Testssl.sh: what’s new?

Good things take a while .
And: what the heck is a KEM and do I need it?

Dirk Wetter, 24.4.2025

Agenda

- What the heck is he talking about??? ;-)
- New stuff
- Cool new stuff
- Short Demo
- Future

One thing upfront

- testssl.sh is not an OWASP project

Whatis . . .

- Testssl.sh is (just) a shell script (25k lines)
- Checks server side TLS encryption of any service
- Runs oob everywhere: Linux, macOS/BSDs, WSL
  - Almost zero dependencies
  - Except regular Linux/BSDish system tools
  - Uses bash sockets + OpenSSL/LibreSSL
- Contributors 123 , main: me, David @ NIST

News / Changes

- This Tuesday new release (3.2.0), finally! 🎉
- Old version still used in distros
- Changelog's major features: 😈 66
- ~1628 commits since 3.0

New stuff

- Rating (SSLlabs)

```
Rating (experimental)

Rating specs (not complete)  SSL Labs's 'SSL Server Rating Guide' (version 2009q from 2020-01-30)
Specification documentation  https://github.com/ssllabs/research/wiki/SSL-Server-Rating-Guide
Protocol Support (weighted)  100 (30)
Key Exchange     (weighted)  90 (27)
Cipher Strength  (weighted)  90 (36)
Final Score                  93
Overall Grade                A+

Done 2025-04-23 16:40:51 [0146s] -->> 172.67.10.39:443 (owasp.org) <<--
```

New stuff

- Order ciphers + prefs by protocol

```
 Testing server's cipher preferences

Hexcode  Cipher Suite Name (OpenSSL)       KeyExch.   Encryption  Bits     Cipher Suite Name (IANA/RFC)
-----------------------------------------------------------------------------------------------------------------
SSLv2
 -
SSLv3
 -
TLSv1
 -
TLSv1.1
 -
TLSv1.2 (server order -- server prioritizes ChaCha ciphers when preferred by clients)
 xc02f   ECDHE-RSA-AES128-GCM-SHA256       ECDH 253   AESGCM      128      TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
 xcca8   ECDHE-RSA-CHACHA20-POLY1305       ECDH 253   ChaCha20    256      TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256
 xc030   ECDHE-RSA-AES256-GCM-SHA384       ECDH 253   AESGCM      256      TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
 xc027   ECDHE-RSA-AES128-SHA256           ECDH 253   AES         128      TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
 xc013   ECDHE-RSA-AES128-SHA              ECDH 253   AES         128      TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
 xc014   ECDHE-RSA-AES256-SHA              ECDH 253   AES         256      TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA
 x9c     AES128-GCM-SHA256                 RSA        AESGCM      128      TLS_RSA_WITH_AES_128_GCM_SHA256
 x2f     AES128-SHA                        RSA        AES         128      TLS_RSA_WITH_AES_128_CBC_SHA
 x35     AES256-SHA                        RSA        AES         256      TLS_RSA_WITH_AES_256_CBC_SHA
TLSv1.3 (server order -- server prioritizes ChaCha ciphers when preferred by clients)
 x1301   TLS_AES_128_GCM_SHA256            ECDH 253   AESGCM      128      TLS_AES_128_GCM_SHA256
 x1302   TLS_AES_256_GCM_SHA384            ECDH 253   AESGCM      256      TLS_AES_256_GCM_SHA384
 x1303   TLS_CHACHA20_POLY1305_SHA256       ECDH 253   ChaCha20    256      TLS_CHACHA20_POLY1305_SHA256

 Has server cipher order?     yes (OK) -- TLS 1.3 and below

Done 2025-04-23 18:42:41 [0012s] -->> 185.199.111.153:443 (owasp.de) <<--
```

Cool new stuff

- Checks for PQ KEMs (Post-Quantum Key Encapsulation Mechanisms)
- Now: Mostly DH(E), ECDH(E) (formerly used also RSA)
- "Forward secrecy": save now, decrypt later
- Shor algorithm
- KEM = Public/Private key encapsulated
  - ML KEMs (Kyber with different lengths): MLKEM512 ++
  - Hybrid Key Agreement KEMs: X25519MLKEM768, SecP256r1MLKEM768 etc

