The Foundation for PayPal's June 2017 TLS 1.2 Upgrade

Technical White Paper

Contents

Summary

As a leader in Financial Technology, PayPal's mission is to provide simple, affordable, secure and reliable financial services and digital payments. To accomplish this mission, we constantly evaluate our security posture by weighing input from various organizations, councils, and standards to create an environment that merchants and customers can trust.

Despite recent PCI Council recommendations to delay the mandate to upgrade to at least TLS 1.1 and preferably TLS 1.2 until 2018, PayPal has elected to move forward with upgrading our systems prior to the PCI Council recommended date. On **June 30, 2017**, PayPal will begin the process of discontinuing support for TLS 1.0 and 1.1. This means all merchant API communications with PayPal will need to use TLS 1.2.

We have authored this technical white paper to explain the rationale behind this decision and direct merchants to the PayPal resources available for upgrade support.

Rationale for 2017 Upgrade

The 2016 date for SHA-256 certificate upgrade remains unchanged.

In 2014, the major browser and Certificate Authority communities announced that starting January 1, 2016, they would no longer issue SHA-1 certificates, and all newly issued certificates would be SHA-256. Starting January 1, 2017, all existing SHA-1 certificates would be considered 'non-trusted', and associated alert messaging would be displayed.

Merchants who integrated prior to 2009 will need to upgrade their operating system and/or SSL stack and key store to support these new SHA-256 certificates. By upgrading to TLS 1.2 and the G5 Root certificate, merchants are guaranteed to support SHA-256. The same cannot be said for TLS 1.0 and 1.1 deployments; additional upgrades and costs may be required.

When merchants upgrade to SHA-256 certifications, the vast majority will already be TLS 1.2-capable, so the work to enable TLS 1.2 only will be minimal. Upgrading to support SHA-256 certificates and TLS 1.2 at the same time is the most efficient and secure way to perform both of these actions.

SSL and TLS threats and mitigations are constantly evolving.

The SSL[i] and TLS[ii] protocols have evolved as new threats emerge. Since 2011, SSL 3.0 and TLS 1.0 and 1.1 have suffered three major attacks[iii] and several deprecations that have cost companies millions of dollars. The risk of breach with older versions of TLS is significant, and PayPal intends to pursue the most secure options available in an effort to eliminate their impact. Enabling only TLS 1.2 eliminates the risk that vulnerabilities in earlier versions can affect PayPal or our merchants. Today, TLS 1.2 offers the best security features to protect our communication channels.

Avoiding unplanned emergency upgrades benefits merchants, consumers, and PayPal.

The October 2014 POODLE attack resulted in an unprecedented action by the PCI Council to elevate SSL 3.0 usage to an *extreme risk* and required upgrades to TLS 1.0 in a highly aggressive timeframe of two months. The industry's assessment of many aspects of TLS 1.0 and 1.1 (e.g. the rapidly increasing vulnerability of SHA-1) has resulted in concerns that a future attack on one of these versions may spark another truncated upgrade timeframe. By adhering to PayPal's June deadline, merchants protect themselves from needing to react to such a sudden decision.

TLS 1.2 upgrades were likely already on merchant roadmaps for 2016/2017.

The PCI Council originally announced the 2016 TLS 1.2 upgrade mandate in April 2015 and the extension in December 2015. By the time that the extension was

announced, most merchants likely had 2016/2017 roadmaps that accounted for the original date. Setting a June 2017 date allows PayPal and merchants to continue with their original roadmaps and better secure their businesses. This is the most modern configuration available and will have the longest lifecycle of all currently existing options.

Many companies are taking the same stance on enabling TLS 1.2 only.

Several companies in the tech industry have also made the determination that only allowing TLS 1.2 is the most secure decision for the internet community. Apple's Application Transport Security (ATS) mandates TLS 1.2 connections for both desktop and mobile platforms[iv]. Any new applications developed for iOS 9 must only use TLS 1.2 or file an exception. Given that Apple holds nearly 60% of mobile traffic, this decision affects the majority of internet users.

In the same spirit as PayPal and Apple, CloudFlare has also decided to mandate TLS 1.2 connections only. CloudFlare accelerates and secures website traffic by acting as a proxy between site visitors and web servers. At 35% of managed DNS traffic, this decision drives a significant portion of site traffic for small and medium-sized businesses.

These company decisions combined with significant commentary from security researchers, bloggers, and companies such as SSL Labs, reinforces our confidence that only allowing TLS 1.2 is the right decision for PayPal and its merchants.

PCI and NIST recommendations state that TLS 1.2 is superior to earlier versions.

While PCI does not explicitly prohibit TLS 1.1, the vast majority of PCI documentation and NIST recommendations strongly advocate moving to TLS 1.2:

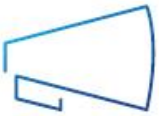
- In [Council][v], page 2, SSL and early TLS no longer meet the security needs of entities implementing strong cryptography to protect payment data over public or untrusted communications channels.
- In [Council], page 2, the PCI Security Standards Council advises to move to a more modern encryption protocol which "*/.../ at the time of publication is a minimum of TLS v1.1, although entities are strongly encouraged to consider TLS v1.2*". On the same page, they also say: *"Note that not all implementations of TLS v1.1 are considered secure – refer to NIST SP 800-52 rev 1 for guidance on secure TLS configurations.*"
- In [Thomas][vi], the (unofficial) PCI Compliance Guide summarizes PCI DSS 3.1 by saying: "*These evolving requirements seek to eliminate SSL and early TLS (versions 1.0 and 1.1)*".
- NIST recommends that both server and client be moved to TLS 1.2. For instance, in [NIST], although TLS 1.1 with "*cipher suites using Approved schemes and algorithms*" is considered the minimum required, the recommendation is "*/.../ that agencies develop migration plans to TLS 1.2, configured using Approved schemes and algorithms, by January 1, 2015*".

Merchant Resources

In order to keep the lines of communication open between PayPal and our merchants, we have created the TLS 1.2 Upgrade Microsite where we will post up to the minute information about this process.

In addition to the microsite, merchants can contact PayPal via their account representative or normal Help Desk contact methods.

In Conclusion



We want to reemphasize that PayPal believes the decision to move forward with a date in advance of the PCI Council 2018 upgrade timeline will provide greater protection to PayPal merchants, their data and customers. While we're grateful to the PCI Council for steering the industry in the right direction, PayPal's decision to upgrade earlier is rooted in doing what is best for the security of our merchants, customers, and the company, as well as meeting the mandate by the Certificate Authority and Browser Industries to upgrade to SHA-256 certificates by December 31, 2016. Minimizing the threat of fraud and stolen financial data is a key tenet of our business and one of the reasons that we are one of the most trusted payment platforms in the world.

---

[i] **SSL 3.0:** Secure Sockets Layer 3.0, the predecessor to TLS, was designed in the 1990s for secure transmission of data across the web. In recent years, most browsers/users were no longer using SSL, but if a browser was outdated, it might default back to using SSL 3.0. