ASR5x00 Backing up .chassisid file (chassis ID) on StarOS releases 20 and Higher

Contents

Introduction

This document describes how to back up **.chassisidfile** (chassis ID) on StarOS releases 20 and higher.

Background Information

The chassis key is used to encrypt and decrypt encrypted passwords in the configuration file. If two or more chassis are configured with the same chassis key value, the encrypted passwords can be decrypted by any of the chassis sharing the same chassis key value. As a corollary to this, a given chassis key value cannot decrypt passwords that were encrypted with a different chassis key value.

The chassis key is used to generate the chassis ID which is stored in a file and is used as the primary key for protecting sensitive data (such as passwords and secrets) in configuration files

For release 15.0 and higher, the chassis ID is an SHA256 hash of the chassis key. The chassis key can be set by users through a CLI command or via the Quick Setup Wizard. If the chassis ID does not exist, a local MAC address is used to generate the chassis ID.

For release 19.2 and higher, the user must explicitly set the chassis key through the Quick Setup Wizard or CLI command. If it is not set, a default chassis ID using the local MAC address is generated. In the absence of a chassis key (and hence the chassis ID), sensitive data does not appear in a saved configuration file.

The chassis ID is the **SHA256 hash (encoded in base36 format) of the user entered chassis key plus a 32-byte secure random number.** This assures that the chassis key and chassis ID have 32-byte entropy for key security.

If a chassis ID is not available encryption and decryption for sensitive data in configuration files do not work.

Problem: Insufficient to back up chassis key value to run for

same configuration on the same node.

Due to the change in behavior starting with release 19.2, it is not sufficient anymore to back up the chassis key value to be able to run the same configuration on the same node.

Moreover, because of the random 32 byte number attached to the configured chassis key, there are always different chassis IDs generated based on same chassis keys.

That is the reason why cli command **chassis keycheck** is concealed now since it always returns negative even if the same old key is entered.

To be able to recover a StarOS machine from a saved configuration (when, for example, all contents of the **/flash** drive were lost) it is required to backup the **.chassisid** (where the StarOS stores the chassis ID)

The chassis ID is stored in **/flash/.chassisid** file on StarOS hard drive. The easiest method of backing up this file is to transfer it via some file transfer protocol to a backup server:

As you see the **.chassisid file** is a hidden one and with newer releases it is not possible to do file management operations with hidden files. For example, this error is displayed with release 20.0.1:

```
[local]sim-lte# copy /flash/.chassisid /flash/backup
Failure: source is not valid.
[local]sim-lte#
```

Or:

```
[local]sim-lte# show file url /flash/.chassisid
Failure: file is not valid.
```

Solution

There is still a way to access this file via this procedure:

Step 1. Ensure the **.chassisid** file is present in **/flash/.chassisid**.

```
[local]sim-lte# dir /flash/.chassisid
-rw-rw-r--    1 root     root           53 Jun 23 10:59 /flash/.chassisid
8       /flash/.chassisid
Filesystem           1k-blocks      Used Available Use% Mounted on
/var/run/storage/flash/part1   523992    192112    331880  37% /mnt/user/.auto/onboard/flash
```

Step 2. Login into hidden mode.

```
[local]sim-lte# cli test-commands
Password:
Warning: Test commands enables internal testing and debugging commands
USE OF THIS MODE MAY CAUSE SIGNIFICANT SERVICE INTERRUPTION
```

```
[local]sim-lte#
```

Note: If there is no hidden mode password configured, configure it with this:

```
[local]sim-lte(config)# tech-support test-commands password <password>
```

Step 3. Start a debug shell.

```
[local]sim-lte# debug shell
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
Cisco Systems QvPC-SI Intelligent Mobile Gateway
[No authentication; running a login shell]
```

