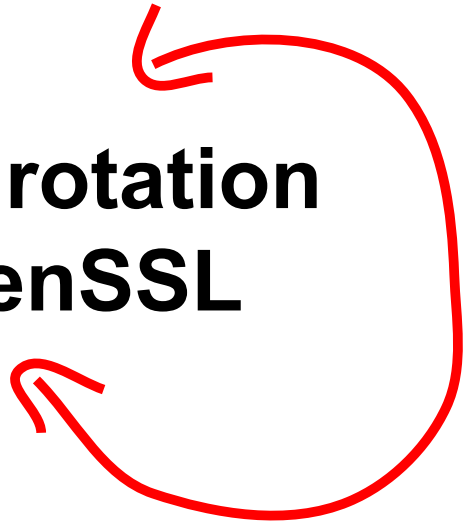
shim in 2026: A certificate rotation and a decade-overdue OpenSSL migration



~~Brian “Bex” Exelbierd, Microsoft~~
~~Nicolas Frayer,~~ Red Hat
Marta Lewandowska

shim in 2026: A decade-overdue OpenSSL migration and a certificate rotation

Marta Lewandowska, Red Hat

Nicolas Frayer
Senior Software Engineer, Red Hat

Brian “Bex” Exelbierd
Product Manager for Azure, Microsoft

the shim

- a “trivial” first-stage UEFI boot loader
- a compatibility layer between firmware and the OS
- the thing that makes Secure Boot on Linux work out of the box
- the thing that does the crypto

the update

23+ YEARS
OF SECURE CONNECTIONS
0.9.1 23 December 1998
0.9.2 22 March 1999
0.9.3 25 May 1999
0.9.4 9 August 1999
0.9.5 28 February 2000
0.9.6 24 September 2000
0.9.7 31 December 2002
0.9.8 5 July 2005
1.0.0 29 March 2010
1.0.2 22 January 2015
1.1.1 11 September 2018
3.0 7 September 2021
3.1 14 March 2023
3.2 23 November 2023
3.3 10 April 2024
3.4 22 October 2024
3.5 8 April 2025

the good

- providers code was easy to implement because it was all new
- once everything was merged, getting it to build didn’t take too long
- once it was building, getting it to boot didn’t take too long

the bad

- extremely time consuming
- shim binary went from 1 MB → 3 MB
- attack surface increased

the incompatible

- working around certain code assumptions
  - OS, userspace
  - filesystems
  - libc
- implementing missing functions
  - pre-boot environment
  - calls to EFI functions

the end (of part I)

- shim's openssl version went from 1.0.2k → 3.5.x
- shim is able to verify ML-DSA signed artifacts
- shim 17 release planned for later this year

A chain of trust, not one certificate

Firmware

Trusts signing authorities

shim

Signed for firmware to load

Distribution

Trusts its own next stage

OS

Continues the boot chain

The certificate had a schedule, everything else didn't always cooperate

2011

Original certificate Family

2023

New family; hardware tests

2025

Linux and firmware-update work

2026

2011 family expires

Firmware exposed the real failure modes

Update path

- Falsely reports success
- Exhausts firmware storage
- OEM update keys unavailable

Boot path

- Update lands, next boot fails
- Hands-on recovery may be needed
- Some systems become unusable

No one controls the whole rollout

Signing and review

- shim community: review
- Distributions: submit and ship shim
- Microsoft: validate submitters, sign artifacts

Firmware and deployment

- OEMs: firmware behavior and updates
- Cloud platforms: their own fleets
- Operators: when updates land

One fleet, three trust states

Firmware trusts	2011-signed shim	2023-signed shim	Assembled shim
2011 family only	Boots	Does not boot	Boots
Both families	Boots	Boots	Boots
2023 family only	Does not boot	Boots	Boots

Expiry does not revoke trust. An assembled shim carries both signatures.

Expiry is not revocation

- Firmware can still trust the 2011 family
- A compromised artifact must be blocked explicitly
- Old-only systems still need revocation updates
- New 2011-signed updates are no longer possible

Preparing for revocations we cannot sign later

Before expiry

Prepare signed individual revocations

Hold

Keep unreleased until needed

Release

Use a prepared update when required

Deliver

Apply multiple updates additively

Firmware and shim updates run on different clocks

Windows

- Can coordinate boot software and firmware updates
- Knows more about the managed update path

Linux

- Firmware often has a separate update path
- A shim update may not know next boot's trust state

Where the certificate side landed

- 2023 CAs are in use
- The 2011 family is expiring, not being revoked as a family
- Operators and distributions choose their schedules
- Revocation has a path; reliable delivery still needs work