swissbit®

User Manual

Net Policy Server

for Swissbit Secure Boot Solution for Raspberry Pi

Version: 2.6

Table of Contents

1. Document Information

1.1 About this Document

This document describes how to setup and use an authentication server (Swissbit Net Policy Server), which is one of the three ways to securely boot a Raspberry Pi using Swissbit Raspberry Edition products, using the so-called Net policy. The configuration of the secure-boot Raspberry Pi and also the two other security policies (PIN policy and USB policy) are described in the User Manual "Swissbit Secure Boot SDK for Raspberry Pi".
Please check www.swissbit.com/secure-boot-rpi (➔ Downloads) for the latest version of the Net Policy Server and documentation.

The server consists of three building blocks. A database to store authentication data, a Python script to handle client authentication requests, retrieve data from the data base and send authentication data back to the client, and an Apache webserver providing a Web interface to add, remove and change database entries.

1.2 Glossary

Abbreviation	Description
API	Application Programming Interface
DP	Data Protection
SDK	Software Development Kit
GUI	Graphical User Interface
CLI	Command Line Interface
SO	Security Officer
SHA	Secure Hash Algorithm
PIN	Personal Identification Number Note: In this document PIN is a synonym for a password as any binary value can be defined. In practice the password will most probably be an ASCII PIN
NVRAM	Non-Volatile Random Access Memory

2. Files provided with this Document

This document is contained within the file SwissbitNetPolicyServer.zip. After unpacking it to a directory, e.g. <ServerRoot>, it will have this structure:

├── Database	- Scripts to setup and completely remove the database.
├── Doc	- Location of this document
├── Image	- Directory with the image of the Swissbit Net Policy Server.
│ └── Zip-File	- Zip file of the image of the Swissbit Net Policy Server.
├── Scripts	- Scripts for the Swissbit Net Policy Server (TCP and UDP)

3. Prerequisites

In order to use the Swissbit Netpolicy server, you first need:

- A Raspberry PI and its peripherals
- A Swissbit microSD card (e.g. S-45u) with minimum 8GB capacity

4. NET Policy Server Installation

The Net Policy server stores the Authentication Secret (e.g. PIN) for a specific Swissbit Data Protection device, which is needed to unlock this Data Protection device (e.g. to perform a secure boot).
After the server installation, the unlock policies for a Data Protection device can be configured and managed in the server's Net Policy web interface.
If you have set the NET policy for a certain Data Protection device (e.g. a Swissbit PS-45u "Raspberry Edition"), you need to use the Swissbit Net Policy server image and then configure it in your network. This Net Policy server is designed for operation within a private network – this means that the client and the Net Policy server should both exist in the same private network.

The following steps describe the setup of a Swissbit Net Policy Authentication Server.

4.1 Step 1: Net Policy Server Setup

The server uses the Raspbian stretch version and provides the registered clients with their authentication secret. For an installation of the Swissbit Net Policy Server use the image SwissbitNetPolicyServer.img and follow the steps below to install it on the micro SD card:

1. Download balenaEtcher and install it on a Windows machine
2. Connect a SD card reader with the SD card inside to the Windows machine
3. Open balenaEtcher and select the SwissbitNetPolicyServer.img file
4. Select the target SD card, to which you wish to write the SwissbitNetPolicyServer image.
5. Review your selections and click 'Flash!' to begin writing the image to the SD card.
6. After successful writing the image to the target SD card, please remove the SD card
7. Insert the SD card into your Raspberry Pi (Swissbit Net Policy Server) and boot it up

Note: The default root username is 'pi' and password is 'raspberry'.
Optionally the default password can be change using the command below:

```
$ sudo passwd root
```

Please determine the Net Policy server's private IP address for later use.
In order to obtain the server's IP address, please open a terminal window and use the following command:

```
$ hostname -I
```

Every client device which is intended to be unlocked with this Net Policy server needs to know the Net Policy server's IP address and it needs to be updated every time the Net Policy server's IP address changes.

Therefore, please consider to configure your network DHCP server such that it assigns the Net Policy server a static IP address.

Before proceeding with the installation, execute the two commands below to update the package lists and get the latest version of each package (... this can take a couple of minutes):

```
$ sudo apt-get update
$ sudo apt-get dist-upgrade
```

Note: To frequently update your repositories, you may consider to configure a corresponding cronjob.

