Experience and insights of an OpenSSL Committer – a peek behind the scenes

Open Source 2024 @ Siemens in Zug, Switzerland, May 15th

David von Oheimb, Siemens Technology, IT Security, PKI team

SIEMENS

My first encounter with open-source software

Open-source software back in 1985:
Source code (in BASIC) and
binary code (8-bit machine code) for Sharp pocket computer PC-1401
distributed on paper in a dedicated magazine
to be manually typed in

No IT network whatsoever, no security concerns ☺

My first IT products were for Sharp PC-1600:
disassembler, macro assembler, debugger,
and word processor,
all in Z80 machine code – so “**no-source SW**” 😉
shipped on 2.5” floppy discs

Started programming in C around 1988 on a Commodore Amiga

SIEMENS

A little history of OpenSSL

Since 1995: open-source library SSLeay by Eric A. Young and Tim Hudson
for securing HTTP connections of web server Apache, containing `libssl` and `libcrypto`

OpenSSL forked from SSLeay, first released on 23rd December 1998
Planned name was “OpenTLS”, but the TLS RFC was not yet published.

OpenSSL
Cryptography and SSL/TLS Toolkit

```
commit 651d0aff98d28e2db146afa1790e9e22f3ef22db
Author: Ralf S. Engelschall <rse@openssl.org>
Date:   Tue Dec 22 15:04:48 1998 +0000

    Various cleanups and fixed by Marc and Ralf to start the OpenTLS project
```

```
commit 5f32680329648886701f5b5832239eecf0b38390
Author: Ralf S. Engelschall <rse@openssl.org>
Date:   Wed Dec 23 07:53:55 1998 +0000

    Switch version string to SSLeay/OpenSSL
```

Due to export control restrictions in 1990s, OpenSSL is (mostly) non-US product!
For 15 years, development by a loose team; major constant was Stephen Henson

In 2014, most infamous security incident: Heartbleed
Wonderful article: “The Internet Is Being Protected By Two Guys Named Steve”

Since then, more stable funding, larger team with full-time employees, formal procedures

SIEMENS

OpenSSL build tooling and code base

Tools: C compiler+linker (`gcc`/`clang`/MSVC, …), `(n)make`, `nasm`, `perl`, `pod2man`
For portability, uses no CMake, `bash`, …

Perl used for generating `Makefile`, code, and test data,
test drivers and test scripts, coding style checking (added by me)

External systems used: GitHub, Coverity

Code directories:	# `.c` files	# LoC	# chars	comment
`crypto/`	817	318,737	11,106,687	includes much more than crypto!
`ssl/`	94	94,818	3,013,465	these days of course (d)TLS, +QUIC
`{apps,demos}/`	142	66,812	2,131,659	mostly command-line
`{providers,engines}/`	213	67,361	2,225,335	crypto algs being moved here
`{test,fuzz}/`	390	148,855	4,984,967	plus many test Perl scripts

SIEMENS

OpenSSL versioning, license, challenges

Version scheme:

Since 3.0 (of Sep 2021): **semantic versioning** – next API breaking changes impossible before 4.0
Since 3.1 (of Mar 2023): minor versions every 6 months
partly with long-term support (3.0 has LTS), so backporting needed

SW license:

Until 1.1.1: proprietary OpenSSL license, incompatible with GPLv2
Since 3.0: Apache License 2.0 (more common, more liberal)

Major development hurdles:

Inherent complexity, partly low-level code, ancient critical code (e.g., X.509 verification)
Legacy features, too broad API, API&ABI compatibility – partial remedy: deprecation
Technical debt also w.r.t. structuring: intermingled code, redundancies
Shortcomings of C: manual memory management, `int` vs. `bool`, etc.

SIEMENS

Why use OpenSSL, also at Siemens?

OpenSSL – still – is most commonly used crypto/TLS library and tool.

- ⊕ FOSS, easily available, reliably maintained, pretty secure
- ⊕ most well-known, most experience available, fits with C-based device software
- ⊕ feature-rich: all sorts of crypto, RNG, (D)TLS, HTTP, QUIC, X.509, ASN.1, OCSP, CMS/PKCS#7, BIOs
- ⊕ since 3.0: future-proof by flexible API, open platform for external, also HW-based crypto: providers
- ⊖ Rather bulky, hard to use
- ⊖ Memory-safety hard to achieve

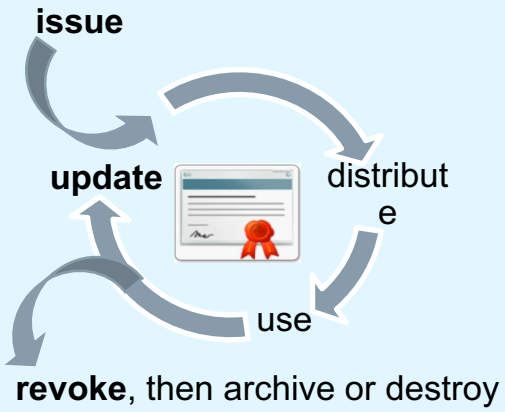
Competitors:

- small footprint: mbedTLS, wolfSSL, …
- lean design: libreSSL, BoringSSL, GnuTLS, …
- other languages: Rustls, Bouncy Castle (Java/C#), …

SIEMENS

Adding CMP to OpenSSL for use with Siemens Product Public-Key Infrastructure (PKI)

- Secure communication needs **public-key certificates**: “digital passports” binding public key to identity
- OpenSSL has been **supporting X.509 certificate** generation and use in validation, but **no management protocol** for requesting and revoking then.
- Certificate management/enrollment protocols: CMP, CMC, SCEP, EST, Let’s Encrypt



- In 2007, Martin Peylo at NSN/**Nokia developed** [CMPforOpenSSL]
  for use in LTE, where CMP is required by the standard.
  Tried incorporating the patch with OpenSSL during 2013 to 2015.
  One of the issues: using libcURL for poor HTTP support in OpenSSL

NOKIA
Connecting People

- In 2014/2015, we tried using EST and contributing to Cisco’s libEST, which did not work out
- In 2015, Siemens Product PKI switched to **CMP** as it is **most flexible and secure**: transport-independent, supports end-to-end authentication, good basis for post-quantum crypto
- I started contributing to CMPforOpenSSL, based on BU requirements

SIEMENS

SIEMENS

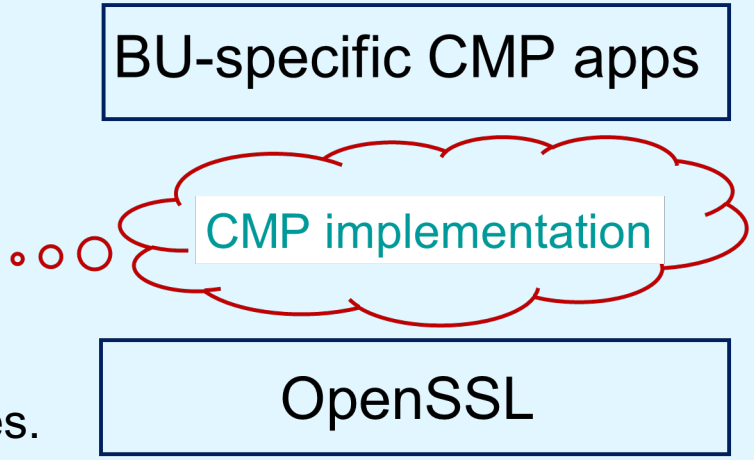
Which way to provide CMP with OpenSSL?

