Software Engineering and OpenSSL is not an oxymoron

Rich Salz
Akamai Technologies
OpenSSL Dev Team
rsalz@{akamai.com,openssl.org}
Main lesson

It’s not the crypto that kills you (or your open source project)
Historical Era’s

• SSLeay
• OpenSSL
• The CVE that Must Not Be Named
• Recovery
• Today and tomorrow
SSLLeay

• Various creation legends:
  – Let’s put on a show (Eric: “I’ve got DES and a BIGNUM package” Tim: “Let’s do SSL”)
  – Two guys in a garage

• Small set of folks sending patches
• Handful of tests; minimal functionality
The Rise of OpenSSL

- Started off big; as many as a dozen members
- Export control: stay away from the US
- Active mailing lists, still took patches
- “Interop with OpenSSL more important than what the RFC says”
The Fall of OpenSSL

• Project had become moribund
• Releases were not pre-announced, no documented policies
• Source code was complex and arcane
• Hard to maintain; harder to contribute
• Main developers were overworked and overcommitted
• Project donations minimal (sub USD$2000 per annum)
The Picture of Stasis

Apr 4, 2012 – Apr 4, 2014
Contributions to master, excluding merge commits

Contributions: Commits

snhenson
448 commits / 24,181 ++ / 7,369 --

#1

dot-asm
340 commits / 50,880 ++ / 11,605 --

#2
Why the fall?

• Long learning cycle to understand code
• Need to get consulting dollars (FIPS) to keep project alive
• Very little time spent on building community
• No ability to make, announce, and keep to plans

• ... all added up to “stay dark” attitude
The CVE That Must Not Be Named

- CVE-2014-0160, April 3
Recovery

• New blood (enthusiasm) on the team
  – CII created, funds two
  – Donations jump, funds two

• We met in Oct 2014:
  – Wrote release, security policies
  – Coding Style (!!!)
  – Socialized; POODLE helped
Going to Meetings is part of Recovery

• We also met in Oct 2016:
  – CVE notification process
  – CII/LF discussions (about and with)
  – How to grow the team
  – How to get more testing
  – Update roadmap and platform doc
  – Regular release cadence
Recovery: Transparency

• **Building community is job 1**
  – Documented what we want to do, and how.
  – Website overhauled (still too wordy)
  – Mailing lists moved
  – RT sped up (multiple moderators), and then removed

• Virtuous cycle: when a project isn’t a black hole, people contribute
Recovery: Code Quality

- Appearances count
  - Almost-repeatable code reformatting
- Mandatory review by a second team member
  - We’re still improving this
- More tests: Coveralls reports 57% of lines
- Modern practices: fuzzing, CI, etc.
  - Remember, OpenSSL is old
No longer a dumping ground

- Removed dozens of old platforms we could not test (Duo-culture is useful)
- Removed old and/or weak cipher suites
- GOST moved to external ENGINE
- Related: most structures are opaque, for future-proof (API/ABI compatibility; did hamper us before)
Recovery: Test Coverage
Zooming in
2016 Project Activity

- 3889 commits
- 431 GitHub users; thousands of forks
- 250 new issues
- 107 new pull requests; 1052 PR’s closed
- Releases:
  - 1.1.0 a-c
  - 1.0.2 a-j
  - 1.0.1 h-u EOL
2016 CVE’s

• 9 high (force a release)
• 20 medium (might force a release)
• 28 low (just fix)

Mostly met the disclosure/fix deadlines

Thankfully no critical yet
GitHub: Current activity

Contributions to master, excluding merge commits

1. mattcaswell: 1,309 commits / 430,109 ++ / 400,635 --
2. levitte: 1,315 commits / 131,315 ++ / 136,985 --
3. snhenson: 953 commits / 47,420 ++ / 77,548 --
4. richsalz: 619 commits / 149,926 ++ / 274,180 --
Today and Tomorrow: Excelsior

• Everything* is done on GitHub now
• Everyone has a CLA
• Major infrastructure components (technical debt) being addressed:
  – Threads, state machine, TLS packet formats
  – CLI flags, help improved
  – All docs are improved
What’s coming?

• FIPS work funded, but on-hold for TLS 1.3. Likely to mean ENGINE extensions.
  – might mean putting “old crypto” into an ENGINE
  – Tension between “safe” crypto and “everyone’s crypto”

• TLS 1.3
  – Contract in place with fixed delivery date and known interoperability

• Licensing
  – Moving to APLv2

• Testing
  – More and more and more and more
  – Can already run boringSSL test suite, e.g.
What Might/Should come

• All SSL public functions documented (101 missing out of 402)
• Need to fix the RNG, *portably*
• A generic STORE facility, for PKI objects.