STM32MP1 Series Signing Tool software description

Introduction

The STM32MP1 Series Signing Tool software (named STM32MP1-SignTool in this document) is integrated in the STM32CubeProgrammer (STM32CubeProg).

STM32MP1-SignTool is a key tool that guarantees a secure platform and ensures the signing of binary images using ECC keys generated by STM32MP1-KeyGen software (refer to the user manual *STM32MP1 Series Key Generator software description* (UM2542) for more details).

The signed binary images are used during the STM32MP1 Series MPU secure boot sequence that supports a trusted boot chain. This action ensures an authentication and integrity check of the loaded images.

STM32MP1-SignTool generates a binary image file, a public key file and a private key file.

The binary image file contains the binary data to be programmed for the device.

The public key file contains the ECC public key in PEM format, generated with STM32MP1-KeyGen.

The private key file contains the encrypted ECC private key in PEM format, generated with STM32MP1-KeyGen.

A signed binary file can also be generated from an already signed file with the batch file mode. In that case, the image entry point, the image load address and the image version parameters are not mandatory.

1 Install STM32MP1-SignTool

This tool is installed with the STM32CubeProgrammer package (STM32CubeProg). For more information about the set-up procedure, refer to the section 1.2 of the user manual *STM32CubeProgrammer software description* (UM2237).

This software applies to the STM32MP1 Series Arm®-based MPUs.

Note: Arm is a registered trademark of Arm Limited (or its subsidiaries) in the US and/or elsewhere.

2 STM32MP1-SignTool command line interface

The following sections describe how to use STM32MP1-SignTool from command line.

2.1 Commands

The available commands are listed below:

- `--binary-image (-bin)`
  - Description: binary image file path (.bin extension)
  - Syntax: `-bin /home/User/binaryFile.bin`
- `--image-version (-iv)`
  - Description: Enters the image version of the signed image file.
  - Syntax: `-iv <version_number>`
- `--private-key (-prvk)`
  - Description: private key file path (.pem extension)
  - Syntax: `-prvk <private_key_file_path>`
  - Example: `-prvk ../privateKey.pem`
- `--public-key (-pubk)`
  - Description: Public key file path (.pem extension)
  - Syntax: `-pubk <public_key_file_path>`
  - Example: `-pubk C:\publicKey.pem`
- `--password (-pwd)`
  - Description: Password of the private key (this password must contain at least four characters)
  - Example: `-pwd azerty`
- `--load-address (-la)`
  - Description: image load address
  - Example: `-la <load_address>`
- `--entry-point (-ep)`
  - Description: image entry point
  - Example: `-ep <entry_point>`
- `--option-flags (-of)`
  - Description: image option flags (default value = 0)
  - Example: `-of <option_flags>`
- `--algorithm (-a)`
  - Description: Specifies one of the prime256v1 (value 1, default) or brainpoolP256t1 (value 2).
  - Example: `-a <2>`
- `--output (-o)`
  - Description: output file path. This parameter is optional. If not specified, the output file is generated at the same source file path (for example, the binary image file is *C:\BinaryFile.bin*). The signed binary file is *C:\BinaryFile_Signed.bin*.
  - Syntax: `-o <Output_File_Path>`
- `--type (-t)`
  - Description: binary type. Possible values are `ssbl`, `fsbl`, `teeh`, `teed`, `teex` and `copro`.
  - Syntax: `-t <type>`
- `--silent (-s)`
  - Description: no message displayed for replacing an existing output file

- `--help (-h and -?)`
  - Description: Shows help.
- `--version (-v)`
  - Description: Displays the tool version.

2.2 Examples

The following examples show how to use STM32MP1-SignTool:

- **Example 1**

```
-bin /home/User/BinaryFile.bin -pubk /home/user/publicKey.pem -prvk /home/user/privateKe
y.pem -iv 5 -pwd azerty -la 0x20000000 -ep 0x08000000
```

The default algorithm (prime256v1) is selected and the option flags value is 0 (default value). The signed output binary file (BinaryFile_Signed.bin) is created in the */home/user/* folder