As a patch: CMPforOpenSSL code patched deeply into OpenSSL code

- **High code maintenance effort** to follow evolution of OpenSSL
- **Patching is not suitable for automated build processes** of Siemens products

As a standalone library based on vanilla OpenSSL

- Hard to achieve due to API restrictions und missing support features
- Still would require long-term maintenance by Siemens



Push upstream to OpenSSL:

- **Push CMPforOpenSSL upstream to OpenSSL project**, **minimizing long-term maintenance effort** and facilitating build processes.
- Provide an **interim library** ready to use with current OpenSSL

- Approach **funded jointly** by Siemens business units since 2017

SIEMENS

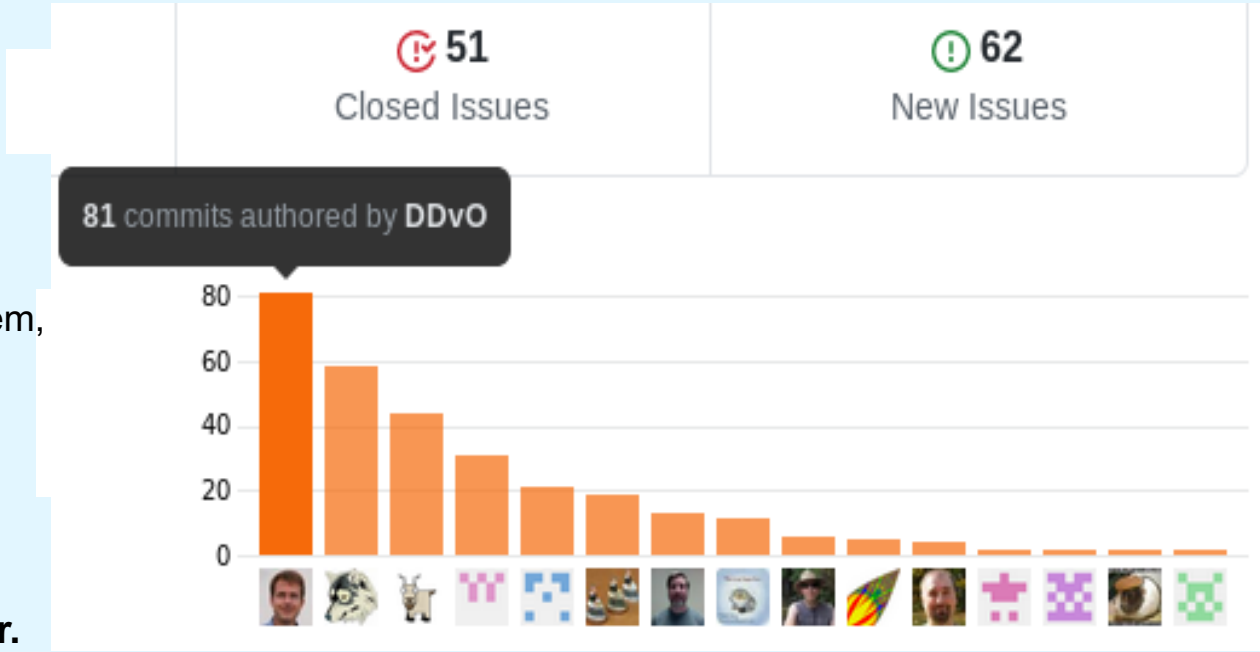
CMP upstream contribution to OpenSSL 3.0 required huge amount of work, technical expertise, and personal dedication

- **In early 2018**, we sought official OpenSSL management buy-in.
- **In August 2018, OpenSSL stated support for the contribution.** Target: version 3.0; feature freeze was planned for end-2019.
- For the CMPforOpenSSL upstream contribution, had to
  - re-structure ~16.000 LoC and slice it into 12 incremental chunks,
  - add many automated tests and extensive documentation,
  - submit pull requests and wait for OpenSSL members to review them,
  - react on feedback, adapting coding style, API usage, etc.
  - **fix bugs and omissions also within OpenSSL** itself.

  In total, I (username: DDvO) authored 165+ pull requests under pressure of the approaching code freeze deadline.
- **In January 2020, I was invited to become OpenSSL Committer.**
  - Got direct access to Git repository and can take part in reviewing and approval of pull requests.
  - This **boosted the throughput of Siemens contributions for two years**.

➔ **In summer 2020, the last CMP contribution chunk was merged, in time for version 3.0**.

