Abridged Certs

draft-ietf-tls-cert-abridge

Dennis Jackson

Recap

- Adopted as a WG document by TLS WG @ IETF 117

- A new TLS certificate compression scheme which achieves much better compression ratios and suppresses intermediates certificates.

- Core idea:
  1. Compress the certificate chain down to just the end-entity certificate using a pre-shared list of WebPKI CA Certificates.
  2. Compress the end entity certificate using a pre-shared Zstd dictionary.

Issue: Complexity of Second Pass

Many folks gave feedback that pass 2 was more complex than it was worth:

- Defining the pre-shared dictionary relied on a messy algorithm for sampling from CT logs
- Zstd is not currently widely deployed for TLS Certificate Compression

Proposed Resolution: Switch to Brotli with no dictionary for Pass 2 (PR 26)

+ Much simpler
- A few % less compression

Issue: Specifying the Pass 1 Dictionary

- CCADB now offer a public export of the certificate lists
- Two options previously considered:
  - Baking it into the draft in an appendix
  - Migrating to a TLS extension which specifies (oldest, newest) entries known to client.
- RFC 7932 - Brotli, takes the first approach with a 244 KB appendix. Propose to keep in the appendix for now (PR 33)

```
The hexadecimal form of the DICT array is the following, where the
length is 122,784 bytes and the CRC-32 of the byte sequence is
0x5136cb04.

   74696d65646f776e6c6966656c6566746261636b636f6465646174617368 6f77
   6f6e6c797369746563697479 6f70656e6a7573746c696b6566726565776f726b
   7465787479656172 6f766572626f64796c6f7665666f726d626f6f6b706c6179
```

Issue: Codepoint Assignment

Happily, RFC 8879 allocates 0x4000 and up for experimental use, with no IANA approvals required.

Propose using **0xab00** for this experimental iteration (PR 30).

Effectiveness

Scheme	p5	p50	p95
Current Certificate Chains	2308	4032	5609
Existing TLS Certificate Compression	1619	3243	3821
This Draft	881	1256	1716

Byte sizes of certificate chains taken from sample of Tranco Top Sites

Estimates for PQ Certificate Chains

Working assumptions:

- ML-DSA 44 (Dilithium) for Handshake Signatures and Certificates
- FN-DSA (Falcon) for 2 SCTs
- Classical Cross Sign for Backwards Compatibility
- Fully FIPS & PQ Secure
- 0.2 to 1 KB of Leaf Certificate Metadata

Estimates for PQ Certificate Chains

Scheme	**Median**	**p95**
Estimated PQ Chains	12 KB	13 KB
Classical Chains Today	4.0 KB	6 KB
Estimated PQ with This Draft	7.5 KB	8.5 KB

Deployability

- Can be deployed by applications via existing TLS Cert Compression Library hooks even if the TLS library does not include Abridged itself.

- Can be deployed by TLS Libraries by default
  - Usage is unobservable to applications
  - Doesn’t impact trust decisions
  - No risk of connection breakage

- Doesn’t rely on DNS, OS integration or other external mechanisms

- Existing CA Practices make for easy versioning:
  - ISRG currently issues new intermediates at 2 year intervals and new roots at 5 year intervals.

Implementation Status

Firefox 132 (October 28th) shipped TLS Certificate Compression support

- Abridged Certificates support in final review, expected in next Nightly

abridged-certs-rs

- Standalone Rust Library implementing compression and decompression
- Integrates Rustls to build a TLS client and server with Abridged Certificates support
- 500 LoC including tests

- Aware of at least one other prototype being developed for evaluation

Next Step: Experimental Evaluation

What performance impact will there be for real world users if…

1. …we compress existing certificate chains from 4 KB to 1.2 KB?

2. …we inflate existing certificate chains from 4 KB to 7.5 KB (or more)?

Most interested in happy eyeball metrics like time to first paint, rather than less directly observable metrics like TLS handshake latency or TTFB / TTLB.

FIN

Issue: Complexity of Second Pass

Many folks gave feedback that pass 2 was more complex than it was worth:

- Defining the pre-shared dictionary relied on a messy algorithm for sampling from CT logs
- Zstd is not currently widely deployed for TLS Certificate Compression

Proposed Resolution: Switch to Brotli with no dictionary for Pass 2 (PR 26)

\+ Much simpler
\- A few % less compression

Scheme	p5	p50	p95
Draft 00	661	1060	1437
After PR	881 (+220)	1256 (+196)	1716 (+279)

Issue: Specifying the Pass 1 Dictionary

```
338  + ff0000:d7a7a0fb5d7e2731d771e9484ebcdef71d5f0c3e0a2948782bc83ee0ea699ef4
339  + ff0001:9a6ec012e1a7da9dbe34194d478ad7c0db1822fb071df12981496ed104384113
340  + ff0002:55926084ec963a64b96e2abe01ce0ba86a64fbfebcc7aab5afc155b37fd76066
341  + ff0003:0376ab1d54c5f9803ce4b2e201a0ee7eef7b57b636e8a93c9b8d4860c96f5fa7
342  + ff0004:0a81ec5a929777f145904af38d5d509f66b5e2c58fcdb531058b0e17f3f0b41b
343  + ff0005:70a73f7f376b60074248904534b11482d5bf0e698ecc498df52577ebf2e93b9a
344  + ff0006:bd71fdf6da97e4cf62d1647add2581b07d79adf8397eb4ecba9c5e8488821423
345  + ff0007:f356bea244b7a91eb35d53ca9ad7864ace018e2d35d5f8f96ddf68a6f41aa474
346  + ff0008:04048028bf1f2864d48f9ad4d83294366a828856553f3b14303f90147f5d40ef
                                    ...
2267  + ff0789:d0c97e56c7b0ba812d944ad771f7799b5d4144a2327a4e416554f7ee2aa0aeae
2268  + ff078a:812c212e9e45dc5005c7f47411183f5fb2ff1baee184d3354b2e93d78c280164
2269  + ff078b:b10b6f00e609509e8700f6d34687a2bfce38ea05a8fdf1cdc40c3a2a0d0d0e45
2270  + ff078c:e6fe22bf45e4f0d3b85c59e02c0f495418e1eb8d3210f788d48cd5e1cb547cd4
2271  + ff078d:2fe357db13751ff9160e87354975b3407498f41c9bd16a48657866e6e5a9b4c7
2272  + ff078e:dc9416c2f855126d6de977677538f2f967ff4998e90dfa435a17219be077fc06
2273  + ff078f:ae0fc852280f1b87cedaf73cfb84cf106efec88e8294253af352ed4034460d7b
2274  + ff0790:847409e63526f162753ac49f75218efaafa7d5c94ade9095ce72e7f6b6e3ac99
2275  + ff0791:6807c97235c5ec6090269a4b5fedfab46986e42f4d67d2edddcf6e45cf0dfa80
2276  + ff0792:72d716f7bb6bd105704f42b9524923510dcb85b2d870c0e9ada5aeb9c969051a
2277  + ```
```

Minor Changes

- Require unrecognised identifiers to result in decompression failures (PR 27)

- Editorial Cleanups
  - Remove old DISCUSS tags (PR 28)
  - Update Benchmarks (PR 29)
  - Optimize server-side footprint (PR 31)
  - Update Acknowledgements (PR 32)