Sigsum

Using Sigsum Logs to Detect Malicious and Unintended Key-Usage

September 20, 2022

Rasmus Dahlberg
rgdd@glasklarteknik.se

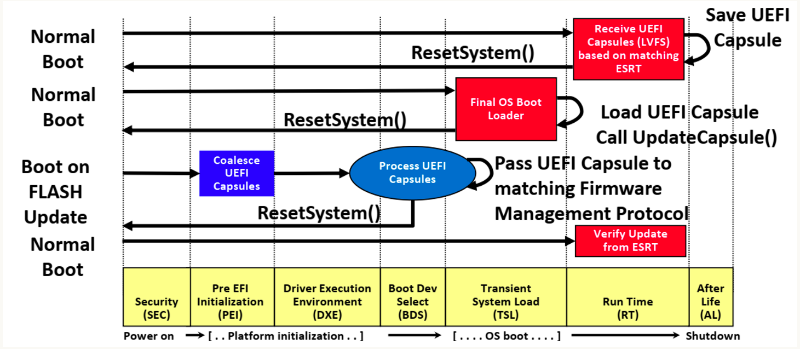
Outline

1. A weak link in firmware updates
2. How transparency logs can help
3. Meet the sigsum logging design

Disclamer

I’m not really a firmware hacker

UEFI firmware updates (1/2)

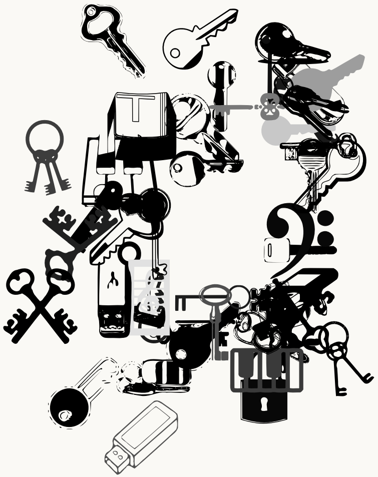


https://embeddedcomputing.com/technology/security/software-security/understanding-uefi-firmware-update-and-its-vital-role-in-keeping-computing-systems-secure

A problem of trust

UEFI firmware updates (2/2)

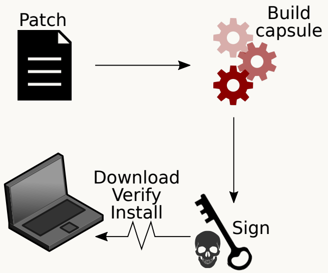
- Signed firmware updates
- Trust policy (public keys)
- Root of trust



The gist is that trusted keys sign firmware updates

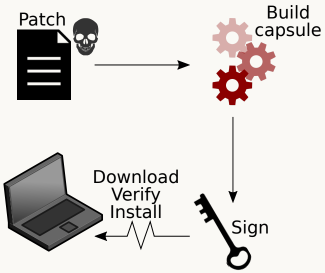
The zoomed out problem

Signer perspective



"Have I been owned?"

Verifier perspective



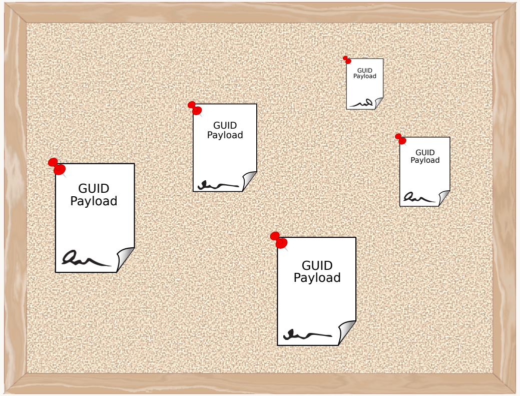
"Am I being targeted?"

It's hard to know which signatures are out there

Can we make signing keys less juicy targets for attackers?

yes.

A bulletin board for signed firmware updates



A transparency log is really just a tamper-evident append-only list

List

```
[ foo, bar, baz ]
```

Append-only

```
[ foo, bar, baz, qux ]
```

Tamper-evident (1/2)

```
[ cat, bar, baz, qux ]
```

Tamper-evident (2/2)

```
[ bar, baz, qux ]
```

Cryptographically verifiable

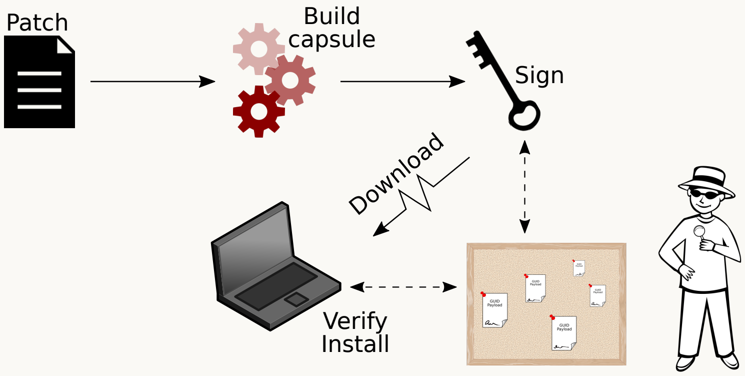
Merkle tree

```
tree head -->     .-  g:=H(e + f) -.           www.rgdd.se/r/tlog-0
                 /                  \
                /                    \
           e:=H(a + b)          f:=H(c + d)
            /      \             /      \
           /        \           /        \
       a:=H("foo") b:=H("bar") c:=H("baz") d:=H("qux")     <-- leaves
              ^           ^           ^           ^
              |           |           |           |
list items:  foo         bar         baz         qux
```