Number of commits merged during Aug 11 to Sep 11, 2020:



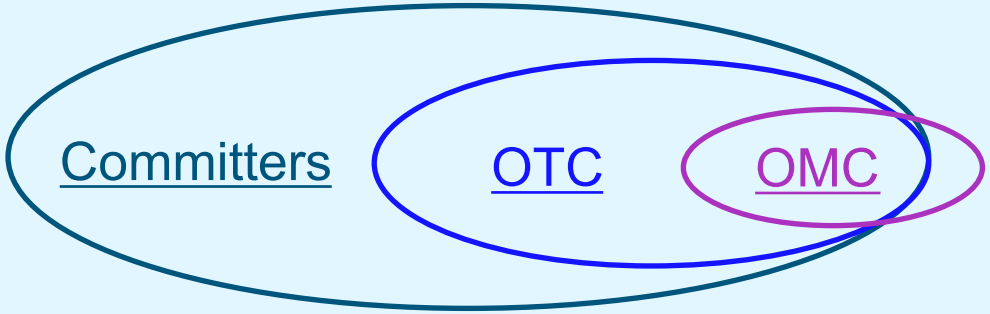
SIEMENS

OpenSSL Committers, Technical and Management Committees, and bylaws

OpenSSL Committers: currently 18 people who can **add commits** to the main OpenSSL project repository. Collectively, they have the **responsibility for maintaining** the contents of that repository.
They have a responsibility to **review code submissions** in accordance with OpenSSL policies and procedures.

OpenSSL Technical Committee (OTC): currently 10 OpenSSL committers being the technical voice of the project. The OTC **makes all technical decisions**, based on votes, of the code and documentation for OpenSSL.

OpenSSL Management Committee (OMC): currently 5 people representing the official voice of the project. The OMC makes all decisions, based on votes, regarding **management and strategic direction** of the project.



Commit access is granted by invitation from the OTC after OMC decision and may be withdrawn by the OMC. Minimal activity required **for keeping committer status is one commit authored or reviewed in 2 quarters**.

SIEMENS

OpenSSL team culture

Collaboration between OTC+OMC members: rather intense, weekly online meetings
6 people are full-time employees of OpenSSL Software Services and do about 2/3 of the (non-trivial) commits
Other Committers like me: sporadic contact with the core team

People being (at least partly) paid by their companies, contribute under CCLA about ¼ of the commits.
Remaining commits contributed under Individual Contributor License Agreement (ICLA).

Very rare face-to-face OTC and Committer meetings – last one was in June 2023 in Brno, CZ.
Great for getting to know people and for technical & strategic alignment.

We have a
Signal group for
informal exchange:
“OpenSSL friends”

