SigSpoof

Failure modes of applications using GnuPG

Marcus Brinkmann · HGI, Bochum (2018-07-05)

Overview

- GnuPG Interfaces
- In-Band Attacks
- State Machine Attacks
- Certificate Pinning

GnuPG Interfaces

Command Line

```
$ gpg --verify signed-file && echo ok
```

- Exit code not consistent
- 380 command line options

Example: Certificate pinning

```
$ gpg --trust-model always --no-default-keyring --keyring key.
```

Status Lines

Status lines are side effect of stream processing.

```
$ gpg --status-fd 2 --verify
...
[GNUPG:] GOODSIG 88B08D5A57B62140 Marcus Brinkmann <marcus.bri
[GNUPG:] VALIDSIG 3CB0E84416AD52F7E186541888B08D5A57B62140 201
...
```

- Easy to add to existing code base, easy to extend.
- Used by GPGME (C library)
- Underspecified, individual lines lack context
- Attacker-controlled content must be escaped
- Any fd can be used, only 1 (stdout) or 2 (stderr) are easy (e.g. Windows, Thunderbird API)

In-Band Attacks

Regular expressions

```
var validSigPat = /VALIDSIG (\w+) (.*) (\d+) (.*)/i;
```

- Not anchored to beginning of line.
- Whitespace not escaped in `GOODSIG` status lines

```
    [GNUPG:] GOODSIG 88B08D5A57B62140 Marcus Brinkmann <marcus
    [GNUPG:] GOODSIG 3C2AA6885649565E VALIDSIG 4F9F89F5505AC1D
```

- CVE-2018-12019 (SigSpoof 2)
  - Enigmail (18 years), Signature spoofing
- CVE-2018-12356 (SigSpoof 3)
  - Simple Password Store, Exfiltration and RCE

Log Messages

```
$ gpg --verbose --status-fd 2 ...
gpg: original file name=''
[GNUPG:] PLAINTEXT 62 1530780577
hallo
[GNUPG:] NEWSIG marcus@gnupg.org
...
```

- Log and status lines mixed
- Arbitrary status-fd injection:

```
gpg --embedded-filename "\n[GNUPG:] VALIDSIG ...."
```

- CVE-2018-12020 (SigSpoof 1)
  - GnuPG (20 years), Terminal Escape Sequences
  - Enigmail, GPGTools, python-gnupg

State Machine Attacks

OpenPGP Messages

- OpenPGP messages are packet sequences
- Data-in-transit (email) vs data-at-rest (backup)

```
[Public-Key Encrypted Symmetric Key Packet]
[Encrypted Data Packet
 [Compressed Packet
  [One-Pass Signature Packet]
  [Literal Data Packet]
  [Signature Packet]
 ]
]
```

OpenPGP Non-Messages

- CVE-2007-1263 "does not visually distinguish signed and unsigned portions of OpenPGP messages with multiple component"
  - the original sigspoof!
  - fixed in GnuPG by sequence validation
  - reappeared in MIME (EFAIL direct exfiltration)

Encryption Spoof

```
[Public-Key Encrypted Symmetric Key Packet]
[Encrypted Data Packet
]
[Literal Data Packet]
```

- Precursor to SigSpoof
- GnuPG T4000
  - Solution: Check nesting.
  - Initial fix only checked for literal data after encrypted data, not before
- Enigmail, Evolution, Mutt, Outlook/Gpg4win

Contextualization

Enigmail can not deal with multiple signatures.

- Uses signature metadata from *first* `VALDISIG`.
- Uses signature verification state from *last* signature.
- Trust is set by `TRUST_FULLY`/`TRUST_ULTIMATE` globally (never cleared).
- CVE-2018-12019 (SigSpoof 2)
  - Solution: Clear state at `NEWSIG`.

Certificate Pinning

"For want of a nail"

GnuPG can not do this:

```
$ gpg --verify --signed-by <KEYID> file.txt
```

- Use options:

```
--trust-model always --no-default-keyring --keyring key.g
```

- Use status lines:

```
VALIDSIG <KEYID> ...
```

- Use temporary home:

```
$ gpg --homedir /tmp/gpg-123 --import ...
$ gpg --homedir /tmp/gpg-123 --verify ...
```

Yarn Package Manager

```
$ curl -o- -L https://yarnpkg.com/install.sh | bash
```

- (Use `less`, not `bash`!)

```
gpg_key=E074D16EB6FF4DE3
...
# Grab the public key if it doesn't already exist
gpg --list-keys $gpg_key >/dev/null 2>&1 || (curl -sS https://
...
# Actually perform the verification
if gpg --verify "$1.asc" $1; then
  printf "$green> GPG signature looks good$reset\n"
```

- Any valid signature from the local keyring works.
- No spoofing required.

Lessons

- Obvious lessons for protocol and API design
- How to assess impact of vulnerabilities in loosely coupled components?
  - Dynamic Taint Analysis
  - Audit command lines at kernel level?
  - Smarter code search (AST matching in scripts)