4.2 Step 2: Net Policy server's IP address and Port Configuration

The usage of a NET policy requires the availability of a Net Policy server to respond to the requests from the various Data Protection devices, on which the NET policy is active. A shell-script is executed on every start-up of the Net Policy server, which calls a python-script listening on the Net Policy server's IP address on a specific port for any incoming requests. The corresponding response is returned to the same port.
To configure the server's IP address and port in order to respond to the incoming requests, please follow the steps below:

1. Browse to the folder "NetPolicyServer" using the command below:

   ```
    $ cd /var/NetPolicyServer/
   ```

2. Open the file "*constants.json*" with root access.
3. In the file "*constants.json*" change in the line with `"server_IP"` the existing IP address to the Net Policy server's IP address (obtained in Step 1: Net Policy Server Setup).
   Please make sure that the IP address of the Net Policy server stays constant and consider a static DHCP configuration for your Net Policy server on the respective DHCP server in your network.
4. Leave the `"server_port"` on "12375". This should not interfere with any other port listener. (In case it does, the port can be changed anytime in this line.)

Note: In the DataProtection SD card which uses the NET policy, the Net Policy server's IP address and corresponding port need to be entered into the Data Protection SD card's NVRAM in the following format using the Swissbit CardManager (please refer also to the corresponding NET policy chapter in the User Manual "Swissbit Secure Boot SDK for Raspberry Pi" – "Set a NET policy"):

```
NET#<IP-address>#<port>
```

4.3 Step 3: SSL Certificate Generation

Now a SSL Certificate needs to be created, which is used by the Net Policy server for a secure communication via the ssl-encrypted https connection over port 443. This is required to prevent a leakage of a password like the authentication secret during the transaction with the Net Policy server.
In order to create a corresponding SSL certificate, please follow the steps below:

1. Delete the old certificate from the location /etc/apache2/ssl using the command below:

   ```
   sudo rm /etc/apache2/ssl/apache.key /etc/apache2/ssl/apache.crt
   ```

2. Generate the new certificate using the command below:

   ```
   $ sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout
     /etc/apache2/ssl/apache.key -out /etc/apache2/ssl/apache.crt
   $ sudo chmod 600 /etc/apache2/ssl/*
   ```

   The certificate expires after 365 days. For further use create a new SSL certificate using the same command or increase the number of days for the certificate validity.

The information you enter will be seen in your certificate and foremost by the administrator using the web interface. The one important information is the "CommonName". Enter your Net Policy server's IP address or domain there.

4.4 Step 4: Net Policy Server Firewall Configuration

The Swissbit Net Policy Server is using three ports, which need to be accessible via the network connection:

80	for Apache (http)
443	for Apache (https)
12375	for UDP-Server (UDP)

Please note that the port "12375" is chosen arbitrarily and may collide with other services – Please check your network setup accordingly.
If a Raspberry Pi is used as a client with a fresh Raspbian installation, no firewall should restrict any ports on the client device (unless a firewall has been installed and configured accordingly).

4.5 Step 5: Net Policy Server Setup – Finish

In order to finish the Swissbit Net Policy server setup, please follow the steps below:

1. Restart all services by restarting the Net Policy server
   or alternatively
2. To restart the services individually use the corresponding commands below in a terminal window:
   a. Apache:
   ```
   sudo service apache2 restart
   ```
   b. MySQL:
   ```
   sudo service mysql restart
   ```
   c. UDP-Server:
   ```
   sudo /etc/init.d/netpolicy-udpserver start
   ```

5. Net Policy Database Entry Management

The authentication database can be managed after the server setup through a web interface. The web interface allows adding and removing a client device as well as changing the PIN of the client device stored in the database. The database interface can be accessed by entering the address of the Net Policy server in the browser address bar:

e.g. https://<Net Policy server IP address>/NetPolicyServer/ or
https://localhost/NetPolicyServer/

Note: Sometimes, the browser asks for permission to access the unauthorised website – please click on "Proceed".
Enter the default user credentials:

Username: 'user'
Password: 'user'

Optionally, a new username and password can be added using the command below:

```
sudo htpasswd -b /etc/apache2/.htpasswd <new_username> <new_password>
```

To delete a user, open the file "*.htpasswd*" in a text editor and simply erase the corresponding line.
To modify a user's password, delete a user and create a new one.

Note: Any action performed through the Swissbit Net Policy Management Console affects only the values stored inside the database. It does not have any effect on the client device and its configuration.