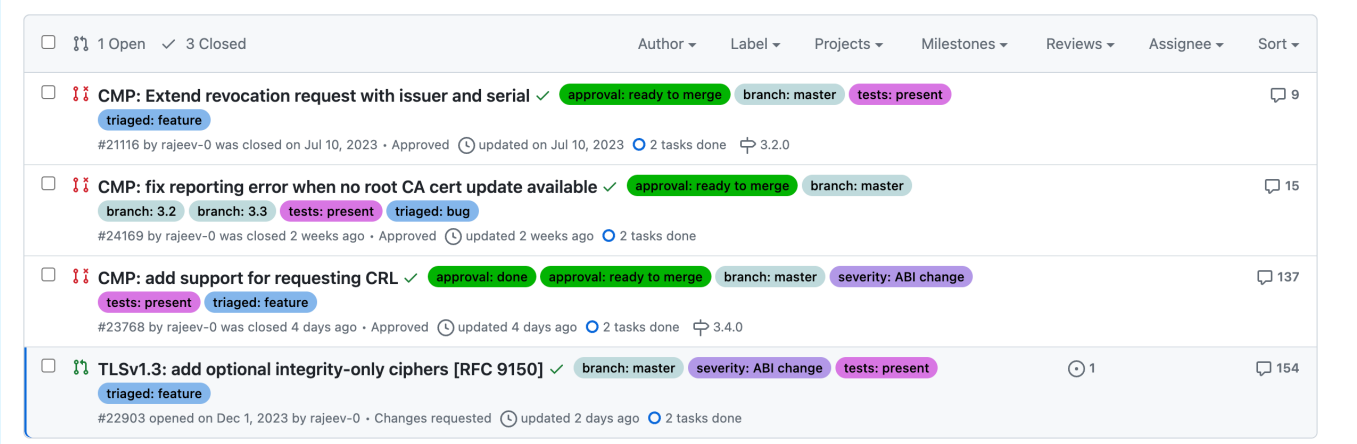
SIEMENS

Struggle bringing new features into OpenSSL, due to slow reviewing and cautious policies

Siemens PKI team main goal: **bring** features and fixes on **certificate management into OpenSSL**
Bulk of CMP already contributed by summer 2020, but few new CMP features still in the pipeline.
Reviewing is generally laggy and CMP is not in OpenSSL focus.
All pull requests (PR) must be **reviewed and approved by at least one committer and one OTC member**.

In total, OpenSSL currently has

- *~ 330 open PRs, 13600+ closed,*
- *~1950 open issues, 7600+ closed.*



Since February 2022, neither of the reviewers can be the author of the submission. **Quality & security plus**, but:
*This has **significantly slowed down our contributions** because whenever PRs contain anything by myself, my approval helps but does not count any more. Remedy is to let another Siemens colleagues fully author PRs.*

Examples: CMP support for CRL fetching, TLS 1.3 integrity-only ciphers

SIEMENS

Handling of OpenSSL pull requests

Things to consider by contributors and reviewers

- relevance, better alternatives?
- functionally correct?
- crashes, memory leaks?
- coding style followed for readability and maintainability?
- documentation sufficient and consistent?
- sufficient tests?
- CI runs (various builds and tests) passing?

Reviewing is a social process

Delays in reviews and reactions on comments lead to

- concurrent changes requires fixing merge conflicts
- forgetting details in between

```
204 +     if (cipher == NULL) {
205 +         RLAYERfatal(rl, SSL_AD_INTERNAL_ERROR, ERR_R_INTERNAL_ERROR);
206 +         return 0;
```

DDvO on Mar 12 — Member

Since you potentially duped `mac_ctx`, this and all subsequent `return 0` need to be replaced by `goto err`

rajeev-0 5 days ago — Contributor, Author

refactored the code and introduced `end_mac` to always free the `mac_ctx` within the same code block.

👍 1

```
95        -     if (ctx == NULL || rec->type == SSL3_RT_ALERT) {
127       +     if (rec->type == SSL3_RT_ALERT) {
```

DDvO 2 days ago — Member

Since meanwhile `cipher = EVP_CIPHER_CTX_get0_cipher(ctx);` moved to further down, which is good,
the condition `ctx == NULL` is no more redundant (your argument of April 3 no more holds),
so now must revert to the original `ctx == NULL || rec->type == SSL3_RT_ALERT`
as otherwise `ivlen = EVP_CIPHER_CTX_get_iv_length(ctx);` will crash on `ctx == NULL`

rajeev-0 51 minutes ago • edited — Contributor, Author

As just discussed , I added a comment `/* enc_ctx is ignored when rl->mac_ctx != NULL */`.
Therefore no crash can occur in the below code if it is NULL.

👍 1

DDvO 2 minutes ago — Member

So I was wrong, thanks for explaining.
With the new comment on `ctx` (which is now called `enc_ctx`) it should be clear.

SIEMENS

My further OpenSSL Committer activity

I authored 500 closed PRs. Still ~30 open PRs on CMP fixes and more general improvements w.r.t. X.509 certificates, HTTP, CMS/PKCS#7, general crypto, tooling, error handling, and documentation.

I follow up these (except for CMP) in my spare time.

Most critical topic: X.509 certificate validation.

