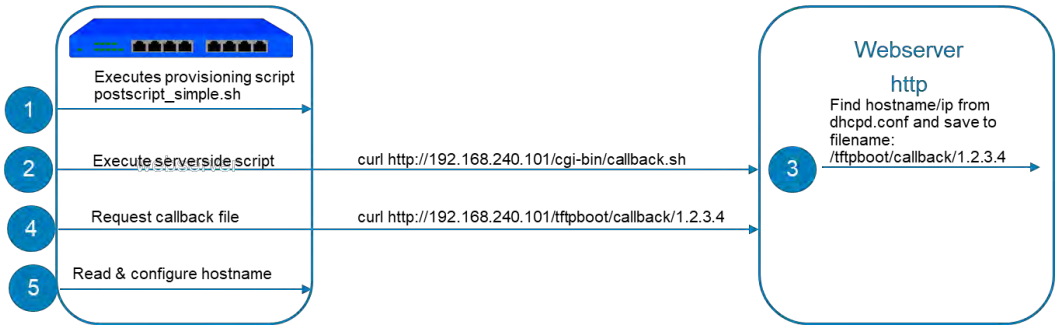

Figure 3. Server side cgi-script to create callback hostname/ip file

As an example, for switch IP address 1.2.3.4 with host entry `my-dell-switch` in the dhcpd.conf file, the callback output file written on the server would be:

```
peter@pp-ztd:/tftpboot/sonic$ cat /tftpboot/callback/1.2.3.4
my-dell-switch
peter@pp-ztd:/tftpboot/sonic$
```

The corresponding entry in `/etc/dhcp/dhcpd.conf` would be:

```
group {
        host my-dell-switch {
                hardware ethernet 4c:76:25:e8:29:c0; #mac of mgmt
eth interface
                fixed-address 1.2.3.4;
                option ztp_json_url
"http://192.168.240.101/tftpboot/sonic/ztp/ztp_data_with_firmware.
json";
        }
}
```

The cgi-script needs to be created and pushed into the HTTPD docker container. First, you create the script. You can store it to your home directory for convenience. After you store it in your container, you do not have to keep it in your home folder. Create the script and set the execute attribute:

```
touch callback.sh && chmod a+x callback.sh
```

Copy and paste the following into the file:

```
#!/bin/bash

DHCPD_CONF=/etc/dhcp/dhcpd.conf
TFTPBOOT=/var/www/html/tftpboot
CALLBACK=callback
SED=`type -p sed`
GREP=`type -tP grep`

## Silent run of SED
## Find lines matching host & start with leading spaces
```

```
## Remove these leading characters & Store what is left in the
hold space
## Find fixed-address with the requesting IP address of the device
## x: Swap pattern and hold space
## p: Print pattern space
## q: Stop processing
## Second SED line is to strip {
$SED -n '/^ *host /{s///;h};/^ *fixed-address
'"$REMOTE_ADDR"'/{x;p}' $DHCPD_CONF >
$TFTPBOOT/$CALLBACK/$REMOTE_ADDR
$SED -i 's/ {//g' $TFTPBOOT/$CALLBACK/$REMOTE_ADDR

## These lines are needed to echo back to apache server and avoid
errors
echo Content-type: text/html
echo
```

Copy the `callback.sh` into the container. Note the container id:

```
docker ps | grep "enonicio/apache2"
<snip>
8b860abfb41f
```

Upload the `callback.sh` script to the webserver `cgi-bin` folder:

```
docker cp callback.sh 8b860abfb41f:/usr/lib/cgi-bin/
```

Be sure to make the script permanent to your webserver image. It is beyond the scope of this whitepaper to cover specific details. See documentation about Docker images and container management.

If the switch requests `http://192.168.240.101/cgi-bin/callback.sh` from the `provisioningsscript postscript_simple.sh`, the webserver runs the `callback.sh` script and saves the desired hostname into the file with the IP address as filename.

Starting zero touch deployment installations

Method 1 - ONIE discovery install (factory default install)

Provided that the following requirements have been set up, powering on a factory default Dell switch starts ONIE OS discovery automatically. Afterward, SONiC ZTP starts automatically.

If you have not met the following requirements, follow the detailed setup steps from the Compatible Dell switches chapter:

- DHCP server has been set up.
- DHCP server configfile has been prepared.
- Web server has been set up.
- Valid default SONiC image has been uploaded to the webroot folder.

- Valid SONiC data json file has been prepared.
- Switch out of band management (OOBM) interface has DHCP broadcast reachability to the DHCP server (udp/67).

When you power on your switch, the ZTP process starts. The following screenshots show a successful ZTP staging with firmware install, basic SONiC config file, and provisioning script that sets the desired hostname.

First, ONIE will install the SONiC 3.5.2 version by discovery and execution method to automatically find the discovery image. When the switch has rebooted with this version, the SONiC ZTP starts and follows the intended SONiC 4.0 version according to the ZTP json file. It loads a default config and performs the hostname task from the final provisioning script.

