



# Code Signing Formats

A brief (always incomplete) overview

[tomas.gustavsson@primekey.com](mailto:tomas.gustavsson@primekey.com)



# Code Signing Purposes

- Software distribution
  - download software (manually/automagically)
  - verify integrity before installing
  - .msi, .deb, .rpm, .xpi, .pkg, .apk, .js, ...
- Software execution
  - software installed
  - verify integrity before executing
  - .exe, .jar, .js, Mach-O, ELF(?), UEFI, ...



# Code Signing Formats



# Pretty Good Privacy (PGP)

- Pretty Good Privacy (PGP/OpenPGP, GnuPG, ...)
- Different PKI (“web of trust”), not X.509 certificates
- Known for e-mail encryption and signing
- Detached signatures
- Signing of software
  - Release files/packages
  - Repositories (i.e. Red Hat, Debian, Maven Central, ...)



# Cryptographic Message Syntax (RFC5652)

- Around since early 1990's (PEM standards)
- Started as e-mail encryption and signing
- X.509 certificates
- Now used as security components of numerous standards and tools
- Signed/encrypted/sign-and-encrypted
- Sender and recipient information
- Signing of software
  - Release files/packages/executables (Authenticode, JAR, ...)



# OpenSSL

- Common for signing “anything”
- Can do CMS
- or just plain signatures
  - “RAW” private and public keys
  - Signing of software
    - Release files/packages (custom, firmware, ...)



# Binary Executables

- i.e. binary files is signed in order to execute, kernel modules, drivers, etc
- MacOS/iOS Mach-O, Authenticode, ...





# Code Signing Formats

TYPE
CMS
OpenSSL
RAW
PGP
XML



# Code Signing Formats

TYPE	USAGE
CMS	Authenticode, Java jar, Custom, FW, XAR (MacOS), ...
OpenSSL	Custom, FW
RAW	Custom, FW
PGP	deb, rpm
XML	SOAP(?)



# Code Signing Formats

TYPE	USAGE	CERTIFICATE
CMS	Authenticode, Java jar, Custom, FW, XAR (MacOS), ...	X.509
OpenSSL	Custom, FW	X.509/RAW
RAW	Custom, FW	X.509/PGP/RAW
PGP	deb, rpm	PGP
XML	SOAP(?)	X.509



# Code Signing Formats

TYPE	USAGE	CERTIFICATE	TSA
CMS	Authenticode, Java jar, Custom, FW, XAR (MacOS), ...	X.509	YES
OpenSSL	Custom, FW	X.509/RAW	NO
RAW	Custom, FW	X.509/PGP/RAW	NO
PGP	deb, rpm	PGP	NO
XML	SOAP(?)	X.509	YES



# Code Signing Formats

TYPE	USAGE	CERTIFICATE	TSA	EXPIRE
CMS	Authenticode, Java jar, Custom, FW, XAR (MacOS), ...	X.509	YES	YES
OpenSSL	Custom, FW	X.509/RAW	NO	NO
RAW	Custom, FW	X.509/PGP/RAW	NO	NO
PGP	deb, rpm	PGP	NO	YES
XML	SOAP(?)	X.509	YES	YES



# Code Signing Formats

TYPE	USAGE	CERTIFICATE	TSA	EXPIRE	RSA/EC	PQC
CMS	Authenticode, Java jar, Custom, FW, XAR (MacOS), ...	X.509	YES	YES	YES	YES
OpenSSL	Custom, FW	X.509/RAW	NO	NO	YES	?
RAW	Custom, FW	X.509/PGP/RAW	NO	NO	YES	?
PGP	deb, rpm	PGP	NO	YES	YES	?
XML	SOAP(?)	X.509	YES	YES	YES	?



# Code Signing Formats

PGP	CMS		XAR	OpenSSL
	Authenticode	JAR		
.deb	.msi	.jar	.pkg	custom
.rpm	.exe, .dll, .sys	.war	.img	...
mvn	.cab	.apk	...	
custom	.appx	.xpi		
...	.ps1	...		
	.xla			
	.js			
	...			



# Code Signing Details





# Signature formats

- “Plain signature”

```
$ openssl dgst sign privatekey.pem -out release.zip.sig -hex release.zip
```



845ad29d1725a..