Cool new stuff

need to wake up again ;)

```
Testing robust forward secrecy (FS) -- omitting Null Authentication/Encryption, 3DES, RC4

FS is offered (OK)           TLS_AES_256_GCM_SHA384 TLS_CHACHA20_POLY1305_SHA256 ECDHE-ECDSA-AES256-GCM-SHA384 ECDHE-ECDSA-CHACHA20-POLY1305
                             TLS_AES_128_GCM_SHA256 ECDHE-ECDSA-AES128-GCM-SHA256
KEMs offered                 MLKEM512 MLKEM768 MLKEM1024 SecP256r1MLKEM768 X25519MLKEM768 SecP384r1MLKEM1024
Elliptic curves offered:     prime256v1 X25519
TLS 1.2 sig_algs offered:    ECDSA+SHA256 ECDSA+SHA384 ECDSA+SHA512
TLS 1.3 sig_algs offered:    ECDSA+SHA256
```

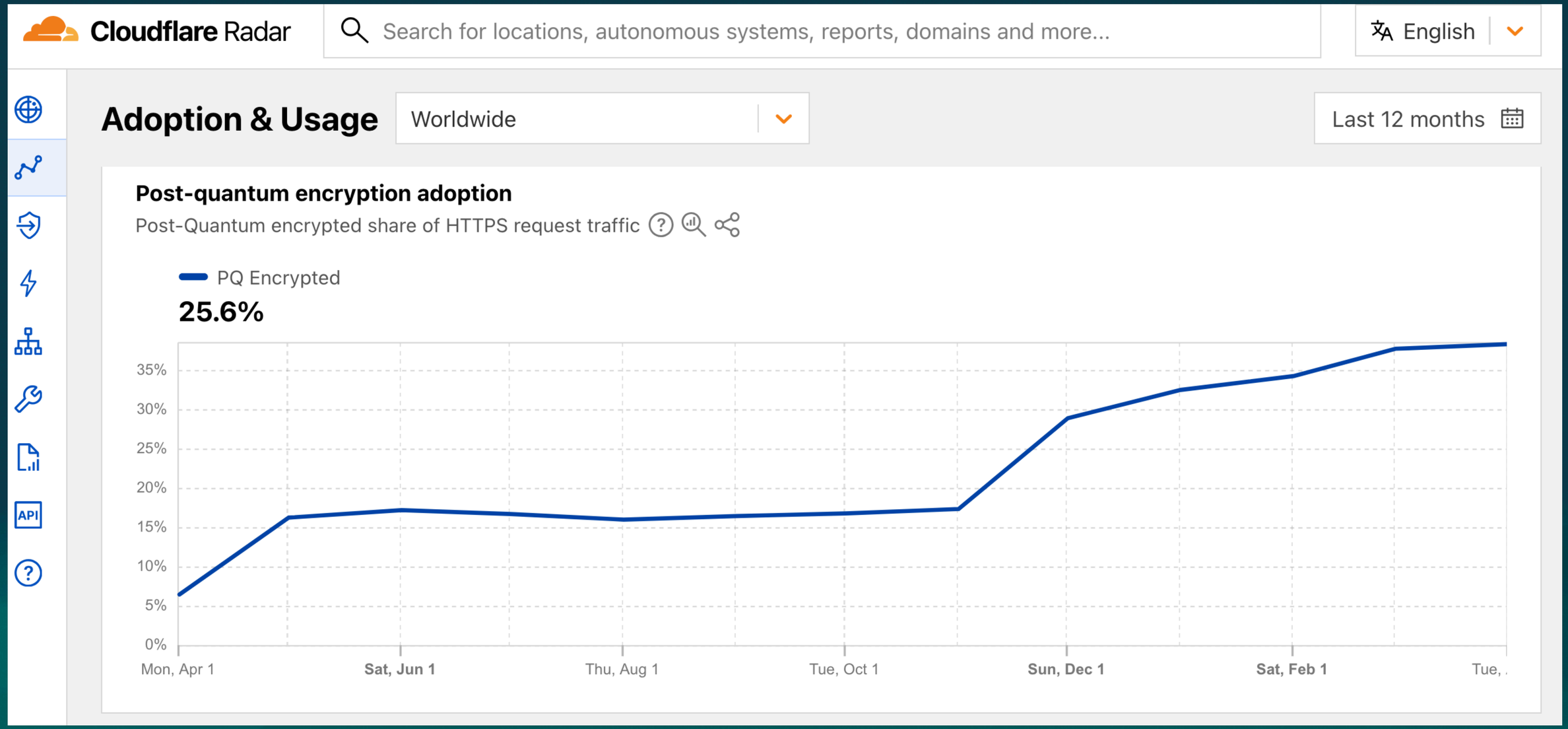
Running client simulations (HTTP) via sockets