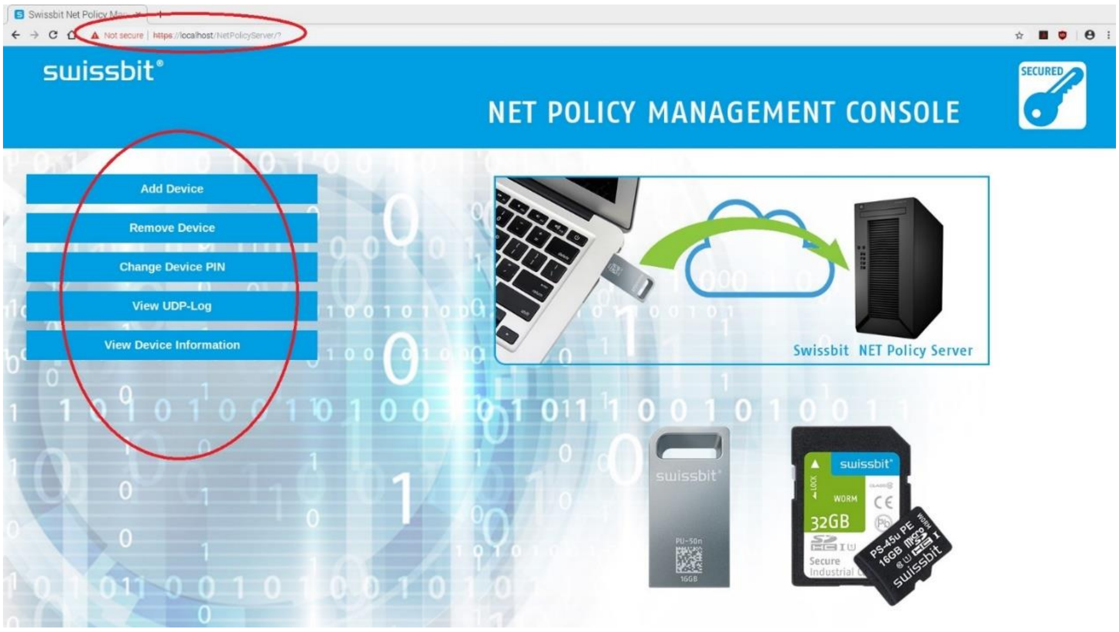

Swissbit Net Policy Ma...
Not secure | https://localhost/NetPolicyServer/?
swissbit
NET POLICY MANAGEMENT CONSOLE
SECURED
Add Device
Remove Device
Change Device PIN
View UDP-Log
View Device Information
Swissbit NET Policy Server

Figure 1: The Net Policy Management Console inside a browser window

5.1 *Adding a New Client Device*

A new Data Protection client device (e.g Swissbit PS-45u "Raspberry Edition") is entered into the database through the "Add Device" button.

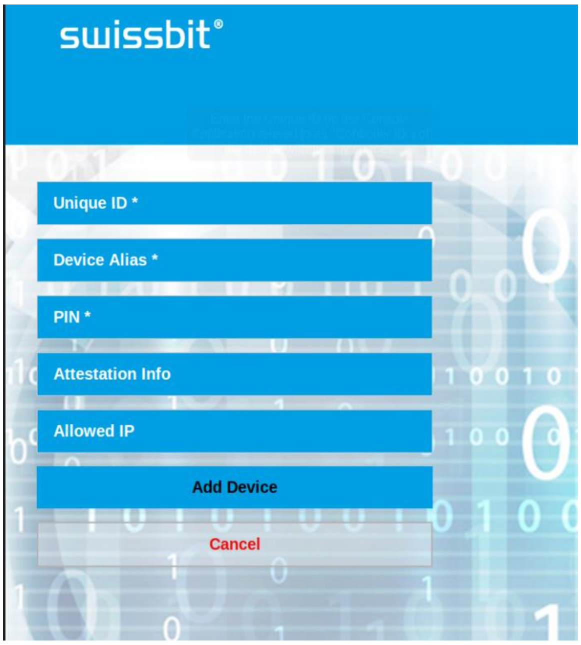


Figure 2: Adding a new device

The client device is identified through its Unique ID. The Unique ID entered must match the Unique ID of the client device. Otherwise, the authentication will fail.

To determine the Unique ID of the Swissbit DataProtection client device, please refer to the Appendix section "Retrieving the Unique ID of a Swissbit DataProtection Card".

