The Logjam Attack

Warning! Your web browser is vulnerable to Logjam and can be tricked into using weak encryption. You should update your browser.

Diffie-Hellman key exchange (https://en.wikipedia.org/wiki/Diffie–Hellman_key_exchange) is a popular cryptographic algorithm that allows Internet protocols to agree on a shared key and negotiate a secure connection. It is fundamental to many protocols including HTTPS, SSH, IPsec, SMTPS, and protocols that rely on TLS.

We have uncovered several weaknesses in how Diffie-Hellman key exchange has been deployed:

1. **Logjam attack against the TLS protocol.** The Logjam attack allows a man-in-the-middle attacker to downgrade vulnerable TLS connections to 512-bit export-grade cryptography. This allows the attacker to read and modify any data passed over the connection. The attack is reminiscent of the FREAK attack (http://freakattack.com), but is due to a flaw in the TLS protocol rather than an implementation vulnerability, and attacks a Diffie-Hellman key exchange rather than an RSA key exchange. The attack affects any server that supports `DHE_EXPORT` ciphers, and affects all modern web browsers. 8.4% of the Top 1 Million domains were initially vulnerable.

2. **Threats from state-level adversaries.** Millions of HTTPS, SSH, and VPN servers all use the same prime numbers for Diffie-Hellman key exchange. Practitioners believed this was safe as long as new key exchange messages were generated for every connection. However, the first step in the number field sieve—the most efficient algorithm for breaking a Diffie-Hellman connection—is dependent only on this prime. After this first step, an attacker can quickly break individual connections.

   We carried out this computation against the most common 512-bit prime used for TLS and demonstrate that the Logjam attack can be used to downgrade connections to 80% of TLS servers supporting `DHE_EXPORT`. We further estimate that an academic team can break a 768-bit prime and that a nation-state can break a 1024-bit prime. Breaking the single, most common 1024-bit prime used by web servers would allow passive eavesdropping on connections to 18% of the Top 1 Million HTTPS domains. A second prime would allow passive decryption of connections to 66% of VPN servers and 26% of SSH servers. A close reading of published NSA leaks shows that the agency's attacks on VPNs are consistent with having achieved such a break.

More Information

We have published a technical report, **Imperfect Forward Secrecy: How Diffie-Hellman Fails in Practice** (imperfect-forward-secrecy.pdf), which has specifics on these attacks, details on how we broke the most common 512-bit Diffie-Hellman group, and measurements of who is affected.

We have also published a **Logjam Server Test** (sysadmin.html), **Guide to Deploying Diffie-Hellman for TLS (sysadmin.html)**, and several **proof of concept demos (logjam.html)**.

This study was performed by computer scientists at CNRS, Inria Nancy-Grand Est, Inria Paris-Rocquencourt, Microsoft Research, Johns Hopkins University, University of Michigan, and the University of Pennsylvania: David Adrian (https://davidadrian.org/), Karthikeyan Bhargavan (https://prosecco.gforge.inria.fr/personal/karthik/), Zakir Durumeric (https://zakird.com), Pierrick Gaudry (http://www.loria.fr/~gaudry/index.en.html), Matthew Green (https://isi.jhu.edu/~mgreen/), J. Alex Halderman (https://jhalderm.com), Nadia Heninger (http://www.cis.upenn.edu/~nadiah/), Drew Springall (https://aaspring.com), Emmanuel Thomé (http://www.loria.fr/~thome/), Luke Valenta (https://www.seas.upenn.edu/~lukev/), Benjamin VanderSloot (https://benjaminvandersloot.com), Eric Wustrow (https://ericw.us/trow/), Santiago Zanella-Beguelin (http://software.imdea.org/~szanella/), and Paul Zimmermann (http://www.loria.fr/~zimmerma/). The team can be contacted at **weakdh-team@umich.edu (mailto:weakdh-team@umich.edu)**.

Who is affected?

Websites, mail servers, and other TLS-dependent services that support `DHE_EXPORT` ciphers are at risk for the Logjam attack. We use Internet-wide scanning (https://zmap.io) to measure who is vulnerable.

Protocol	Vulnerable to Logjam
HTTPS — Top 1 Million Domains	8.4%
HTTPS — Browser Trusted Sites	3.4%
SMTP+StartTLS — IPv4 Address Space	14.8%
POP3S — IPv4 Address Space	8.9%
IMAPS — IPv4 Address Space	8.4%

Websites that use one of a few commonly shared 1024-bit Diffie-Hellman groups may be susceptible to passive eavesdropping from an attacker with nation-state resources. Here, we show how various protocols would be affected if a single 1024-bit group were broken in each protocol, assuming a typical up-to-date client (e.g., most recent version of OpenSSH or up-to-date installation of Chrome).

Vulnerable if most common 1024-bit group is broken

	Vulnerable if most common 1024-bit group is broken
HTTPS — Top 1 Million Domains	17.9%
HTTPS — Browser Trusted Sites	6.6%
SSH — IPv4 Address Space	25.7%
IKEv1 (IPsec VPNs) — IPv4 Address Space	66.1%

What should I do?

If you run a server…

If you have a web or mail server, you should disable support for export cipher suites and generate a unique 2048-bit Diffie-Hellman group. We have published a **Guide to Deploying Diffie-Hellman for TLS (sysadmin.html)** with step-by-step instructions. If you use SSH, you should upgrade both your server and client installations to the most recent version of OpenSSH, which prefers Elliptic-Curve Diffie-Hellman Key Exchange.

If you use a browser…

Make sure you have the most recent version of your browser installed, and check for updates frequently. Google Chrome (including Android Browser), Mozilla Firefox, Microsoft Internet Explorer, and Apple Safari are all deploying fixes for the Logjam attack.

If you're a sysadmin or developer …

Make sure any TLS libraries you use are up-to-date, that servers you maintain use 2048-bit or larger primes, and that clients you maintain reject Diffie-Hellman primes smaller than 1024-bit.

These results were published on May 20, 2015.