Browser	Protocol	Cipher Suite Name (OpenSSL)	Forward Secrecy
Android 7.0 (native)	TLSv1.2	ECDHE-ECDSA-AES128-GCM-SHA256	256 bit ECDH (P-256)
Android 8.1 (native)	TLSv1.2	ECDHE-ECDSA-AES128-GCM-SHA256	253 bit ECDH (X25519)
Android 9.0 (native)	TLSv1.3	TLS_AES_128_GCM_SHA256	253 bit ECDH (X25519)
Android 10.0 (native)	TLSv1.3	TLS_AES_128_GCM_SHA256	253 bit ECDH (X25519)
Android 11/12 (native)	TLSv1.3	TLS_AES_128_GCM_SHA256	253 bit ECDH (X25519)
Android 13/14 (native)	TLSv1.3	TLS_AES_128_GCM_SHA256	253 bit ECDH (X25519)
Chrome 101 (Win 10)	TLSv1.3	TLS_AES_128_GCM_SHA256	253 bit ECDH (X25519)
Chromium 137 (Win 11)	TLSv1.3	TLS_AES_128_GCM_SHA256	X25519MLKEM768
Firefox 100 (Win 10)	TLSv1.3	TLS_AES_128_GCM_SHA256	253 bit ECDH (X25519)
Firefox 137 (Win 11)	TLSv1.3	TLS_AES_128_GCM_SHA256	X25519MLKEM768
IE 8 Win 7	No connection		
IE 11 Win 7	TLSv1.2	ECDHE-ECDSA-AES128-GCM-SHA256	256 bit ECDH (P-256)
IE 11 Win 8.1	TLSv1.2	ECDHE-ECDSA-AES128-GCM-SHA256	256 bit ECDH (P-256)
IE 11 Win Phone 8.1	TLSv1.2	ECDHE-ECDSA-AES128-GCM-SHA256	256 bit ECDH (P-256)
IE 11 Win 10	TLSv1.2	ECDHE-ECDSA-AES128-GCM-SHA256	256 bit ECDH (P-256)
Edge 15 Win 10	TLSv1.2	ECDHE-ECDSA-AES128-GCM-SHA256	253 bit ECDH (X25519)
Edge 101 Win 10 21H2	TLSv1.3	TLS_AES_128_GCM_SHA256	253 bit ECDH (X25519)
Edge 133 Win 11 23H2	TLSv1.3	TLS_AES_128_GCM_SHA256	X25519MLKEM768
Safari 18.4 (iOS 18.4)	TLSv1.3	TLS_AES_128_GCM_SHA256	253 bit ECDH (X25519)
Safari 15.4 (macOS 12.3.1)	TLSv1.3	TLS_AES_128_GCM_SHA256	253 bit ECDH (X25519)
Safari 18.4 (macOS 15.4)	TLSv1.3	TLS_AES_128_GCM_SHA256	253 bit ECDH (X25519)
Java 7u25	No connection		
Java 8u442 (OpenJDK)	TLSv1.3	TLS_AES_256_GCM_SHA384	253 bit ECDH (X25519)
Java 11.0.2 (OpenJDK)	TLSv1.3	TLS_AES_128_GCM_SHA256	256 bit ECDH (P-256)
Java 17.0.3 (OpenJDK)	TLSv1.3	TLS_AES_256_GCM_SHA384	253 bit ECDH (X25519)
Java 21.0.6 (OpenJDK)	TLSv1.3	TLS_AES_256_GCM_SHA384	253 bit ECDH (X25519)
go 1.17.8	TLSv1.3	TLS_AES_128_GCM_SHA256	253 bit ECDH (X25519)
LibreSSL 3.3.6 (macOS)	TLSv1.3	TLS_CHACHA20_POLY1305_SHA256	253 bit ECDH (X25519)
OpenSSL 1.0.2e	TLSv1.2	ECDHE-ECDSA-AES128-GCM-SHA256	256 bit ECDH (P-256)
OpenSSL 1.1.1d (Debian)	TLSv1.3	TLS_AES_256_GCM_SHA384	253 bit ECDH (X25519)
OpenSSL 3.0.15 (Debian)	TLSv1.3	TLS_AES_256_GCM_SHA384	253 bit ECDH (X25519)
OpenSSL 3.5.0 (git)	TLSv1.3	TLS_AES_256_GCM_SHA384	X25519MLKEM768
Apple Mail (16.0)	TLSv1.2	ECDHE-ECDSA-AES128-GCM-SHA256	256 bit ECDH (P-256)
Thunderbird (91.9)	TLSv1.3	TLS_AES_128_GCM_SHA256	253 bit ECDH (X25519)