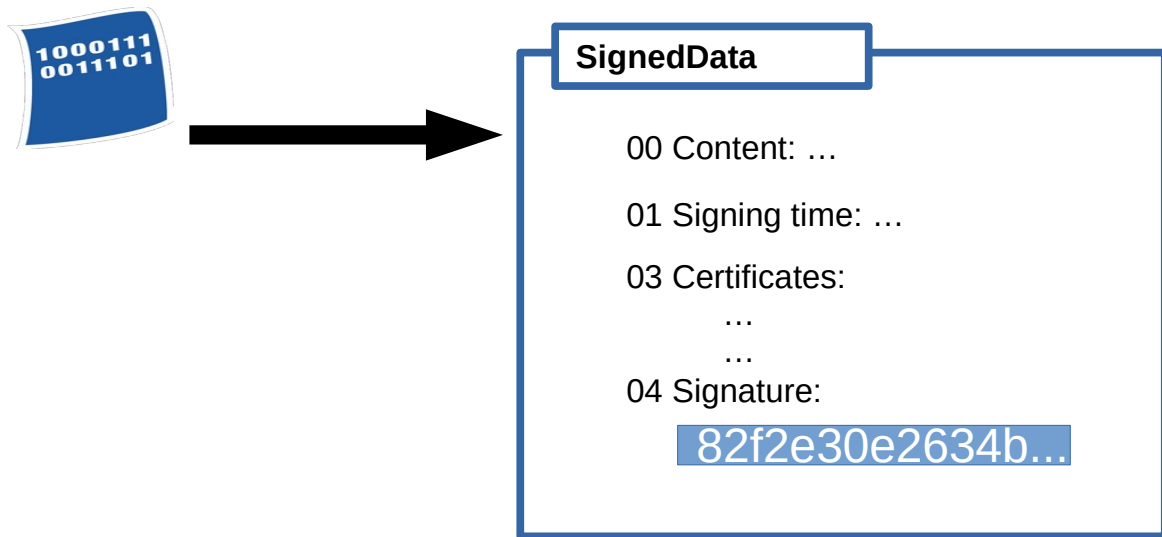
```
$ openssl dgst -verify publickey.pem -signature release.zip.sig release.zip
```

Receiver needs:

- Signature
- Original document
- Sender public key

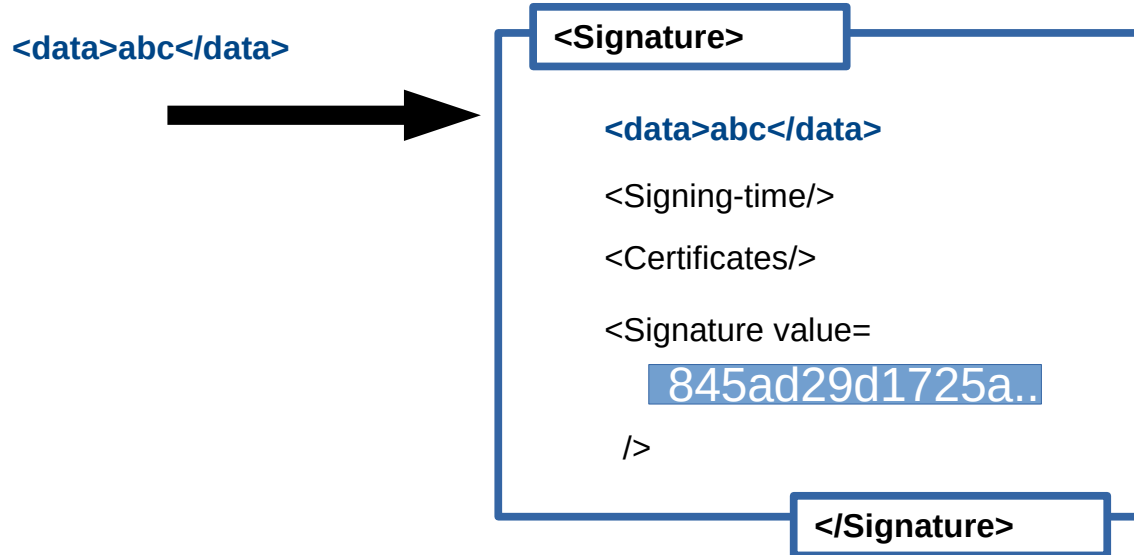
# Signature formats

- Cryptographic Message Syntax CMS (RFC5652)  
`$openssl cms ...`



# Signature formats

- XML Signature, XMLDSig



# Signing with PGP

```
$ gpg --sign -a -b -o release.zip.asc release.zip
```

```
$ cat release.zip.asc
```

```
-----BEGIN PGP SIGNATURE-----
```