The Unique ID has to be entered inside the field Unique ID. It is an alphanumeric value without any blank spaces. Please see ch. 9.1 in the appendix for instructions how to retrieve the Unique ID of a Swissbit DP Card.

Enter a name identifying the card in the field "Device Alias". This name can be freely chosen, but it has to be unique. It is not allowed to enter the same Device Alias for two different client devices.
Enter the user PIN (password) in the field "PIN", which has been chosen when activating the device. If the value entered here does not match the user PIN of the client device, the authentication will fail and the counter of failed logins will be increased on the client device.
Any further information can be entered about the device in the field "Attestation Information".
The fields "Unique ID", "Device Alias" and "PIN" are mandatory fields, all other fields are optional.
Please use the button "Add Device" to store the entries into the database.

5.2 Removing a Device

Select the Device, which should be removed from the database by selecting it from the dropdown list. The devices are represented in the format <alias>#<uniqueID>.

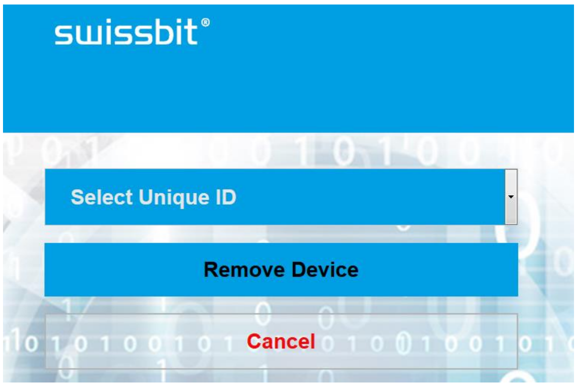


Figure 3: Removing a device

By clicking on "Remove Device" all information stored with the selected device is removed from all tables in the database.

5.3 Changing the PIN of a Device

The PIN of an existing device stored in the database is changed inside the database through the "Change Device" button.
From the "Select Unique ID" drop down box, select the device for which the PIN should be changed. Enter the old PIN and the new PIN and finally press the "Change PIN" button to store the new PIN in the database. If the value in the "Old PIN" field does not match the value currently stored as PIN for the selected device, the PIN will not be changed.
Note: This dialog only changes the PIN stored in the database, not the PIN on the device itself!

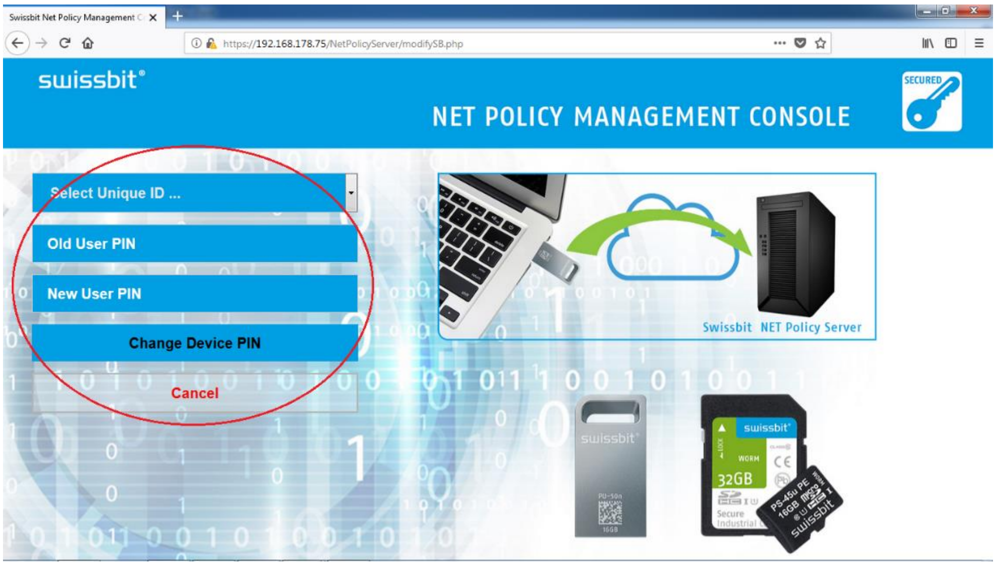
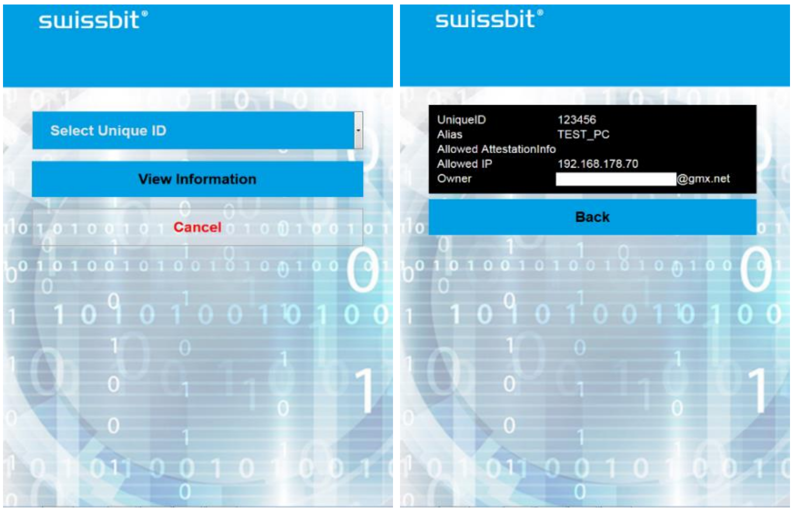