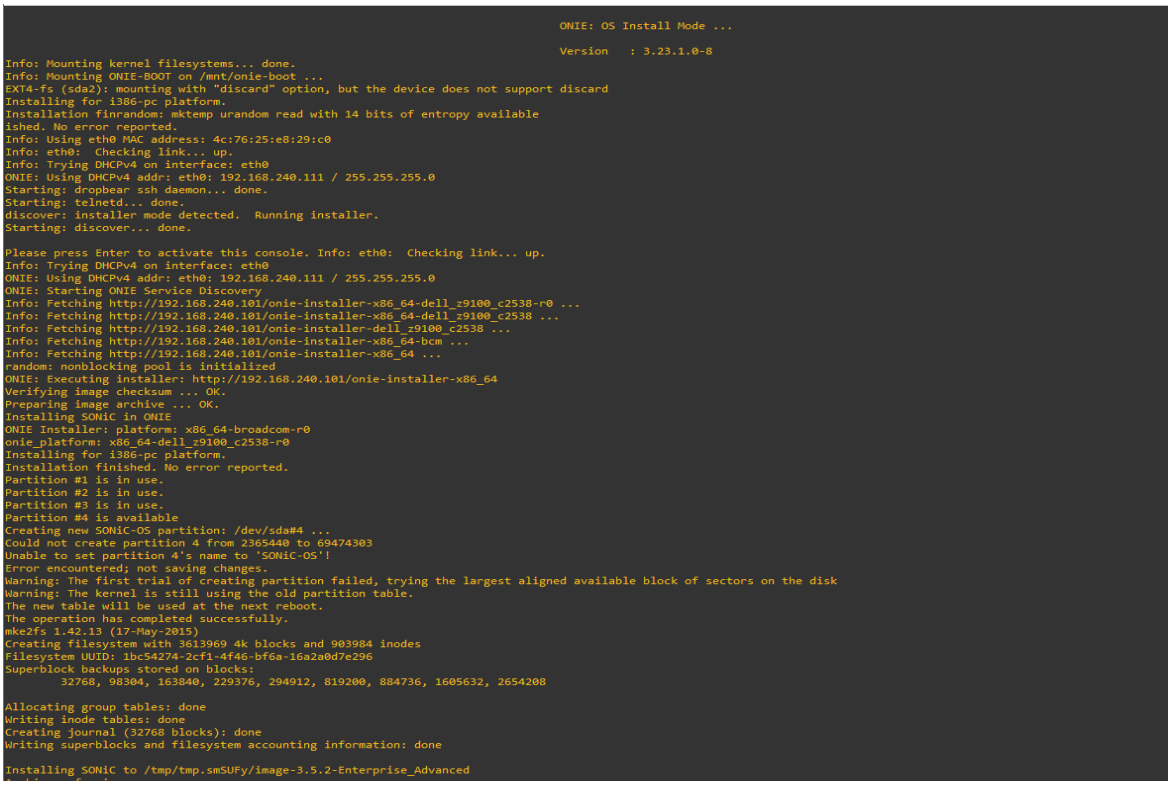

Figure 4. ONIE locates an initial image requested from webserver and installs it

Method 2 - SONiC initiated ZTP install

When the switch is already running with a SONiC version, you can reboot the switch in ZTP mode. If the ZTP mode requires another SONiC version, SONiC creates a multiboot.

SONIC deletes the current SONiC config, reboots, and follows the settings in the ZTP data json file. Before you start the ZTP staging process, be sure that your environment meets the following requirements:

- DHCP server has been set up.
- DHCP server configfile has been prepared.
- Web server has been set up.

- Valid SONiC image has been uploaded to the webserver firmware folder.
- Valid SONiC data jsonfile has been prepared.
- Switch OOBM has IP reachability to the webserver and layer2 DHCP broadcast reachability to the DHCP server (udp port 67).

When the above requirements have been met, log in to the Linux shell of your SONiC switch and run the following commands sequentially (as root):

- sudo ztp enable
- sudo ztp run -y

Note: You could also send the ZTP commands using Ansible or a REST API call to your switch if you have these remote facilities set up. This paper does not cover setting up Ansible or REST API syntax to your SONiC switch.

The switch deletes the SONiC configuration and start ZTP staging flow. The ZTP cycles start with DHCP and request the option67 URL which DHCP offers.

The following screenshot shows the starting screen when ZTP is started from SONiC. It may take 10 minutes for the entire cycle to complete.

```
admin@ztd-Z9100:~$
admin@ztd-Z9100:~$ sudo ztp enable
admin@ztd-Z9100:~$ sudo ztp run
ZTP will be restarted. You may lose switch data and connectivity, continue?[yes/NO] yes
admin@ztd-Z9100:~$ Jun 07 15:03:30.021333 2022 ztd-Z9100 INFO sonic-ztp[22939]: Installing ZTP configuration profile to initiate ZTP discovery.
Jun 07 15:03:36.632975 2022 ztd-Z9100 INFO sonic-ztp[22939]: Platform and HWSKU Valid - /usr/share/sonic/device/x86_64-dell_z9100_c2538-r0/Force10-Z9100-C32
Jun 07 15:03:36.633249 2022 ztd-Z9100 INFO sonic-ztp[22939]: Running command: /usr/bin/db_migrator.py -o check_version -f /tmp/ztp_config_db.json

Zero Touch Provisioning discovery in progress. Please disable ZTP or logout.

Jun 07 15:03:41.003877 2022 ztd-Z9100 INFO sonic-ztp[22939]: FRR Retain Cleared ...
Jun 07 15:03:41.003926 2022 ztd-Z9100 INFO sonic-ztp[22939]: Stopping service udld ...
Jun 07 15:03:43.004118 2022 ztd-Z9100 INFO sonic-ztp[22939]: Stopping service dhcp_relay ...

Zero Touch Provisioning discovery in progress. Please disable ZTP or logout.


admin@ztd-Z9100:~$ exit
logout


Jun 07 15:03:52.953397 System is not ready - Core services are down

 Jun 07 15:03:53.005210 2022 ztd-Z9100 INFO sonic-ztp[22939]: Stopping service swss ...

Debian GNU/Linux 9 ztd-Z9100 ttyS1

ztd-Z9100 login: Jun 07 15:04:39.012664 2022 ztd-Z9100 INFO sonic-ztp[22939]: Stopping service mgmt-framework ...
Jun 07 15:04:39.012711 2022 ztd-Z9100 INFO sonic-ztp[22939]: Stopping service vrrp ...
Jun 07 15:04:39.012749 2022 ztd-Z9100 INFO sonic-ztp[22939]: Stopping service hostcfgd ...
Jun 07 15:04:39.012785 2022 ztd-Z9100 INFO sonic-ztp[22939]: Stopping service nat ...
Jun 07 15:04:40.013587 2022 ztd-Z9100 INFO sonic-ztp[22939]: Stopping service telemetry ...
Jun 07 15:04:40.013631 2022 ztd-Z9100 INFO sonic-ztp[22939]: Stopping service resrcmgrd ...
Jun 07 15:04:40.013674 2022 ztd-Z9100 INFO sonic-ztp[22939]: Running command: queuestat -D
Jun 07 15:04:40.013711 2022 ztd-Z9100 INFO sonic-ztp[22939]: Running command: /usr/local/bin/sonic-cfggen -j /etc/sonic/copp_config.json --write-to-db
Jun 07 15:04:40.013746 2022 ztd-Z9100 INFO sonic-ztp[22939]: Running command: /usr/local/bin/sonic-cfggen -j /etc/sonic/init_cfg.json --write-to-db
Jun 07 15:04:41.013720 2022 ztd-Z9100 INFO sonic-ztp[22939]: Running command: /usr/local/bin/sonic-cfggen -j /tmp/ztp_config_db.json --write-to-db
Jun 07 15:04:43.014126 2022 ztd-Z9100 INFO sonic-ztp[22939]: Resetting failed status for service bgp ...
```

Figure 5. ZTP install initiated from a SONiC switch

Monitoring and diagnostics

Screenshots of an ONIE discovery install

The following figures are screenshots captured during the factory default ZTP staging that occurs when a factory default Dell switch is powered on. The same flow also occurs when you delete all NOSes from the switch, and you reboot the switch.