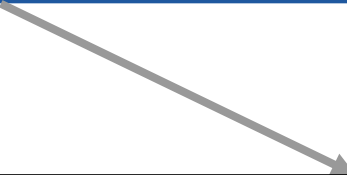
```
iQGcBAABCAAGBQJdJcSsAAoJE0d6BN8zdH9iKt  
RNHrba4EA1L2t2J60KfHiHdI10aKE0XfgJnEL3  
zNoJwuhDY72j2dW7A2K76jSygJlVM+Y4MGsK23  
XtaU7H5vDeEuMBPBvpgmZ+J3GXsr2sMAsPu629  
V07SJ6bqLIkf8PpEuU20qXLaR80IF0UJP1ehgy  
1Iy+Lity2rk1+8eKVT6YSbB81YxNfAkCbFK6yV  
7o4ru1PMqNzVEFV0peS2AjoWSXbrBLE/3ncY8v  
o/dEQ7g7dIMuPgt2ocE0TWLLBKswr19orN9FtB  
nv+qzL1ITqMP0lC0v/s7qfgoqS4aYBbnf4y2yU  
=lCJO
```

```
-----END PGP SIGNATURE-----
```



# Verifying PGP Signature

```
$ gpg --verify release.zip.asc release.zip
```



```
Signature made Tue 03 Sep 2019 11:10:30 AM CEST  
using RSA key EC1113F59E0D9D03  
  
Good signature from "Markus (Code Signing) <markus@primekey.se>"
```

