

Wrapper Python

Project Documentation

Release: August 30th, 2019





Proprietary Notice

The following document offers information, which is subject to the terms and conditions described hereafter.

While care has been taken in preparing this document, some typographical errors, error or omissions may have occurred. We reserve the right to make changes to the content and information described herein or update such information at any time without notice. The opinions expressed are in good faith and while every care has been taken in preparing this document, some typographical errors, error or omissions may have occurred. We reserve the right to make changes to the content and information described herein or update such information at any time without notice. The opinion expressed are in good faith and while every care has been taken in preparing this document.

Authors

Nicoló MAUNERO (*CINI Cybersecurity Natinal Lab*) nicolo.maunero@consorzio-cini.it

Paolo PRINETTO (*President, CINI*) paolo.prinetto@polito.it

Antonio VARRIALE (*Managing Director, Blu5 Labs Ltd*) av@blu5labs.eu

Trademarks

Words and logos marked with ® or ™ are registered trademarks or trademarks owned by Blu5 View Pte Ltd. Other brands and names mentioned herein may be the trademarks of their respective owners. No use of these may be made for any purpose whatsoever without the prior written authorization of the owner company.

Disclaimer

THIS DOCUMENT AND THE INFORMATION CONTAINED HEREIN IS PROVIDED ON AN “AS IS” BASIS AND ITS AUTHORS DISCLAIM ALL WARRANTIES, EXPRESS, OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY TAHT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OR MERCHANTABILITY OR FITNESS FOR A PURPOSE. THE SOFTWARE IS PROVIDED TO YOU “AS IS” AND WE MAKE NO EXPRESS OR IMPLIED WARRANTIES WHATSOEVER WITH RESPECT TO ITS FUNCTIONALITY, OPERABILITY, OR USE, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PURPOSE, OR INFRINGEMENT. WE EXPRESSLY DISCLAIM ANY LIABILITY WHATSOEVER FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, INCIDENTAL OR SPECIAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOSS REVENUES, LOST PROFITS, LOSSES RESULTING FROM BUSINESS INTERRUPTION OR LOSS OF DATA, REGARDLESS OF THE FORM OF ACTION OR LEGAL THEREUNDER WHICH THE LIABILITY MAY BE ASSERTED, EVEN IF ADVISED OF THE POSSIBILITY LIKELIHOOD OF SUCH DAMAGES.





Contents

1	Features	6
2	Wrapper Python Installation Guide	7
3	Examples	7
3.1	SEcube™ Initialization	7
3.2	Add Key	7
3.3	Functional Test	8



1 Features

This project is intended to present python wrapper for the **SEcube™** Host-Side API. You can find information on what the Host-Side API provides in terms of functionalities and how they are structured in the general wiki that you can find on the **SEcube™** website¹

¹<https://www.secube.eu/resources/open-sources-sdk/>



2 Wrapper Python Installation Guide

In order to install and start using the python wrappers download the **SEcube™** SDK from the **SEcube™** website², then perform the following steps:

```
cd SEcube-py/SEcube
make clean
make
make install
cd bin
sudo cp secube-x86.so /usr/lib
```

Now everything is ready for running the provided example and starting developing your projects. In order to modify the source code which is used by the python wrapper perform the following steps:

```
cd SEcube-py/SEcube/secube-wrapper
Modify the file you want
cd ..
make clean
make
make install
cd bin
sudo cp secube-x86.so /usr/lib
```

Now your modifications are applied.

3 Examples

Here is a list of provided examples and how to use them.

3.1 SEcube™ Initialization

This python script provides an example of how to initialize the **SEcube™** after it has been programmed. Remember that the initialization has to be done the very first time and every time a full flash erase is performed on the **SEcube™** DevKit, if you just re-program it without a full flash erase performing again the initialization is not required.

```
cd SEcube-py/SEcube/py
python3 secube_init.py
```

3.2 Add Key

This python script provides an example of how to add new keys to the **SEcube™**. The script receives as input the key to be added and it has to be in hexadecimal format (you can use this website³ to convert a string).

```
Encryption key (AESWSSBoella201820192020)
cd SEcube-py/SEcube/py
sudo python3 add_key.py 0 Encryption_key_1 414553575353426
f656c6c61323031383230313932303230
```

²<https://www.secube.eu/resources/open-sources-sdk/>

³<https://codebeautify.org/string-hex-converter>



```
Disgest key (HMACSHAWSSBoella2018SHAWSSBoella)
sudo python3 add_key.py 1 Digest_key_1 484
d4143534841575353426f656c6c6132303138534841575353426f656c6c61
```

3.3 Functional Test

This script performs some operations such as logging-in, encrypt and decrypt a string, etc. in order to test if the **SEcube™** has been initialized correctly.

```
cd SEcube-py/SEcube/py
python3 provaEncDec.py
```