Inclusion proof

Append-only proof

A complete overview



No signed firmware goes unnoticed

Details of course matter in practise

Who
What
where
NOT
why
Logging
Downtime
Latency
poison
spam
complexity
Gossip
privacy
reactive
proactive
verification
m-of-n
API
sharding
ed25519
json
x509
SCTs
ssh

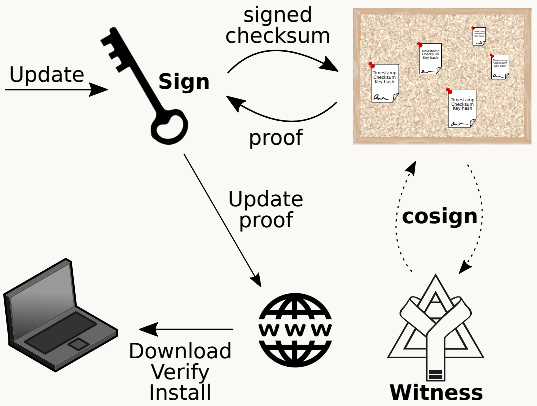
Sigsum is about nailing the details for a particular setting

What is Sigsum?

- A transparency log design
- A transparency log API
- A FOSS software project

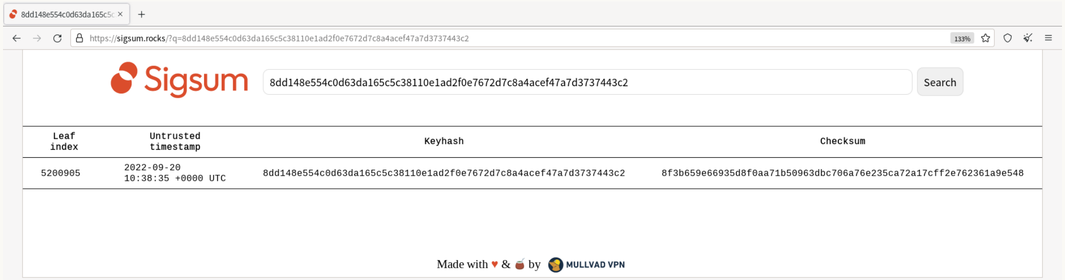
What you get from Sigsum

- Logging of signed checksums
- Centrally operated logs
- Distributed trust (m-of-n)
- Offline verification



Threat model: attacker runs everything but m witnesses you choose

I might have sigsum-logged these slides...



8dd148e554c0d63da165c5c
https://sigsum.rocks/?q=8dd148e554c0d63da165c5c38110e1ad2f0e7672d7c8a4acef47a7d3737443c2
133%
Sigsum
8dd148e554c0d63da165c5c38110e1ad2f0e7672d7c8a4acef47a7d3737443c2
Search

Leaf index	Untrusted timestamp	Keyhash	Checksum
5200905	2022-09-20 10:38:35 +0000 UTC	8dd148e554c0d63da165c5c38110e1ad2f0e7672d7c8a4acef47a7d3737443c2	8f3b659e66935d8f0aa71b50963dbc706a76e235ca72a17cff2e762361a9e548

Made with ♥ & by MULLVAD VPN

168c1008d0208bb6bcb73e34a15b98526ee50c1a1966141b23d342584ffaf5f7

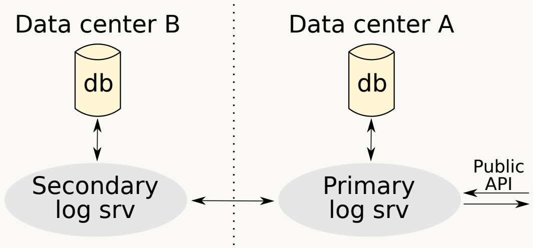
https://gitlab.glasklarteknik.se/rgdd/osfc-22/-/blob/main/releases/

What you don’t get with Sigsum

- Real-time logging latency
- Cryptographic agility
- Complicated parsers
- Poisoned logs
- Spammed logs

Current status

- Pretty stable foundation
- V0 design and API documents
- Log aimed for self-hosting
- Prototype witness and monitor
- Debug tool for log interations



Ongoing work: bump version to v1, cut a log release, better tooling

Take away

- Firmware signing keys are juicy targets
- Transparency logs add detection and deterrence
- Sigsum's trade-offs are promising for firmware
  - Avoids non-essential complexity
  - Offline verification
  - Everything but m-of-n witnesses are broken

Thank you

- GitLab: `https://www.sigsum.org/r/src`
- Design document: `https://www.sigsum.org/r/design`
- API specification: `https://www.sigsum.org/r/api`
- Speaker: `https://www.rgdd.se`
- Slides: `https://www.rgdd.se/r/osfc-22`

Contact: `#sigsum` at OFTC.net and Matrix, `https://lists.sigsum.org`