Done 2025-04-23 17:10:20 [0047s] -->> 104.22.26.77:443 (owasp.org) <<--

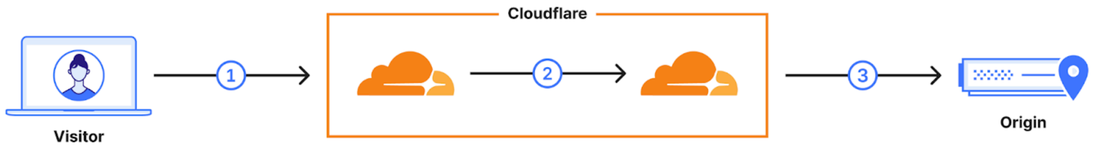
(copy

https://radar.cloudflare.com/adoption-and-usage?dateRange=52w :



Test your browser @ https://pq.cloudflareresearch.com/

Cloudflare Research: Post-Quantum Key Agreement



On essentially all domains served (1) through Cloudflare, including this one, we have enabled hybrid quantum key agreement. We are also rolling out support for post-quantum key agreement for connec Cloudflare to origins (3). Check out our blog post the state of the post-quantum Internet for more co

You are using *X25519MLKEM768* which is **post-quantum secure**.

You are using *X25519MLKEM768* which is **post-quantum secure**.

- Firefox
- Chrome-Derivate

You are using *X25519* which is **not post-quantum secure**.

- Safari

Further stuff

- Added some vulnerability checks
  - Winshock (very old one)
  - STARTTLS injection (the new one) *)
- STARTTLS
  - XMPP server
  - LDAP / AD support
  - Sieve
  - TN 3270/telnet (!)

*) Damian, Fabian, Hanno & Sebastian: https://www.usenix.org/conference/usenixsecurity21/presentation/poddebniak)

Count down to demo -1 ;-)

- Detection of wildcard certificates (+warning)
- Expiration + bad OCSP status for intermediate certificates
- TLS 1.3 decrypting server response
- More new ciphers to test
- Server signature algorithms list
- EdDSA (ed25519 + ed448)
- ML-DSA

Demo

Outlook (no, not that...)

- Important:
  - Faster
  - QUIC / HTTP/3 (possible?)
  - Pluggable compliance scans

Outlook (no, not that…)

- Features
  - DNS HTTP RR (RFC 9460)
    - PR pending
  - MTA STS (RFC 8461), PR pending but WIP
  - DANE / TLSA (DNS-based Authentication of Named Entities)
  - Query Hanno’s badkeys (https://badkeys.info/)
  - IPv6 handling
  - …

```
Start 2024-03-30 17:41:36                -->> 104.16.133.229:443 (cloudflare.com) <<

Further IP addresses:   104.16.132.229 2606:4700::6810:84e5 2606:4700::6810:85e5
rDNS (104.16.133.229):  --
Service detected:       HTTP
DNS HTTPS RR (experim.) yes 1 . alpn="h3,h2" ipv4hint=104.16.132.229,104.16.133.229
                                ipv6hint=2606:4700::6810:84e5,2606:4700::6810:85e5
```

Danke

- Contact: dirk at owasp dot org or at testssl dot org, linked.in
- https://github.com/testssl/testssl.sh/, mastodon / bluesky
- Slides are CC-BY-SA 4.0 will be uploaded later