Figure 6. ONIE starts discovery and execution process for ZTP install and seeks an image

Figure 7. ONIE locates an initial image requested from webserver and installs it

Figure 8. ONIE reboots switch and Grub loads SONiC 3.5.2

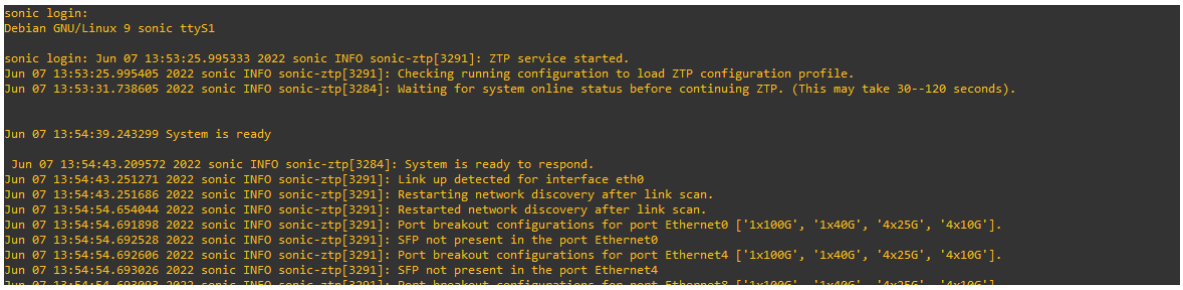

Figure 9. SONiC loads its first run while still in ZTP mode

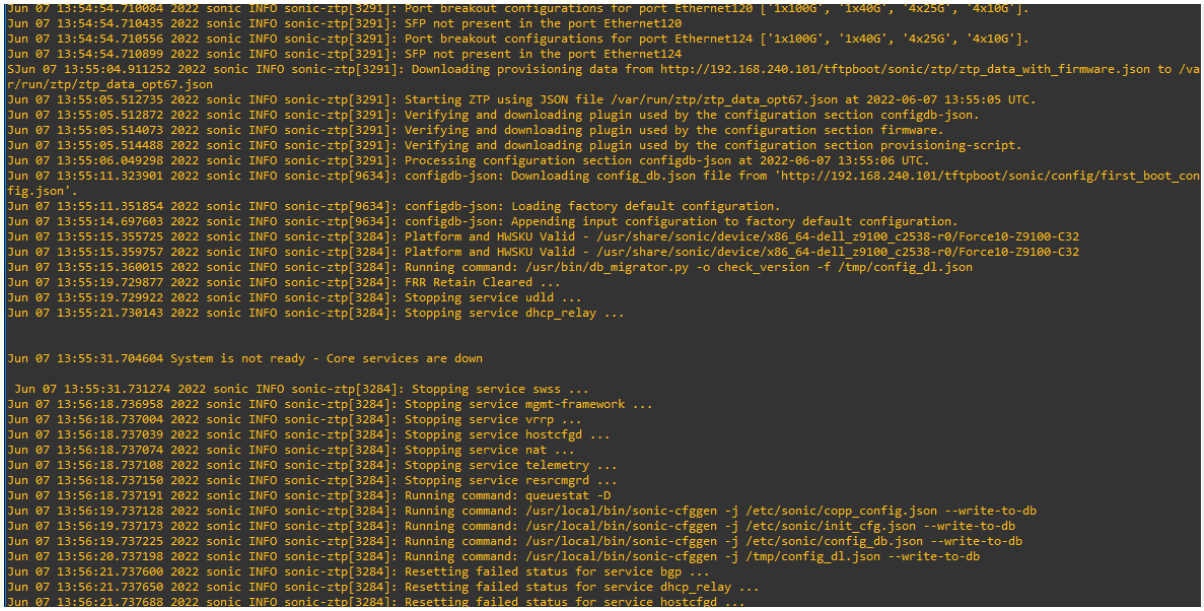

Figure 10. SONiC comes online in ZTP mode and requests ZTP data json file

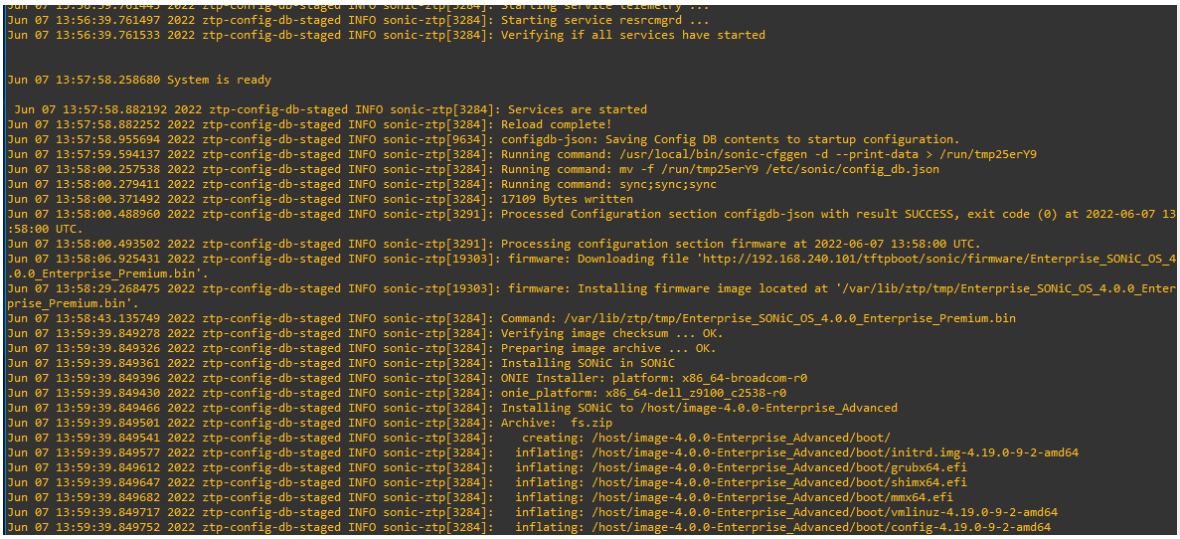

Figure 11. SONiC ZTP downloads all relevant files according to ZTP data JSON file and installs desired firmware

