TLS 1.3

Kyle Murray, Mateusz Piekut

What is it:

TLS stands for Transport Layer Security, and is based on an older version of SSL. TLS relies on symmetric cryptography that encrypts data throughout the conversation. Each key is unique to its conversation; they are based on a shared secret which is determined through the TLS handshake at the beginning of the session. The keys and algorithm used are part of the shared secret.

How is it different from previous standards:

TLS 1.3 is faster than the previous versions, as it only requires one round trip to complete the TLS handshake, compared to the two required for TLS 1.2. Application data can now be sent on the initial handshake; this reduces the latency clients feel when doing the initial connection, which will increase the initial server connection speed and results in less waiting for the browser to load. Caching is also handled a bit differently in TLS 1.3. In previous versions a session id is sent from the client which is compared server side to see if the client is cached. 1.3 implemented a new ticketing system which allows for a stateless server. The ticket contains information from older sessions.

An in depth look at the new handshake:

In TLS 1.2 the client sends a “client hello” to the server. In this client hello the client will say what version of TLS it has and give some cipher suites it can use. The server will then respond with a “server hello”, where the server will also pick one of the given cipher suites. If the client and server cannot agree on a cipher suite the conversation is terminated. Along with the selected cipher suite the server will also share a key with the client. The client will then share a key with the server. After all this is done the encrypter channel can then be used.

Now in TLS 1.3 the process is simplified. Added to the client hello is a key for an estimated algorithm the server probably supports and will use. If the server supports the algorithm it will select it and accept the client sent key. Then the server will respond with a server hello that just sends over its key and a confirmation, saving a complete round trip.

An in depth look at the new caching system:

In TLS 1.2 a client and server can speed up the handshake session if the client has visited the site recently. The client would have some sort of ticket or identifier. The client would send this information in the client hello. Most servers will handle this by storing these tickets in a database and checking it against the one received in the client hello. If the server remembers the client they can skip the rest of the handshake process, making the initial connection last only one round trip.

TLS 1.3 cuts out a round trip in regards to caching, meaning a client who is remembered on a server can connect in zero round trips with almost no overhead. If a client has recently connected to a server the server and client choose a pre shared key. The server also will give the client a ticket. When the client wants to reconnect they send the ticket as well as their http request encrypted with the agreed upon pre-shared key in their hello. The server can then decrypt the request using the ticket send by the client.

Removed Features:

TLS 1.3 has trimmed the fat, removing many outdated, unneeded, or vulnerable features. The static RSA handshake has been removed, replaced with the new handshake which removes one round trip. Cipher block chaining has been removed to mitigate some vulnerabilities. RC4 has been removed because it has known vulnerabilities. SHA1 and MD5 are some outdated hashing algorithms that have been removed. Removing SHA1 and MD5 defends against the SLOTH attack. Compression has been removed, fixing the vulnerability exploited by the CRIME attack.

Added Features:

Feature have been added to TLS 1.3 to increase the security and speed of your internet connection. Elliptic curve algorithms Curver448 and Curve25519 have been added. Full handshake signature has been included in the update, improving security against FREAK and logjam attacks. The improved resumption of sessions is a big feature added in 1.3. This feature is meant to speed up even further the caching system of users to servers.

When can you expect TLS 1.3:

TLS 1.3 has been fighting to get into production for some time now. In early 2017 Mozilla Firefox started using TLS 1.3, however it was soon disabled because it was incompatible to a lot of users. Google did a similar thing also in 2017 and also disabled it after a brief period because of middleboxes being incompatible with it.

Most recently TLS 1.3 has proposed an internet draft to the Internet Engineering Task Force (IETF) on March 21st 2018. An internet draft is basically a list of added specs, removed specs, technical documents, and other such things. The internet task force will examine the draft and determine if it is ready to start being implemented as the new standard. It is quite possible we will start seeing the switch to TLS 1.3 within this year.

The Drown attack:

The drown attack is a known vulnerability of TLS. Drown stands for decrypting tsa with obsolete and weakened encryption. Essentially how this attack works is an attacker would probe a server for a previous version of ssl, specifically SSLv2. This version of ssl would still be present, which allows the attacker to decrypt the conversation with that version. This vulnerability is removed in TLS version 1.3 by removing primitives. Sorry bad guys, maybe next time.

Lucky Thirteen:

The lucky thirteen attack has stalked TLS for years. This attack is based of the padding oracle attack founded by Serge Vaudenay. Lucky thirteen uses a timing side channel attack against the message authentication code. These MAC headers have 13 bytes, hence Lucky 13. Given the known 13 byte headers, as well as how long the encryption algorithm takes, an attacker can calculate the input. Worry not peers, TLS 1.3 has your back, as it removed CBC and with it the vulnerability Lucky Thirteen relied on.