Further **spare time activity**:

- Occasionally review other folks' PRs, e.g., by former colleague on OCSP multi-stapling
- Occasional bug/issue reporting

Filters | is:open is:issue author:DDvO sort:updated-desc | Labe

Clear current search query, filters, and sorts

67 Open | 53 Closed | Author | Label | Project

- Improve dignostics on provider loading — issue: bug report, triaged: feature
  #21992 opened on Sep 7, 2023 by DDvO · updated on Feb 12
- Severe efficiency degradation of credential loading in comparison to 1.1.1 — triaged: feature, triaged: performance, triaged: question
  #18814 opened on Jul 17, 2022 by DDvO · updated on Feb 12
- X509: wrong? rejection of cert without CDP if CRL contains IDP — triaged: feature, triaged: question
  #23325 opened on Jan 17 by DDvO · updated on Feb 2

Filters | is:open is:pr author:DDvO | Labels 62 | Milestones 11 | New pull request

Clear current search query, filters, and sorts

29 Open | 500 Closed | Author | Label | Projects | Milestones | Reviews | Assignee | Sort

- CMP app: fix combination of -certout and -chainout with equal filename arg — approval: review pending; branch: master; branch: 3.0; branch: 3.2; branch: 3.3; tests: present; triaged: bug; triaged: documentation; 2
  #24267 opened last week by DDvO · Approved · 2 tasks done
- CMP: fix #23706 and add warning notes on `OSSL_CMP_OPT_PERMIT_TA_IN_EXTRACERTS_FOR_IR` — approval: review pending; branch: master; branch: 3.0; branch: 3.1; branch: 3.2; tests: exempted; triaged: bug; triaged: documentation; 1; 4
  #23814 opened on Mar 11 by DDvO · Approved
- CMP `test_connection.csv` : disable `localhost` test as not supported on some hosts — approval: review pending; branch: master; branch: 3.2; tests: present; triaged: bug; 1; 4
  #23756 opened on Mar 6 by DDvO · Approved
- EVP: add missing `*_get/settable*_params()` error queue entries and error result doc — approval: otc review pending; approval: review pending; branch: master; severity: fips change; tests: exempted; triaged: bug; triaged: documentation; 16
  #22914 opened on Dec 2, 2023 by DDvO · 1 of 2 tasks
- APPS/ `pkeyutl` : improve `-rawin` usability and doc — approval: review pending; branch: master; severity: ABI change; tests: present; triaged: documentation; triaged: feature; triaged: refactor; 24
  #22910 opened on Dec 2, 2023 by DDvO · Approved · 2 tasks done
- APPS: refactor `load_key_certs_crls()` — approval: review pending; branch: master; tests: exempted; triaged: refactor; 9
  #22528 opened on Oct 27, 2023 by DDvO · Approved
- CMS and PKCS7: add support for EdDSA with curves 25519 and 448 — branch: master; help wanted; severity: fips change; tests: present; triaged: cleanup; triaged: feature; 2; 12
  #22391 opened on Oct 15, 2023 by DDvO · Draft · 2 tasks done
- crypto/{CMS,PKCS7,OCSP,TS,X509}: constify cert list parameters; tidy up app code — branch: master; tests: exempted; triaged: cleanup; triaged: refactor; 13
  #22304 opened on Oct 7, 2023 by DDvO · Approved · 4.0.0

SIEMENS

Experience and insights of an OpenSSL Committer: Wrap-up

- **OpenSSL is most important security library** for Siemens
- Contributions to OpenSSL due to business demand
- **Being OpenSSL Committer is very helpful**
  - Learn a lot on coding, tools, and SW processes
  - Speed up contributions
  - More insight and leverage for fixing things
- Social aspects do play a role
- **Loads of work – not always paid**
- **Rewarding** to be part of an **important OSS project**

SIEMENS

OpenSSL
Cryptography and SSL/TLS Toolkit

SIEMENS

Contact

Dr. David von Oheimb

Security Architecture, PKI Team
Siemens T CST SEA-DE

Otto-Hahn-Ring 6
D-81739 Munich
Germany

David.von.Oheimb@siemens.com

Product PKI Wiki: https://wiki.siemens.com/display/ProductPKI