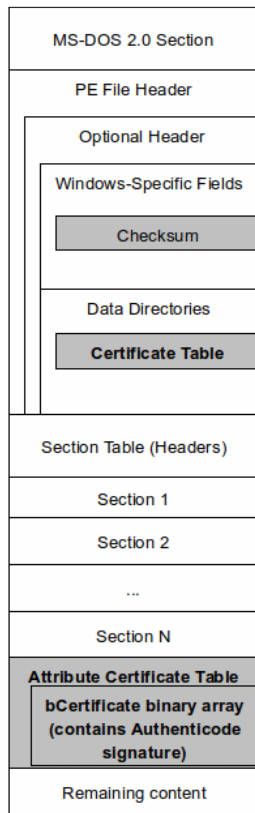


# Authenticode

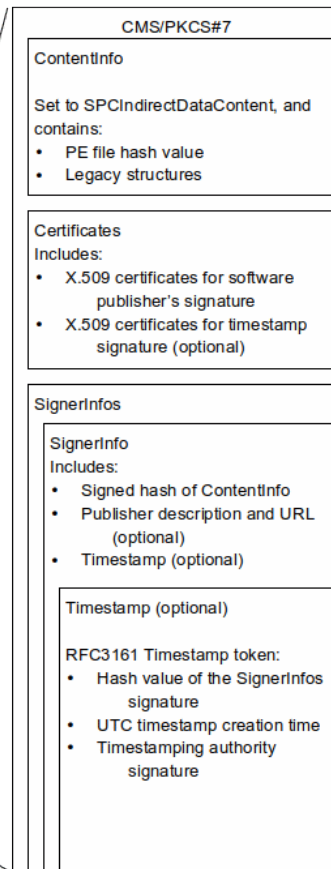


# Inside Authenticode

## Typical Windows PE File Format



## Authenticode Signature Format



Objects with gray background are omitted from the Authenticode hash value



# Java Code Signing



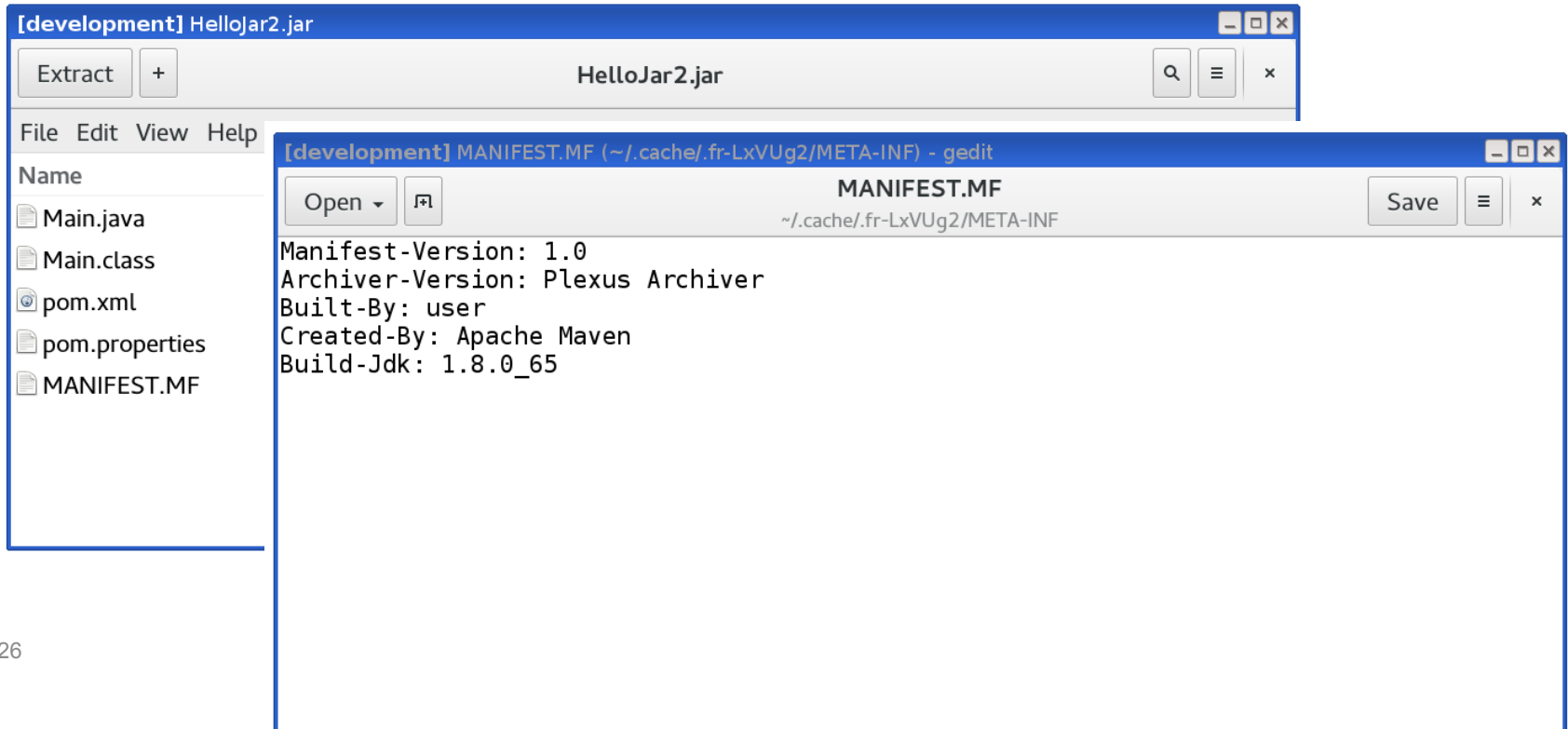


# Inside a JAR

Name	Size	Type	Modified	Location
Main.java	1.0 kB	Java so...	09 Mar...	/org/signserver/hellojar
Main.class	561 by...	Java cl...	09 Mar...	/org/signserver/hellojar
pom.xml	666 by...	XML do...	09 Mar...	/META-INF/maven/org.signserver/HelloJar
pom.properties	116 by...	unknown	09 Mar...	/META-INF/maven/org.signserver/HelloJar
MANIFEST.MF	123 by...	unknown	08 Mar...	/META-INF