Poodle:

Poodle stands for padding Oracle on downgraded legacy encryption. This attack preys on clients and servers who still have SSL 3.0. The attack is very slow, only revealing one byte of plaintext for every 256 requests. This attack is mitigated in TLS 1.3 by removing primitives.

SLOTH:

SLOTH, otherwise known as Security Losses from Obsolete and Truncated Transcript Hashes, is an attack against hashes. The attack forces TLS to choose an obsolete hashing algorithm such as MD5 and SHA224. Using obsolete hashing functions reduces security. The hashing function is selected in the signature. A man in the middle can select a different hashing function than what is up to the current standards. TLS 1.3 has fixed this error by removing outdated hashing functions.

CRIME:

Compression Ratio Info-Leak Made Easy is an attack which exploits compression algorithm done by the client. This attack can be used to take over a users session by accessing their cookies. TLS version 1.3 has removed compression compatibility, making this attack obsolete.

FREAK:

Factoring Attack on RSA-Export Keys is an attack which intercepts connections between client and server and forces them to use a weaker type of encryption. When the conversation is set to use this lower grade encryption standard the man in the middle attacker can then easily decrypt the encrypted dialogue. This attack has been mitigated in TLS 1.3 with the addition of the full handshake signature.

The Logjam Attack:

The logjam attack exploits vulnerabilities in the Diffie-Hellman key exchange. This vulnerability allows a man in the middle attack to reduce the key size of susceptible channels. Reducing the key size allows attackers to brute force through the encryption within a reasonable timeframe. Along with the FREAK attack this vulnerability has been mitigated with the addition of the full handshake signature introduced in TLS version 1.3.

Recap:

TLS 1.3 is slimmer, faster and more secure. Slimmer because it has dropped a lot of dead weight. Many outdated features were removed mitigating vulnerabilities as well as simply making it more compact compared to previous versions. The most important feature of TLS 1.3 is the speed. The handshake has been reduced by one round trip in a both cached users and new users when connecting to the server. It has become more secure because it has mitigated vulnerabilities; by either dropping components or adding new ones in, many known attacks have been made obsolete with the new version of TLS.

References:

Jackson, Brian. "An Overview of TLS 1.3 – Faster and More Secure." *Kinsta.* Kinsta, 25 March 2018. Web. 3 May 2018. https://kinsta.com/blog/tls-1-3/

"Differences between TLS 1.2 and TLS 1.3." *WolfSSL.* WolfSSL, 15 June 2017. Web. 3 May 2018. https://www.wolfss

Valsorda, Filippo. "An overview of TLS 1.3 and Q&A." *Cloudflare.* Cloudflare, 23 Sept. 2016. Web. 3 May 2018. https://blog.cloudflare.com/tls-1-3-overview-and-q-and-a/

"The Drown Attack." *Drownattack.* DROWN, 1 July 2016. Web. 3 May 2018. https://drownattack.com/

"Lucky 13 – a new attack against SSL/TLS." *InfoSecurity.* InfoSecurity Group, 7 Feb. 2013. Web. 3 May 2018. https://www.infosecurity-magazine.com/news/lucky-13-a-new-attack-against-ssltls/

"SSL 3.0 Protocol Vulnerability and POODLE Attack." *US-CERT.* US-CERT, 30 Sept. 2016. Web. 3 May 2018. https://www.us-cert.gov/ncas/alerts/TA14-290A

"SLOTH: TLS 1.2 vulnerability (CVE-2015-7575)." *Redhat.* Redhat, 18 Jan. 2018. Web. 3 May 2018. https://access.redhat.com/articles/2112261

"CRIME SSL/TLS attack." *Acunetix.* Acunetix, n/a. Web. 3 May 2018. https://www.acunetix.com/vulnerabilities/web/crime-ssl-tls-attack

"The FREAK Attack." *CensysBlog.* Censys, 3 March 2015. Web. 3 May 2018. https://censys.io/blog/freak

"Weak Diffie-Hellman and the LogjamAttack." *Weakdh.* Weakdh, 13 Oct. 2015. Web. 3 May 2018. https://weakdh.org/

Presentation images:

Valsorda, Filippo. "TLS-1.3.004.png." *Cloudflare,* Cloudflare, 23 Sept. 2016. https://blog.cloudflare.com/tls-1-3-overview-and-q-and-a/

Valsorda, Filippo. "TLS-1.3.012.png." *Cloudflare,* Cloudflare, 23 Sept. 2016. https://blog.cloudflare.com/tls-1-3-overview-and-q-and-a/

Valsorda, Filippo. "tls-1.2-vs-1.3-a.png." *Cloudflare,* Cloudflare, 23 Sept. 2016. https://www.cloudflare.com/learning-resources/tls-1-3/

Valsorda, Filippo. "tls-1.2-vs-1.3-b.png." *Cloudflare,* Cloudflare, 23 Sept. 2016. https://www.cloudflare.com/learning-resources/tls-1-3/