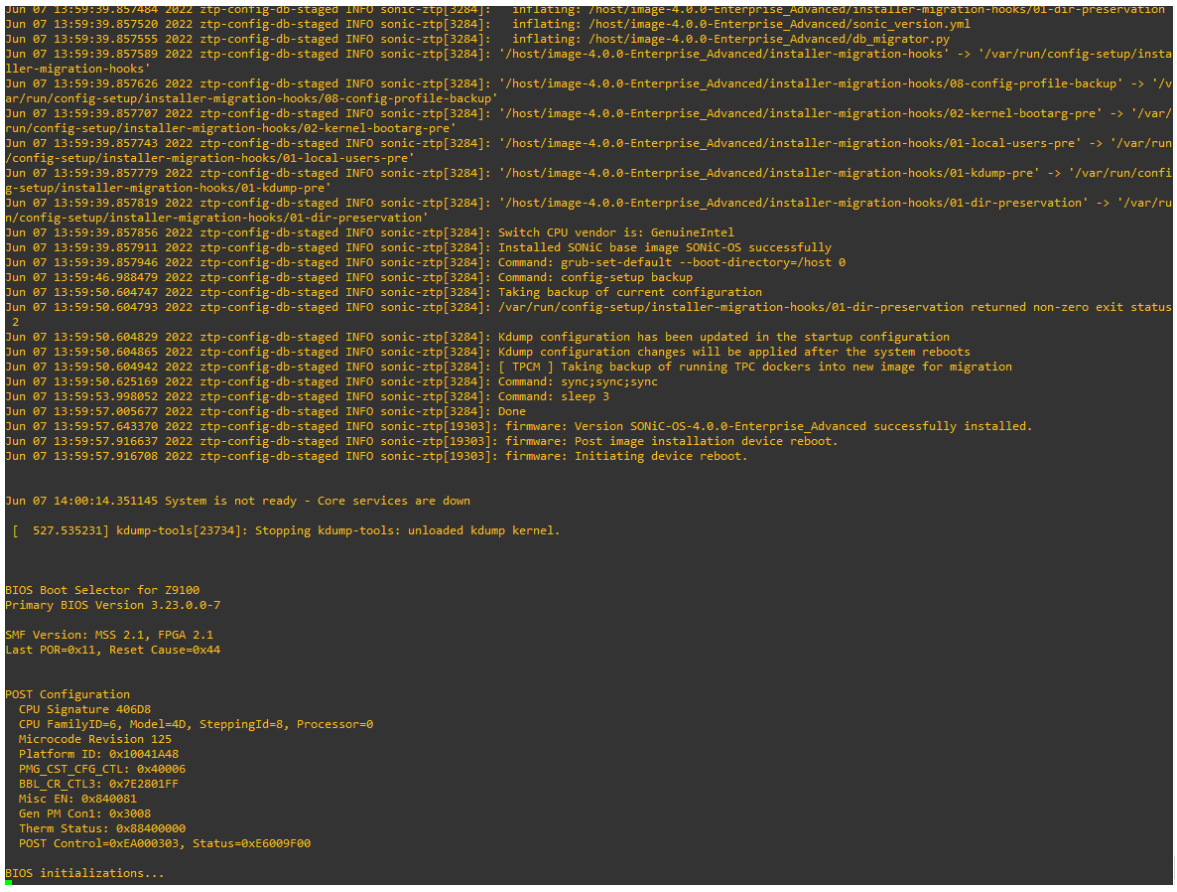

Figure 12. Switch reboots with SONiC 4.0

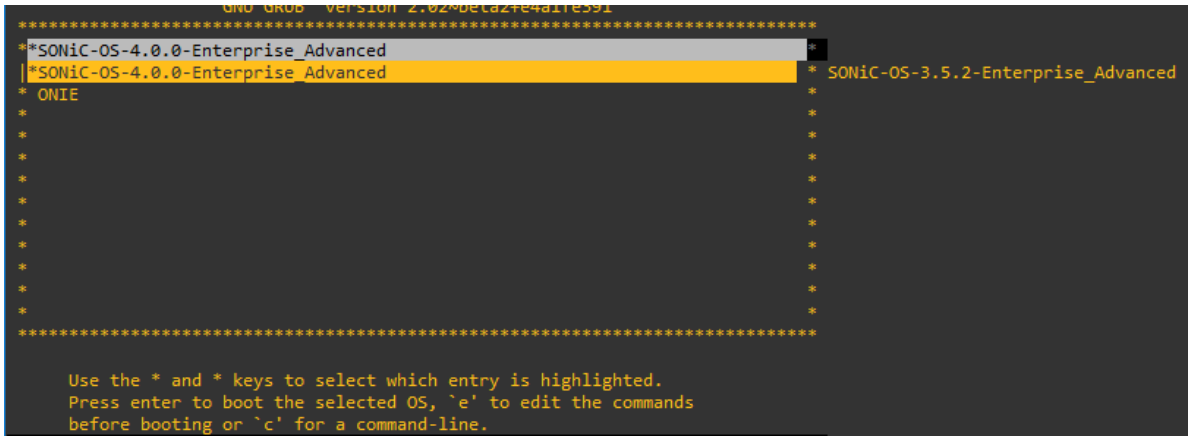

Figure 13. Updated Grub menu with both SONiC versions; 4.0 is default selected

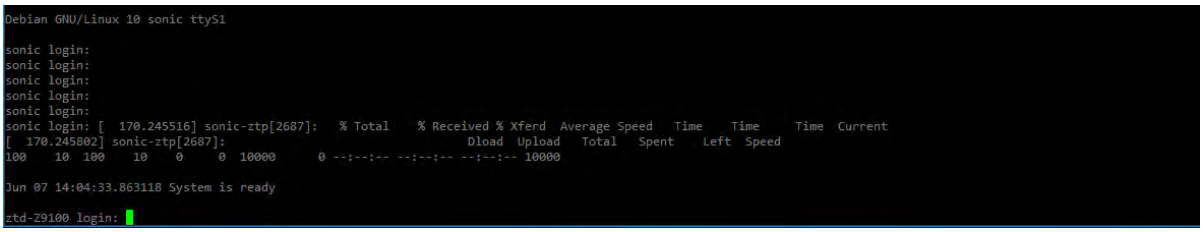

Figure 14. SONiC 4.0 loads and provisioning script finishes

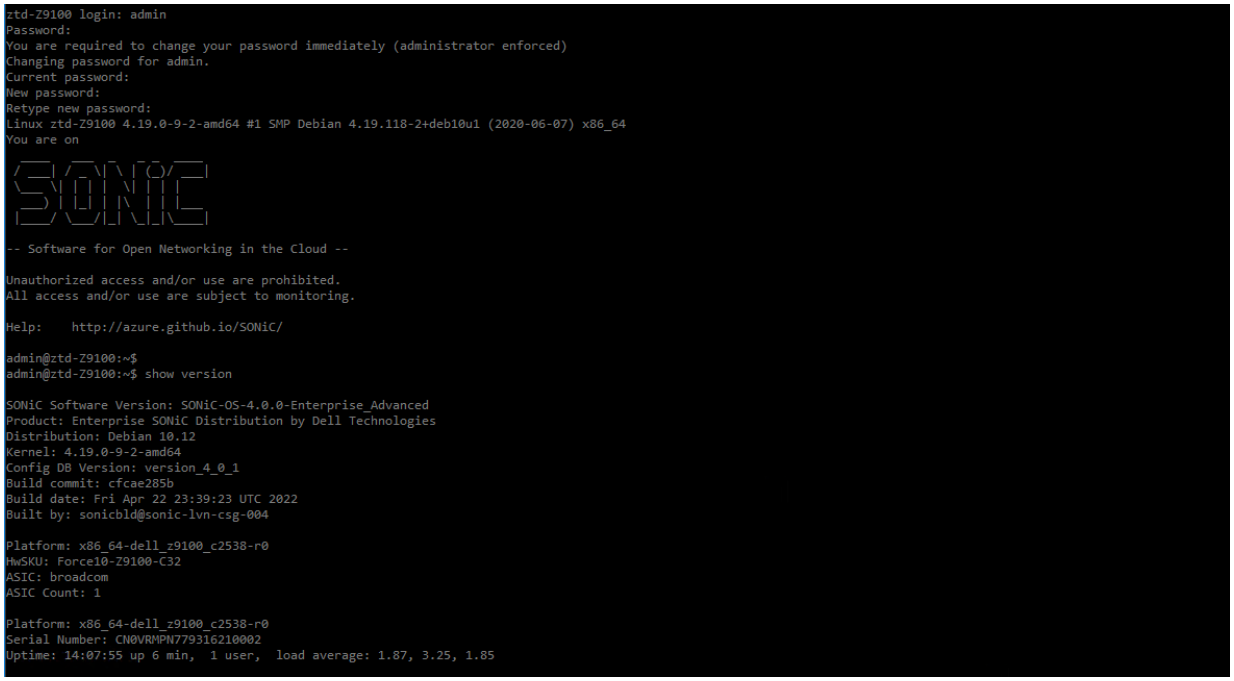

Figure 15. Hostname is set and version check of correct SONiC version