# Inside a JAR manifest

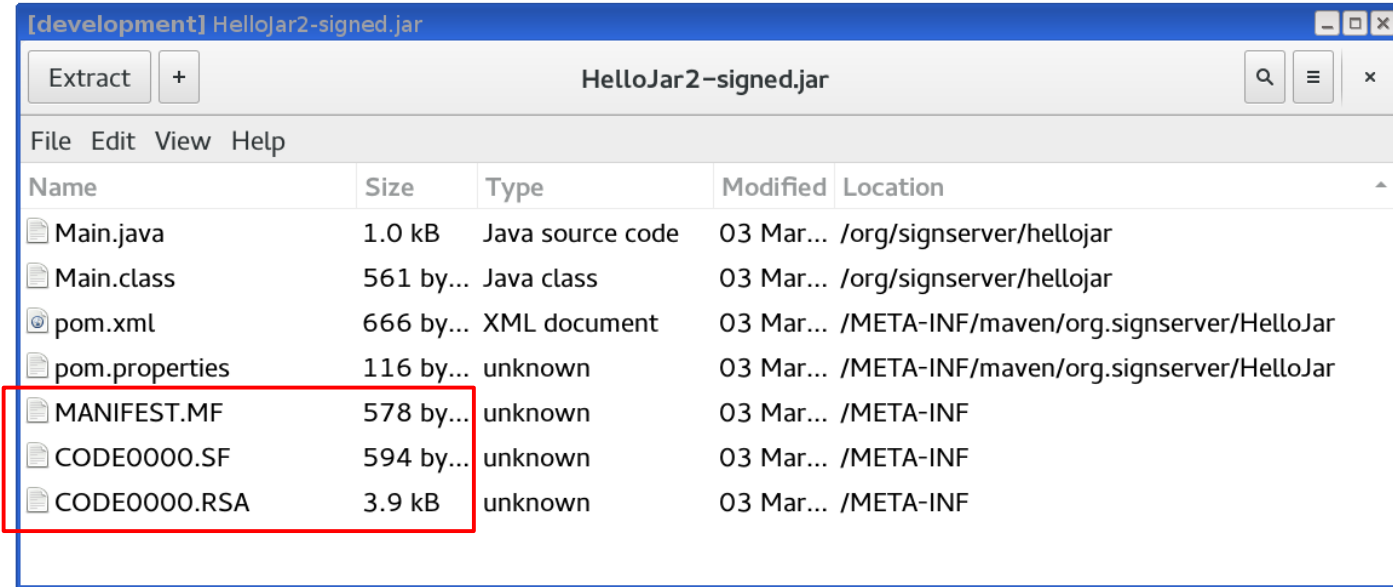


The image shows a screenshot of a file explorer window and a text editor window. The file explorer window, titled "[development] HelloJar2.jar", displays a list of files: Main.java, Main.class, pom.xml, pom.properties, and MANIFEST.MF. The text editor window, titled "[development] MANIFEST.MF (~/.cache/.fr-LxVUg2/META-INF) - gedit", shows the contents of the MANIFEST.MF file:

```
Manifest-Version: 1.0
Archiver-Version: Plexus Archiver
Built-By: user
Created-By: Apache Maven
Build-Jdk: 1.8.0_65
```



# Inside a signed JAR



File Edit View Help

Name	Size	Type	Modified	Location
Main.java	1.0 kB	Java source code	03 Mar...	/org/signserver/hellojar
Main.class	561 by...	Java class	03 Mar...	/org/signserver/hellojar
pom.xml	666 by...	XML document	03 Mar...	/META-INF/maven/org.signserver/HelloJar
pom.properties	116 by...	unknown	03 Mar...	/META-INF/maven/org.signserver/HelloJar
MANIFEST.MF	578 by...	unknown	03 Mar...	/META-INF
CODE0000.SF	594 by...	unknown	03 Mar...	/META-INF
CODE0000.RSA	3.9 kB	unknown	03 Mar...	/META-INF



# Signed JAR manifest

[development] HelloJar2-signed.jar

Extract + HelloJar2-signed.jar

File Edit View Help

Name

- Main.java
- Main.class
- pom.xml
- pom.properties
- MANIFEST.MF
- CODE0000.SF**
- CODE0000.RSA