Swissbit Net Policy Management C
https://192.168.178.75/NetPolicyServer/modifySB.php
swissbit®
NET POLICY MANAGEMENT CONSOLE
SECURED
Select Unique ID ...
Old User PIN
New User PIN
Change Device PIN
Cancel
Swissbit NET Policy Server

Figure 4: Change device PIN dialogue inside a browser window

5.4 Inspecting the device Information

Select the Device (given by Unique ID and Alias) in the dropdown table "Select Unique ID" and click "View Information". NULL Values are represented by an empty cell (Figure 5: Inspecting the device information from the web interface).



UniqueID	123456
Alias	TEST_PC
Allowed AttestationInfo	
Allowed IP	192.168.178.70
Owner	[redacted]@gmx.net

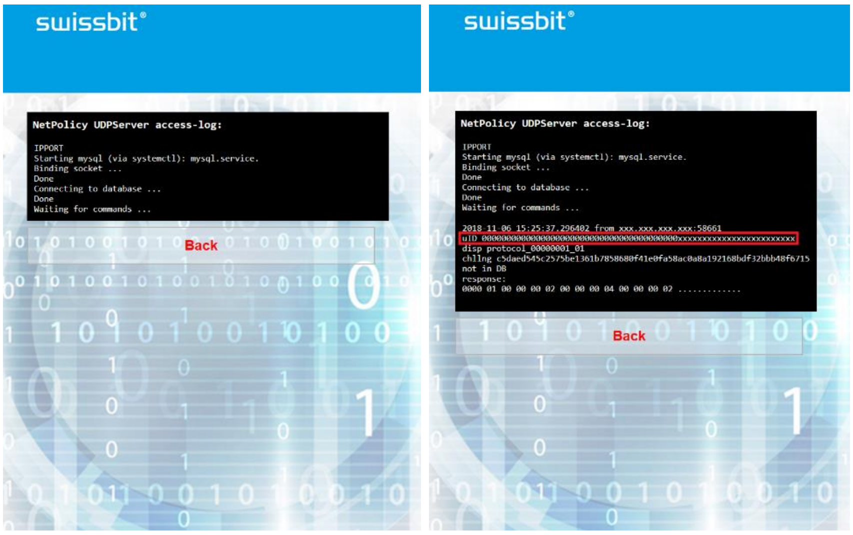
Figure 5: Inspecting the device information from the web interface

5.5 Inspecting the Log-File

The Log displayed in your browser will not synchronize with the actual log file.
Its main purpose is to assist while adding a new device to the database. Every request from a client device (even one, which is not registered to this server yet) is displayed here.

- Request an unlock code for a specific device, open the log and scroll to mentioned request.
- Copy the unique ID, navigate to the add-page and insert the data.