- **Example 2**

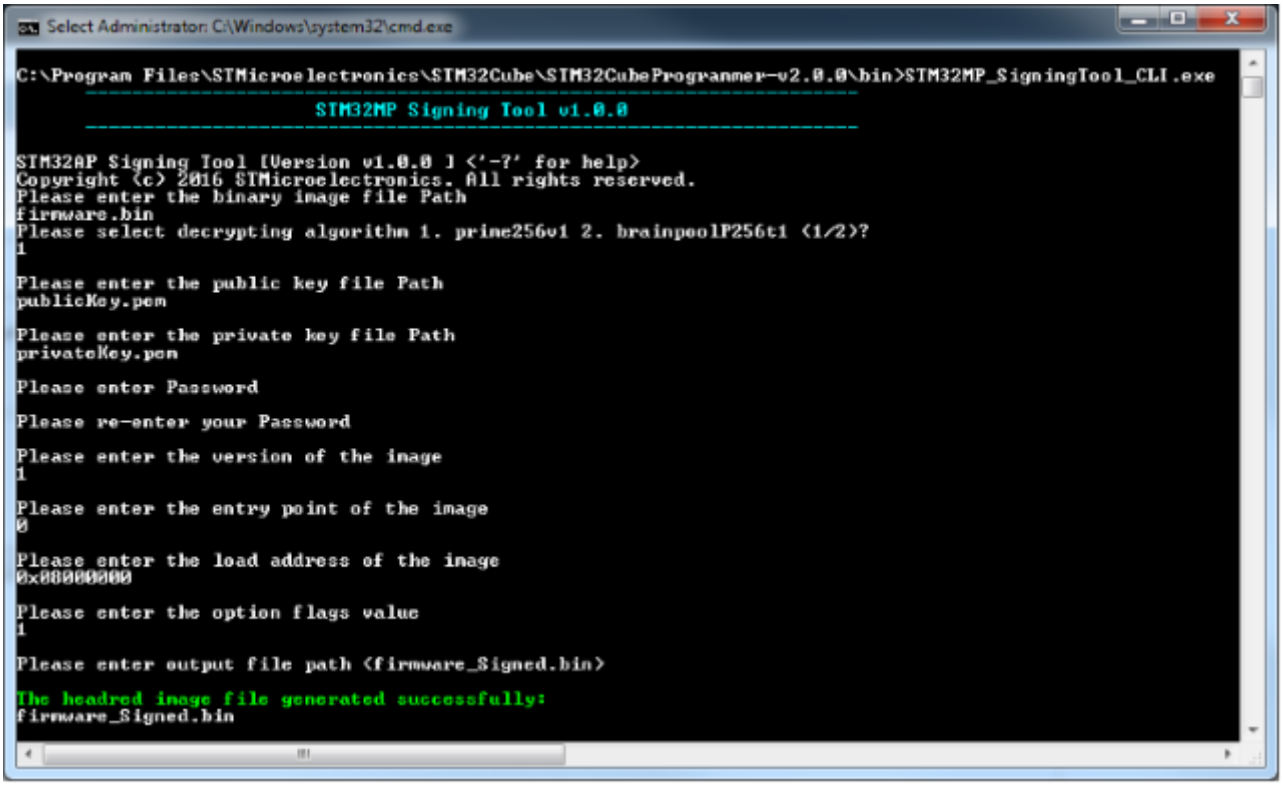
```
-bin /home/User/Folder1/BinaryFile.bin -pubk /home/user/publicKey.pem -prvk /home/user/p
rivateKey.pem -iv 5 -pwd azerty -s -la 0x20000000 -ep 0x08000000 -a 2 -o /home/user/Fold
er2/Folder3/signedFile.bin
```

The BrainpoolP256t1 algorithm is selected in this case. Even if Folder2 and Folder3 does not exist, they are created. With the `-s` command, even if a file exists with the same specified name, it is automatically replaced without any message.

2.3 Standalone mode

When executing STM32MP1-SignTool in Standalone mode, an absolute path must be entered first, then a password is requested twice for confirmation, as shown in the figure below.

Figure 1. STM32MP1-SignTool in Standalone mode



Next steps are the following:

- Select one of the two algorithms.
- Enter the image version, the image entry point and the image load address.
- Enter the option flags value.

Another output file path can be specified if needed or press enter to continue with the existing one.

Revision history

Table 1. **Document revision history**

Date	Version	Changes
14-Feb-2019	1	Initial release.

Contents