STM32 header for binary files

Contents

Stable: 24.03.2021 - 08:22 / Revision: 11.02.2021 - 20:15

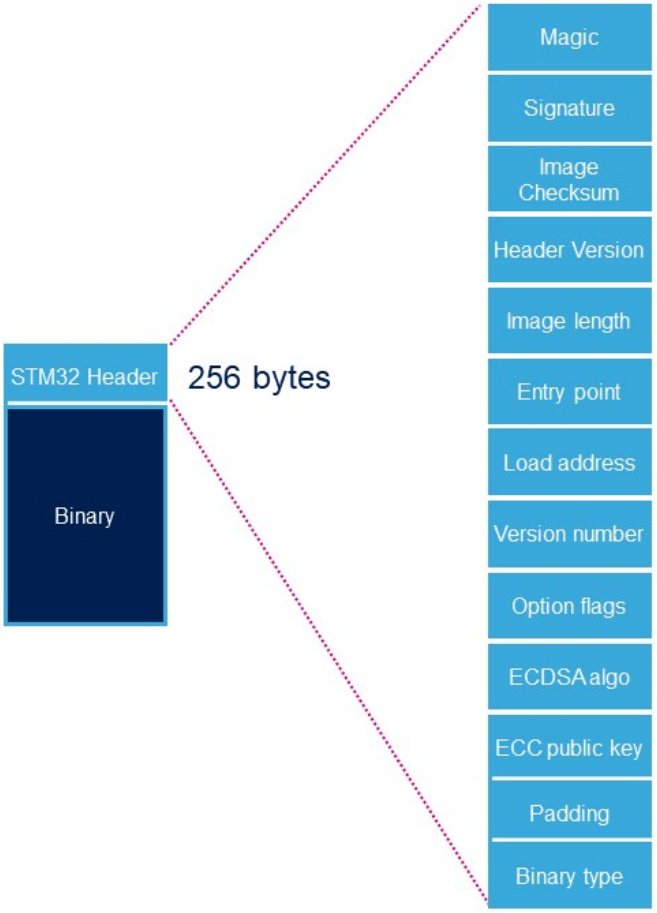
Stable: 24.03.2021 - 09:22 / Revision: 11.02.2021 - 20:13

A quality version of this page, accepted on *24 March 2021*, was based off this revision.

The STM32 header is a STMicroelectronics header needed for binaries loaded by ROM code.

Description

Each binary image (signed or not) loaded by ROM code need to include a specific STM32 header added on top of the binary data. The header includes the authentication information.



Name	Length	Byte Offset	Description
Magic number	32 bits	0	4 bytes in big endian: 'S', 'T', 'M', 0x32 = 0x53544D32
Image signature	512 bits	4	ECDSA signature for image authentication[Note 1]
Image checksum	32 bits	68	Checksum of the payload[Note 2]
Header version	32 bits	72	Header version v1.0 = 0x00010000 Byte0: reserved Byte1:major version = 0x01 Byte2: minor version = 0x00 Byte3: reserved
Image length	32 bits	76	Length of image in bytes[Note 3]
Image entry Point	32 bits	80	Entry point of image

Name	Length	Byte Offset	Description
Reserved1	32 bits	84	Reserved
Load address	32 bits	88	Load address of image[Note 4]
Reserved2	32 bits	92	Reserved
Version number	32 bits	96	Image Version (monotonic number)[Note 5]
Option flags	32 bits	100	b0=1: no signature verification[Note 6]
ECDSA algorithm	32 bits	104	1: P-256 NIST ; 2: brainpool 256
ECDSA public key	512 bits	108	ECDSA public key to be used to verify the signature.[Note 7]
Padding	83 Bytes	172	Reserved padding bytes[Note 8]. Must all be set to 0
Binary type	1 Byte	255	Used to check the binary type 0x10-0x1F: TF-A 0x30: Copro

- Signature is calculated from first byte of header version field to last byte of image given by image length field.
- 32-bit sum of all payload bytes accessed as 8-bit unsigned numbers, discarding any overflow bits. Used to check the downloaded image integrity when signature is not used (if b0=1 in Option flags).
- Length is the length of the built image, it does not include the length of the STM32 header.
- This field is not used by ROM code.
- Image **version number** is an anti rollback monotonic counter. The ROM code checks that it is higher or equal to the monotonic counter stored in OTP.
- Enabling signature verification is mandatory on secure closed chips.
- This field is an extract of PEM public key file that only kept the ECC Point coordinates *x* and *y* in a raw binary format (RFC 5480). This field will be hashed with SHA-256 and compared to the **Hash of pubKey** that is stored in OTP.
- This padding forces STM32 header size to 256 bytes (0x100).

The OTP used for the **Version number** and the **Hash of pubKey** are defined in the chapter “OTP configuration“ of the ROM code overview.