It is not necessary to copy all the leading zeros. They are added automatically when inserting the device information.



```
NetPolicy UDPServer access-log:

IPPORT
Starting mysql (via systemctl): mysql.service.
Binding socket ...
Done
Connecting to database ...
Done
Waiting for commands ...
```

```
NetPolicy UDPServer access-log:

IPPORT
Starting mysql (via systemctl): mysql.service.
Binding socket ...
Done
Connecting to database ...
Done
Waiting for commands ...

2018-11-06 15:25:37.296402 from xxx.xxx.xxx.xxx:58661
uID 0000000000000000000000000000000000000000xxxxxxxxxxxxxxxxxxxxxxxx
disp protocol_00000001_01
chllng c5daed545c2575be1361b7858680f41e0fa58ac0a8a192168bdf32bbb48f6715
not in DB
response:
0000 01 00 00 00 02 00 00 00 04 00 00 00 02 .............
```

Figure 6: Retrieving the Unique ID using the log in the web interface

5.6 Possible Results

Case 1: "Success: Database updated" (Figure 7) is shown once the database ...

- has been updated with the new device information (add) or
- a device information has been removed or
- the stored PIN has been successfully updated.

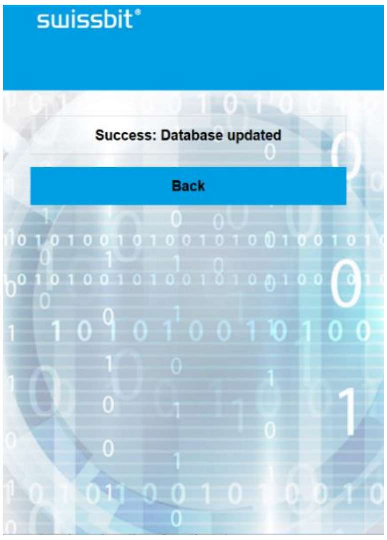
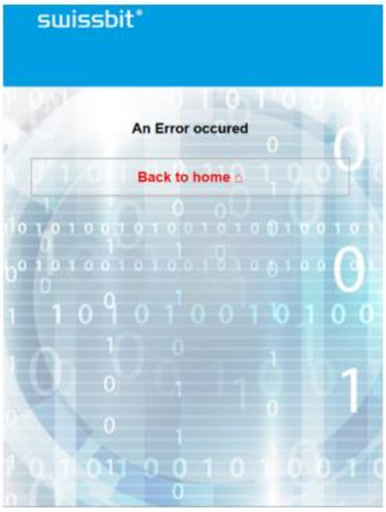


Figure 7: Case 1: Successful query result

Case 2: "An **Error** occurred" (Figure 8): This means that the php script, which is responsible for the database management, reported an error and terminated (leaving any started process unfinished).

Possible reasons:

- The database is not activated or not even installed.
  Solution: Restart the database by restarting the NetPolicy server using the respective commands (see Chapter 4.5). If the error still exists, follow the instructions in Chapter 7 to install the database. Please make sure to remove any previous database instance first.
- If Figure 8 is shown as a result of the "viewing the log" operation, then no log file is available.
  Solution: Create one by restarting the NetPolicy server (see Chapter 4.5).



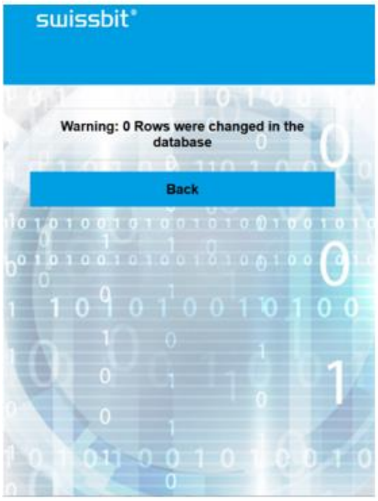
An Error occured

Figure 8: Case 2: Error

Case 3: **"Warning:** 0 Rows were changed in the database" (Figure 9**Fehler! Verweisquelle konnte nicht gefunden werden.**):

This error is shown, when:

- the old Pin does not match the one set in the database (change device Pin)
- no device exists with this unique ID (remove device)

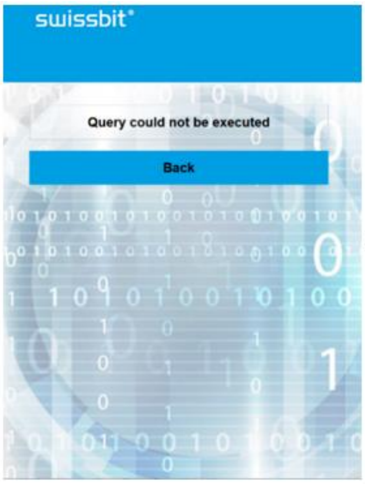


Warning: 0 Rows were changed in the database

Figure 9: Case 3: Database Warning

Case 4: "Query could not be executed" (Figure 10)

This error is shown, when the Unique ID, which is intended to be added, is already assigned.