Figure 16. ZTP status shows all JSON data elements correctly finished and ZTP inactive

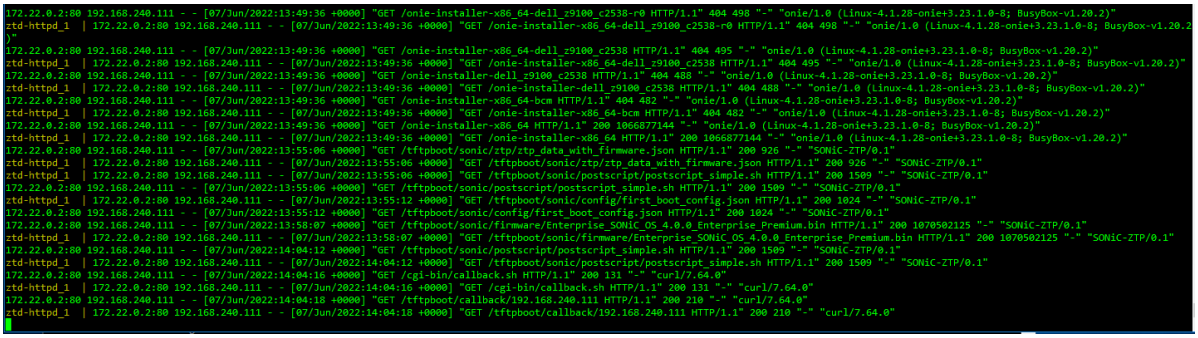

Figure 17. Logging of http container shows all requested files during this factory default ZTP staging flow

Screenshots of a SONiC initiated ZTP install

The following figures are screenshots of a successful ZTP staging with firmware install, basic SONiC config file, and provisionings script.