Step 4. Move in the **/flash** directory. Verify if the file is there.

```
sim-lte:ssi#
sim-lte:ssi# ls
bin cdrom1 hd-raid param rmm1 tmp usr
boot dev include pcmcia1 sbin usb1 var
boot1 etc lib proc sftp usb2 vr
boot2 flash mnt records sys usb3
sim-lte:ssi#
sim-lte:ssi# cd flash
sim-lte:ssi# ls -a
. ldlinux.sys restart_file_cntr.txt
.. module.sys sftp
.chassisid patch staros.bin
crashlog2 persistdump syslinux.ban
crsh2 rc.local syslinux.cfg
```

Step 5. Copy the hidden file to a non-hidden one.

```
sim-lte:ssi# cp .chassisid chassisid.backup
sim-lte:ssi#
sim-lte:ssi#
sim-lte:ssi# ls
chassisid.backup patch staros.bin
crashlog2 persistdump syslinux.ban
crsh2 rc.local syslinux.cfg
ldlinux.sys restart_file_cntr.txt
module.sys sftp
```

Step 6. Exit the debug shell. You should be able to transfer the backup file created without any issues.

```
sim-lte:ssi# exit
Connection closed by foreign host.
[local]sim-lte#
[local]sim-lte# copy /flash/chassisid.backup /flash/chasisid.backup2
********************************************************************************
Transferred 53 bytes in 0.003 seconds (17.3 KB/sec)
[local]sim-lte#
[local]sim-lte#
[local]sim-lte# show file url /flash/chassisid.backup
1ke03dqfdb9dw3kds7vdslvuls3jnop8yj41qyh29w7urhno4ya6
```

UPDATE for Ultra-M upgrade procedure

Upgrading N5.1 to N5.5 will destroy the vpc instance and OSP. Before initiating the upgrade procedure we should backup vPC configuration file and chassis-id if we are to re-use them.

Step 1. backup the chassisid and last configuration file :

```
bash-2.05b# ls -alrt
-rwxrwxr-x 1 root root 53 Jul 11 14:43 .chassisid
-rwxrwxr-x 1 root root 381973 Jul 11 14:41 GGN-2017-07-28.cfg
```

from copied file :

```
cpedrode@CPEDRODE-xxxxx:~/Desktop$ more 2017-07-28.chassis-id
1swbwpd8fd8ca3kf33kn6qxb2h33ihfkqu1tu7x1ndf82znag1b5^@
```

Note: the configuration file will have a derived key from .chasssisid:

```
[local]GGN# show configuration url /flash/GGN-2017-07-28.cfg | more
Monday July 11 14:59:34 CEST 2016
#!$$ StarOS V21.1 Chassis c95bf13f030f6f68cae4e370b2d2482e
config
```

Step 2. Procede with Ultra-M upgrade

Step 3. Once system has been upgraded and StarOS vpc CF bootup, copy chassisid (the regular file) and configuration file (make sure the proper O&M ip address is also changed) to **/flash/sftp** (StarOS >R20)

Step 4. Backup the hidden default .chassisid file from /flash in "test-command" mode and delete it.

Step 5. Copy the chassisid file from /flash/sftp into /flash in hidden mode as "**.chassisid**". Copy the configuration file as well

Note: you can check the derived key issuing cli - *show configuration* url */flash/xxxxxx.cfg* | *more* and compare it with the backup config file

Step 6. Add the boot priority pointing to the new configuration file

Note: At this point StarOS will give an error:

```
[local]GGN(config)# boot system priority 6 image /flash/staros.bin config /flash/GGN-2017-07-
28.cfg
Monday July 28 08:45:28 EDT 2017
Warning: Configuration was generated using a different chassis key, some encrypted information
may not
be valid
```

If you have followed the correct steps, you will have a config file with a Chassis derived key equal to the backup configuration file and a chasssisid equal to the backup chasssisid.

Note that when you view the chassisid file it will append the PS1 prompt :

```
bash-2.05b# cat .chassisid
1swbwpd8fd8ca3kf33kn6qxb2h33ihfkqu1tu7x1ndf82znag1b5bash-2.05b#
```

Step 7. Reboot vPC

At this point, the system should reboot and you may use the login credentials of the backup configuration file.