Query could not be executed

Figure 10: Case 4: Query Error

6. Optional: Managing the TCP and UDP Services

When the Net Policy server has been setup following the instructions in the previous chapters, the TCP and UDP services are launched every time the Net Policy server is started up.
The services are managed by init-scripts, which have two different options.

To use an option requires root privileges:

```
$ /etc/init.d/netpolicy-udpserver <option>
$ /etc/init.d/netpolicy-tcpserver <option>
```

Options:

- `start` : kills all port listeners on the defined port and restarts the service
- `status` : displays the log for each service, which is stored in *\/var/log/NetPolicyServer/access_udp.log*

The “netpolicy-tcpserver” script can also be used by the user as a replacement for the “netpolicy-udpserver” script.

7. Optional: Installing the Net Policy Database

The Net Policy database is the main part of the NETPolicy Server. It stores the Data Protection device’s uniqueID, the hashed authentication secret and an alias. The alias is used as an ASCI text to distinguish the Data Protection devices by human readable names.
Only the local admin is able to modify the whole database (unless it is already implemented differently). During the installation of the Net Policy database, a user is created with the necessary rights for the webserver and for the python script handling the UDP-socket. The access rights include reading from *access_keys*, inserting new rows and updating and deleting old entries. This user is used by the Apache server and the python script.
The Net Policy database can co-exist with other databases as long as there are no interferences concerning database-, table- and user-names.

- Execute the following commands to install the components for the Net Policy database. The latter is going to open the database terminal, which accpets SQL queries as an input.

```
$ apt-get install mariadb-server mysql-server mysql-client -y
$ mysql
```

- Open the file “setup.sql” in the subfolder “database” with a text editor and copy the whole file content into a command line.

  The commands execute the following:
  - Create a database called ‘swissbit_data_access’
  - Create a table called ‘access_keys’ in this database (Please make sure, that this table does not exist in this database with different columns.)
  - Create a user ‘access’ with the password ‘sbitacc’.
    This user only has the necessary rights to read and write into the columns from the Apache server and to read the table from the python script handling the UDP socket for client requests.
- Alternatively, the following command can be executed if nothing else can overwrite the database entries:

```
$ mysql < <pathTo>/setup.sql
```

Note: In case the Net Policy database needs to be removed, the script “*uninstall.sql*” can be used to perform the required steps.

!!! No backup is created and all information in the tables will be removed. !!!

Reference:
https://howtoraspberrypi.com/mariadb-raspbian-raspberry-pi/
https://www.digitalocean.com/community/tutorials/how-to-create-and-manage-databases-in-mysql-and-mariadb-on-a-cloud-server

8. Reference Material

8.1 Swissbit SDK Manual

Swissbit Secure Boot SDK for Raspberry Pi (User Manual)

8.2 Apache

http://www.apache.org/
https://www.digitalocean.com/community/tutorials/how-to-set-up-password-authentication-with-apache-on-ubuntu-14-04
https://variax.wordpress.com/2017/03/18/adding-https-to-the-raspberry-pi-apache-web-server/comment-page-1/
https://www.digitalocean.com/community/tutorials/how-to-create-a-ssl-certificate-on-apache-for-debian-8
https://www.raspberrypi.org/documentation/remote-access/web-server/apache.md

8.3 Database

https://howtoraspberrypi.com/mariadb-raspbian-raspberry-pi/
https://www.digitalocean.com/community/tutorials/how-to-create-and-manage-databases-in-mysql-and-mariadb-on-a-cloud-server

8.4 Raspberry PI

http://www.instructables.com/id/Raspberry-Pi-Launch-Python-script-on-startup/

9. Appendix

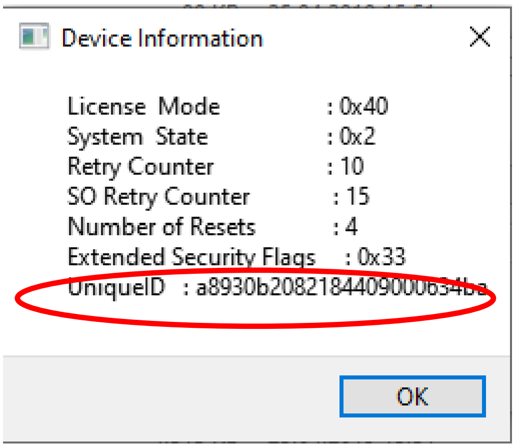
9.1 Retrieving the Unique ID of a Swissbit DP Card