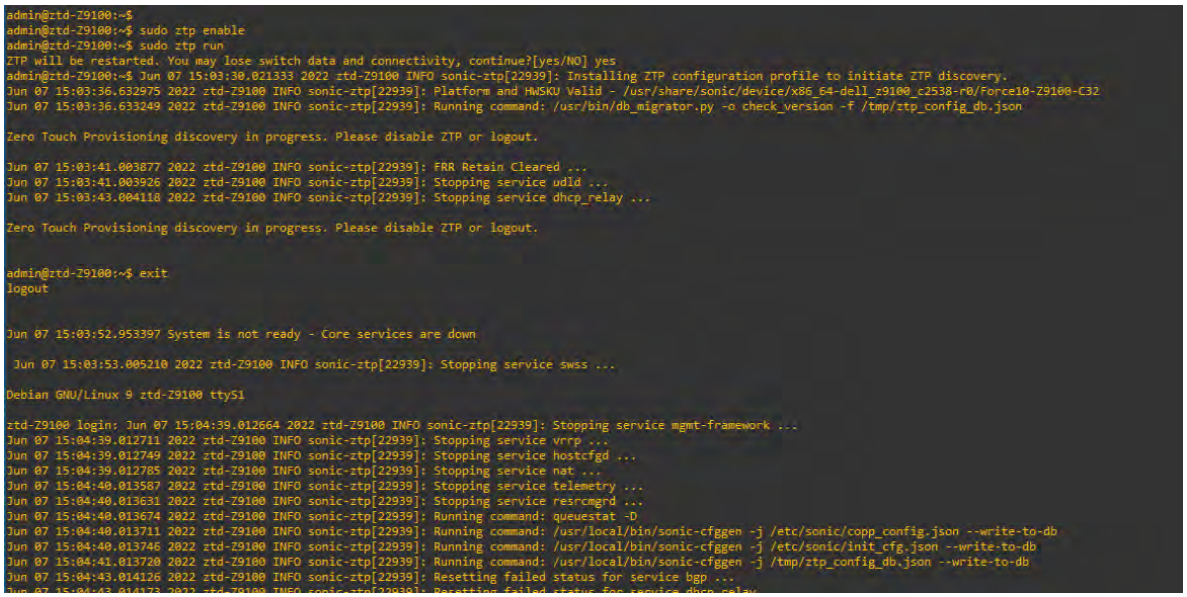

Figure 18. ZTP install started from a SONiC switch

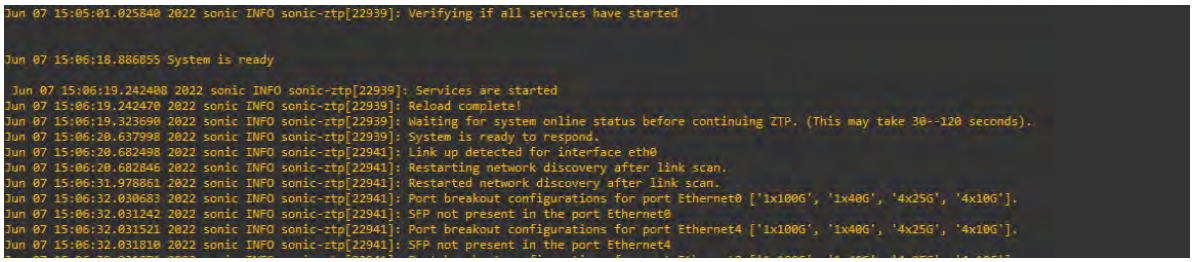

Figure 19. ZTP progress on a SONiC switch

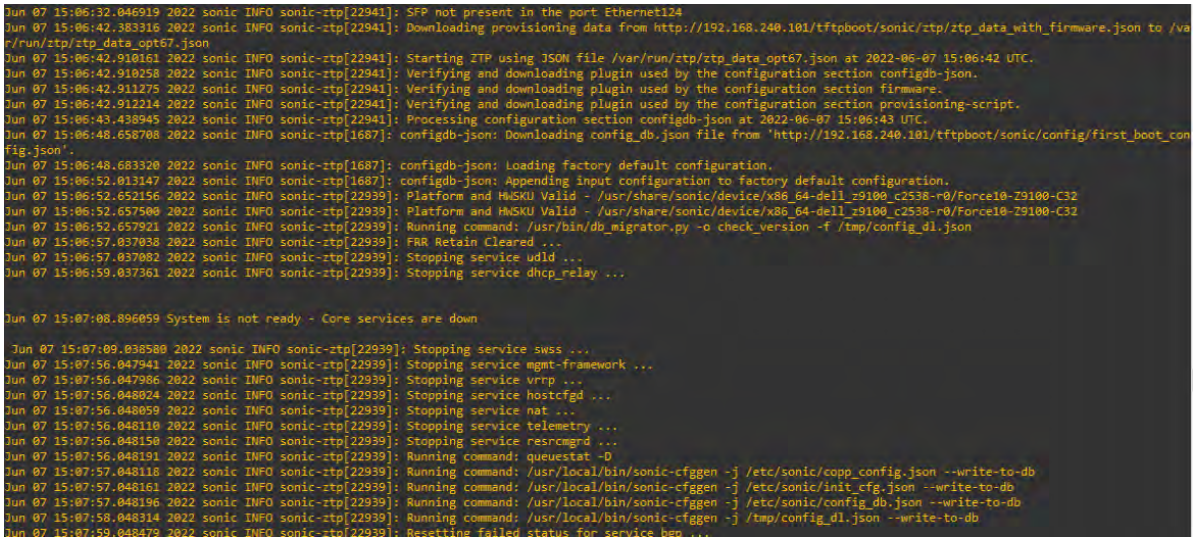

Figure 20. ZTP json file processing on a SONiC switch

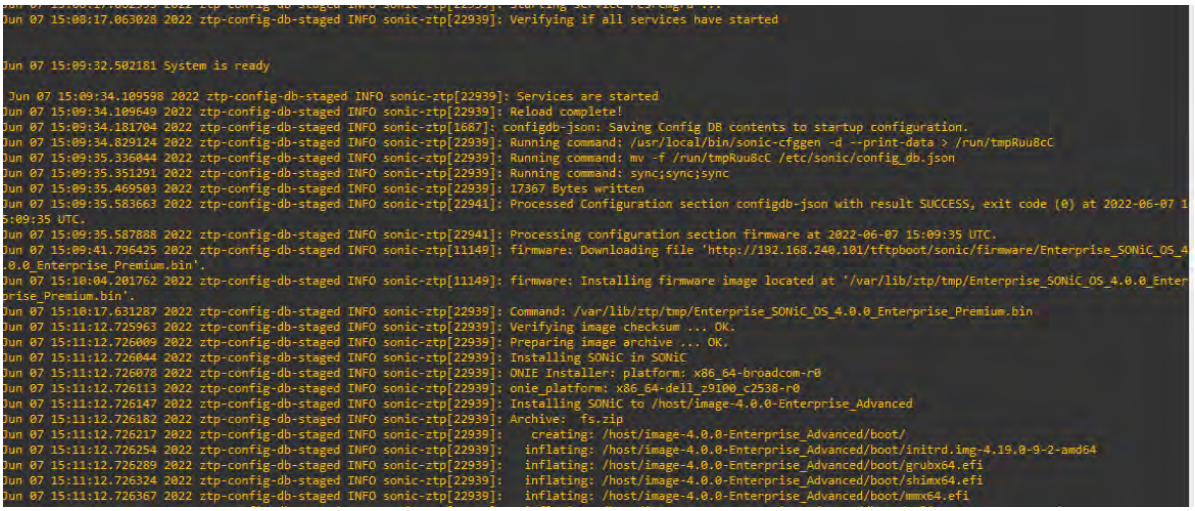

Figure 21. SONiC firmware installation during a ZTP install

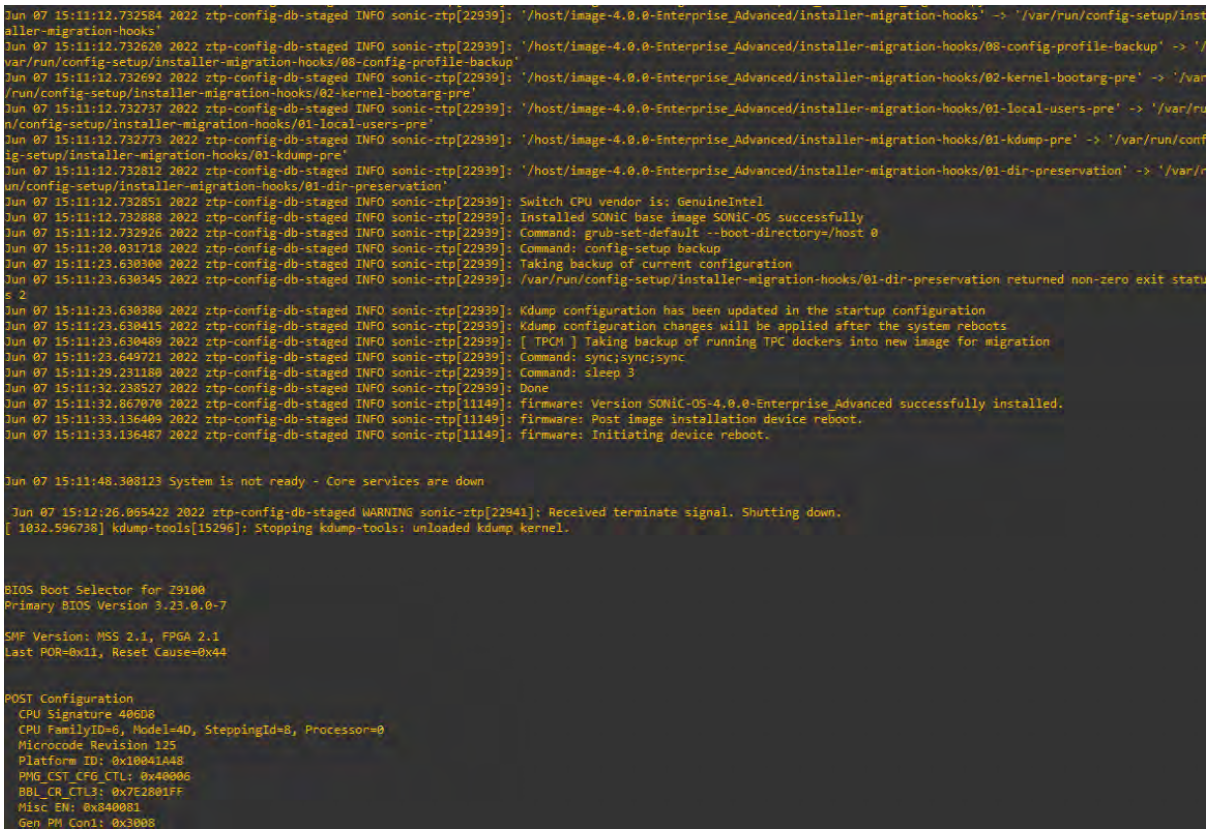

Figure 22. Switch reboots after a new SONiC version is installed

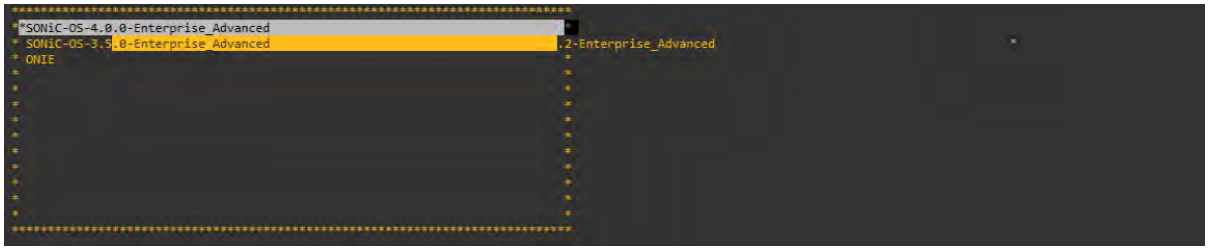

Figure 23. Multiboot Grub menu on a Dell SONiC switch after ZTP install

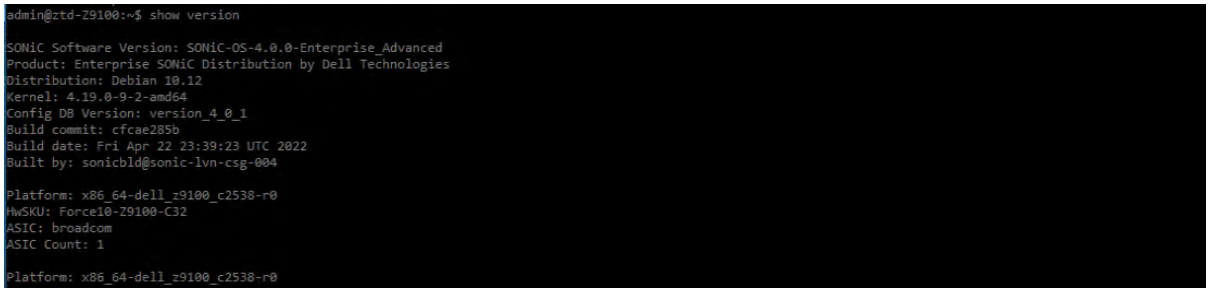

Figure 24. Verification after a successful ZTP install for SONiC

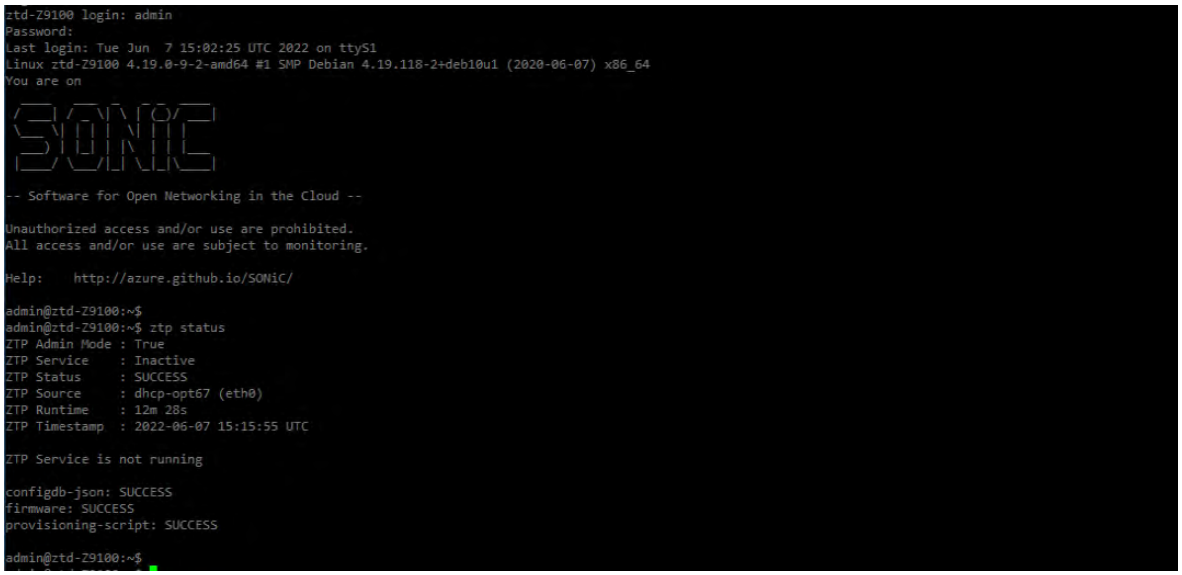

Figure 25. ZTP status shows a successful ZTP installation for SONiC

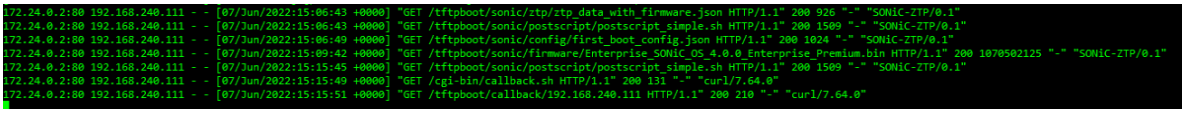

Figure 26. Logging of the http webserver during a SONiC ZTP install which requests required files

Diagnostic commands

The following tables provide an overview of some useful commands to monitor or diagnose status. Some commands are related to the SONiC Click CLI, and others are related to the Dell management CLI (mf-cli or sonic-cli). The click-cli is available after you log in to SONiC. If you type `sonic-cli`, you enter the management framework CLI. This CLI resembles such as the Cisco, Dell OS10 or Arista CLIs. In the SONiC Click shell, you need to run some commands as root, that is, prefix them with `sudo`.