[development] CODE0000.SF (~/.cache/fr-kkt3XD/META-INF) - gedit

Open CODE0000.SF Save

MANIFEST.MF CODE0000.SF

```
Signature-Version: 1.0
SHA-256-Digest-Manifest: mVsKD9aYxRotwLB3nEpH7piZfxB7P0zXsm/L007lkbU=
Created-By: SignServer EE 4.0.0-SNAPSHOT

Name: META-INF/maven/org.signserver/HelloJar/pom.properties
SHA-256-Digest: LqqW4SwepYEGvDwSMDKZeMnaq4qfc+7iE3BIQm4xwcc=

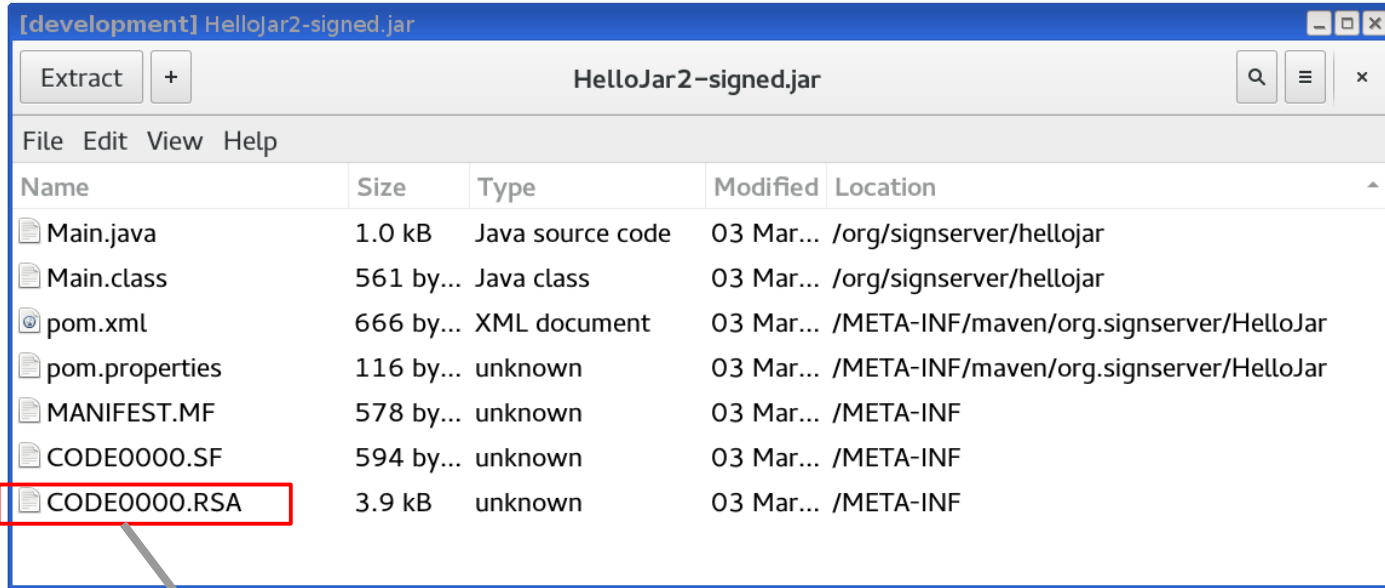
Name: org/signserver/hellojar/Main.java
SHA-256-Digest: mPaRTa8KoVtli77pNAjqhmfxijsAaXm8nxYbF6syowI=

Name: META-INF/maven/org.signserver/HelloJar/pom.xml
SHA-256-Digest: 3ho30WLX1wEX5RAckltJh5SEo4t6LZGe0JKIDn4ftaY=

Name: org/signserver/hellojar/Main.class
SHA-256-Digest: /mlS/ZdadiP2SmTXHSI 9d6SUaJaR5IGVHk3MhmOilaw=
```



# Signed JAR manifest



[development] HelloJar2-signed.jar

Extract + HelloJar2-signed.jar

File Edit View Help

Name	Size	Type	Modified	Location
Main.java	1.0 kB	Java source code	03 Mar...	/org/signserver/hellojar
Main.class	561 by...	Java class	03 Mar...	/org/signserver/hellojar
pom.xml	666 by...	XML document	03 Mar...	/META-INF/maven/org.signserver/HelloJar
pom.properties	116 by...	unknown	03 Mar...	/META-INF/maven/org.signserver/HelloJar
MANIFEST.MF	578 by...	unknown	03 Mar...	/META-INF
CODE0000.SF	594 by...	unknown	03 Mar...	/META-INF
CODE0000.RSA	3.9 kB	unknown	03 Mar...	/META-INF

```
$ openssl cms -verify -noverify -content META-INF/CODE0000.SF -in \  
META-INF/CODE0000.RSA -inform der
```



# How to sign code



# How to digitally sign code?

- Vendor provided command:
  - Microsoft: `signtool`
  - Java: `jarsigner`
  - Android: `jarsigner`, `SignApk`
  - Apple: `codesign`
  - Other/all: `gpg`, `openssl`
- Non vendor commands
  - `xar` + `openssl` for MacOS `pkg`
  - `openssl` for `jar`
  - ...
- Code signing service
- Own code
- etc