In order to obtain the Unique ID of the client Swissbit DataProtection device, please follow the instructions below:

Option 1:

Get the Unique ID of the Swissbit microSD card for the NET policy server:

1. Please insert the Swissbit DP card into a Windows machine.
2. Start the Swissbit Device Manager
3. Go to menu "Information > Device Status" or press "CTRL-S"
4. Write down the UniqueID of the Swissbit microSD card



```
License  Mode              : 0x40
System  State              : 0x2
Retry Counter              : 10
SO Retry Counter           : 15
Number of Resets           : 4
Extended Security Flags    : 0x33
UniqueID  : a8930b2082184409000634ba
```

Figure 11

Option 2:

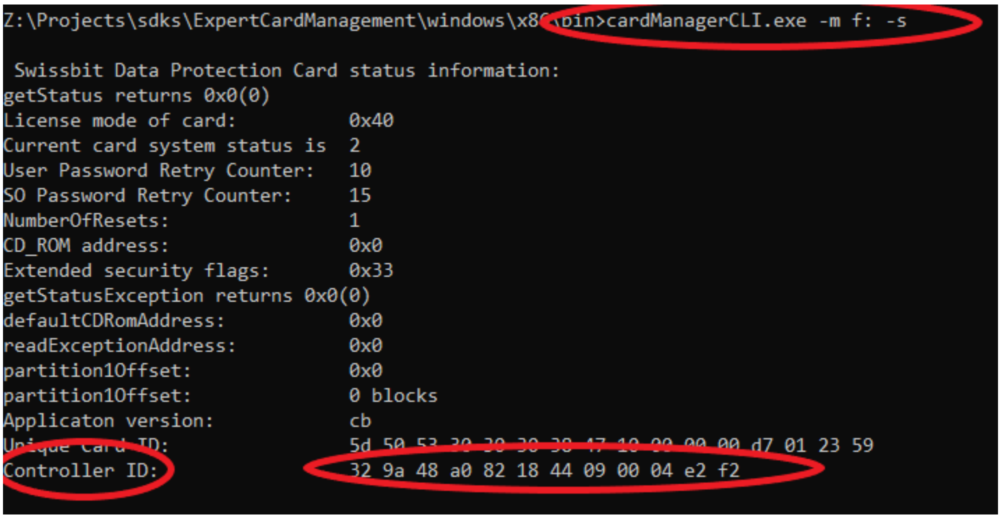
1) Please insert the Swissbit DP card into a Windows machine.
2) Note the drive letter, which is assigned to the device.
3) Open a cmd Window and navigate to the folder, where the tool cardManagerCLI.exe from the Swissbit Secure Boot SDK is located.
4) Execute this command:

   cardManagerCLI.exe -m <DriveLetter> -s

   <DriveLetter> denotes the drive letter, which is assigned to the DataProtection device. E.g if the DataProtection device has the drive letter "f" assigned, the corresponding command would be:

   cardManagerCLI.exe -m f: -s
5) Please note the last value in the output ("Controller ID", Figure 12). This alphanumeric sequence without any blank spaces is the Unique ID of the DataProtection device, which is needed for the Net Policy database entry in the Net Policy server.

   Note: It is not the value shown as the Unique Card ID!



```
Z:\Projects\sdks\ExpertCardManagement\windows\x86\bin>cardManagerCLI.exe -m f: -s

 Swissbit Data Protection Card status information:
getStatus returns 0x0(0)
License mode of card:          0x40
Current card system status is  2
User Password Retry Counter:   10
SO Password Retry Counter:     15
NumberOfResets:                1
CD_ROM address:                0x0
Extended security flags:       0x33
getStatusException returns 0x0(0)
defaultCDRomAddress:           0x0
readExceptionAddress:          0x0
partition1Offset:              0x0
partition1Offset:              0 blocks
Applicaton version:            cb
Unique Card ID:                5d 50 53 30 30 30 30 47 10 00 00 00 d7 01 23 59
Controller ID:                 32 9a 48 a0 82 18 44 09 00 04 e2 f2
```

Figure 12: Retrieving the Unique ID of the client DataProtection SD card for the Net Policy server from a Windows command prompt

10. Document History

Version	Updated on	Updated by	Short description
2.6	April 20th, 2020	Swissbit AG	First public release