Table 1. Diagnostic commands on SONiC switch

COMMAND	SHELL	PURPOSE
`ztp status`	SONiC click	Show the status of ztp staging
`sudo ztp`	SONiC click	Show available ztp options
`sudo ztp run -y`	SONiC click	Start a ztp install
`sudo config save -y`	SONiC click	Write configuration to config-db.json file
`sudo cat /var/log/ztp.log`	SONiC click	Show SONiC ztp logfile
`sonic-cli`	SONiC click	Log in to management-cli (sonic-cli)
`sudo cat /etc/sonic/config_db.json`	SONiC click	Show SONiC saved configuration file (startup config)
`show interface Management`	Sonic-cli	Show management interface details
`show version`	Sonic-cli	Show version details of SONiC
`onie_mode_set --help`	SONiC click	Show available ONIE options that can be set from SONiC

COMMAND	SHELL	PURPOSE
`sudo onie_mode_set -o uninstall`	SONiC click	Set ONIE to delete all NOSes and start factory default discovery and execution method of ztp installation (reboot needed after command)

Table 2. Diagnostic commands on Linux server host system

COMMAND	SHELL	PURPOSE
`docker-compose ps`	Linux	Show started containers
`docker-compose ps -a`	Linux	Show all containers (started and not started)
`docker ps`	Linux	Show started containers with container ids
`docker ps -a`	Linux	Show all containers with container ids (started and not started)
`docker logs <container id>`	Linux	Show log of container
`docker logs <container id> -f`	Linux	Follow log of container

Problem solving and troubleshooting

Dell highly recommends having a serial console connection available on the switch. This will ensure the best possible monitoring capabilities during all ZTP stages. The following information explains how to connect the console to the RS232 port and how to connect to the microUSB port

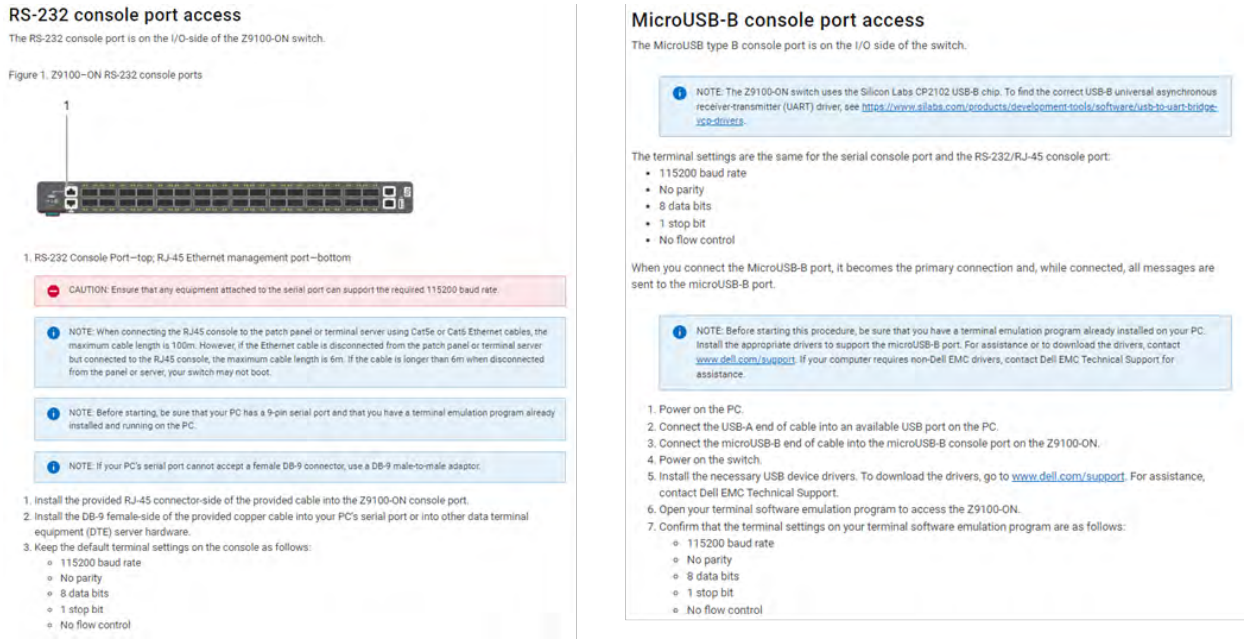

Figure 27. Console to RS232 connection overview

Table 3. Troubleshooting solutions

Problem	Possible causes	Possible solutions
No space left on switch	Too many operating systems installed.	Remove specific NOS or all NOSes with `ONIE Uninstall OS`
ONIE keeps on starting its discovery loop and execution loop	Discovery image could not be requested from webserver.	Check if discovery name is on webserver is correct Check if discovery name is reachable in webserver root Check if webserver is up Check if you can fetch image in a web browser

Firmware image timeout	Management VRF is configured in your initial configdb json file, which prevents the curl process on SONiC from reaching the webserver. Curl does not follow IP routing in the management VRF and thus has no IP routing available in the global table.	Disable management vrf global;
		`"MGMT_VRF_CONFIG": {`
		`"vrf_global": {`
		`"mgmtVrfEnabled": "false"`
		`}`
		`},`

Firmware download aborts unexpected	No space left on switch.	‘ U ’
Failure on requesting ztp json file, firmware, configdb or provisioning script	Availability or reachability of files on webserver.	Check logging of webserver requests Check file rights on webserver Are files available on webserver Start webserver if not started
Errors on processing ztp json file, configdb json or provisioning script	Wrong json syntax or unknown key/values.	Check correctness of files. Use a json lint checker for valid json Use github SONiC ztp repo for key/value options (see Chapter **Error! Reference source not found.**)
Switch gets wrong IP address	MAC-address specified in dhcpd.conf on DHCP server configured in wrong IP subnet.	Configure correct mgmt eth mac-address on DHCP server config
Switch gets no IP address	DHCP server is down or an incorrect mac-address is specified in dhcpd.conf on DHCP server.	Configure correct mgmt eth mac-address on DHCP server config Start DHCP server if not started
DHCP server will not start	Subnet declaration is incorrectly configured.	Check DHCP server log during start
Switch cannot reach webserver IP	Wrong IP address in ZTP URL provided in dhcpd.conf on DHCP server or no option routers are configured.	Check ZTP URL IP address Check option routers IP address in dhcpd.conf
ZTP unsuccessfully finished	Errors in processing scripts.	k ‘ ’ failed and check syntax correctness. Use github SONiC ztp repo for key/value options and syntax (See chapter **Error! Reference source not found.** for the reference link)
Multiple failures on ZTP json file processing	Syntax error.	Start with one plugin and if successful, add one by one. We recommend starting with the configdb-json plugin.

References

- Official Zero touch deployment repository for SONiC on Github
- ONIE Design specification
- Enterprise SONiC Distribution by Dell Technologies - Lifecycle Management
- Enterprise SONiC Distribution by Dell Technologies – Zero Touch Provisioning
- Install Docker Engine on Ubuntu
- Install Docker Compose