Elliptic Curve Digital Signature Algorithm

Trusted Firmware for Arm® Cortex®-A

Read Only Memory

One Time Programmed

Privacy Enhanced Mail (File format for storing and sending cryptographic keys, certificates, and other data)

Elliptic curve cryptography

Error Correction Capability

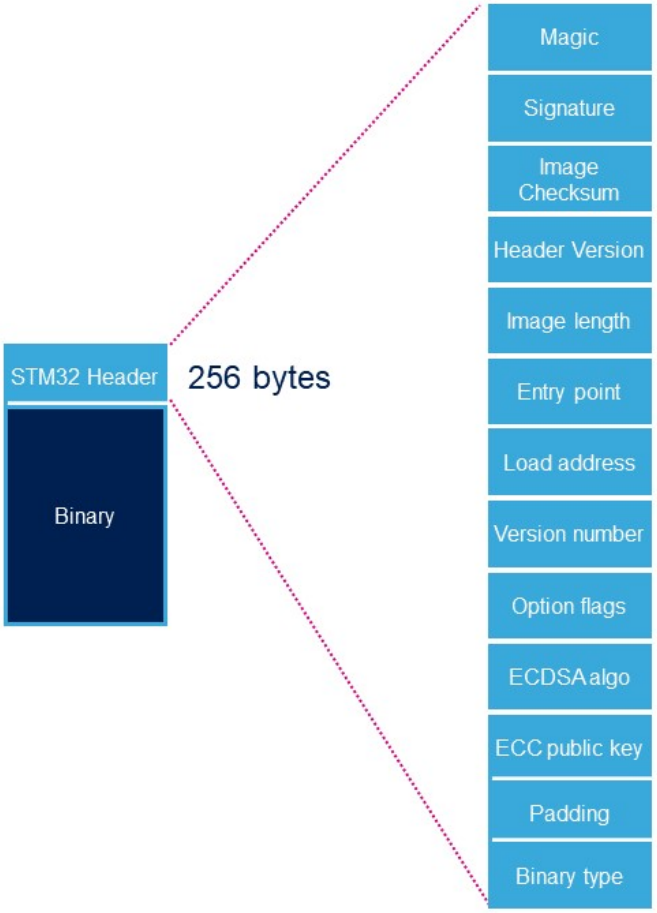
Secure Hash Algorithm

Stable: 17.06.2020 - 15:26 / Revision: 16.01.2020 - 09:28

The STM32 header is a STMicroelectronics header needed for binaries loaded by ROM code.

Description

Each binary image (signed or not) loaded by ROM code need to include a specific STM32 header added on top of the binary data. The header includes the authentication information.



Name	Length	Byte Offset	Description
Magic number	32 bits	0	4 bytes in big endian: 'S', 'T', 'M', 0x32 = 0x53544D32
Image signature	512 bits	4	ECDSA signature for image authentication[Note 1]
Image checksum	32 bits	68	Checksum of the payload[Note 2]
Header version	32 bits	72	Header version v1.0 = 0x00010000 Byte0: reserved Byte1:major version = 0x01 Byte2: minor version = 0x00 Byte3: reserved
Image length	32 bits	76	Length of image in bytes[Note 3]
Image entry Point	32 bits	80	Entry point of image

Name	Length	Byte Offset	Description
Reserved1	32 bits	84	Reserved
Load address	32 bits	88	Load address of image[Note 4]
Reserved2	32 bits	92	Reserved
Version number	32 bits	96	Image Version (monotonic number)[Note 5]
Option flags	32 bits	100	b0=1: no signature verification[Note 6]
ECDSA algorithm	32 bits	104	1: P-256 NIST ; 2: brainpool 256
ECDSA public key	512 bits	108	ECDSA public key to be used to verify the signature.[Note 7]
Padding	83 Bytes	172	Reserved padding bytes[Note 8]. Must all be set to 0
Binary type	1 Byte	255	Used to check the binary type 0x10-0x1F: TF-A 0x30: Copro

- Signature is calculated from first byte of header version field to last byte of image given by image length field.
- 32-bit sum of all payload bytes accessed as 8-bit unsigned numbers, discarding any overflow bits. Used to check the downloaded image integrity when signature is not used (if b0=1 in Option flags).
- Length is the length of the built image, it does not include the length of the STM32 header.
- This field is not used by ROM code.
- Image **version number** is an anti rollback monotonic counter. The ROM code checks that it is higher or equal to the monotonic counter stored in OTP.
- Enabling signature verification is mandatory on secure closed chips.
- This field is an extract of PEM public key file that only kept the ECC Point coordinates *x* and *y* in a raw binary format (RFC 5480). This field will be hashed with SHA-256 and compared to the **Hash of pubKey** that is stored in OTP.
- This padding forces STM32 header size to 256 bytes (0x100).

The OTP used for the **Version number** and the **Hash of pubKey** are defined in the chapter “OTP configuration“ of the ROM code overview.

Elliptic Curve Digital Signature Algorithm

Trusted Firmware for Arm® Cortex®-A

Read Only Memory

One Time Programmed

Privacy Enhanced Mail (File format for storing and sending cryptographic keys, certificates, and other data)

Elliptic curve cryptography

Error Correction Capability

Secure Hash Algorithm

Pages in category "ROM code"

The following 4 pages are in this category, out of 4 total.

0

- STM32 header for binary files
- STM32MP15 ROM code overview
- STM32MP15 ROM trace analyzer
- STM32MP15 ROM code secure boot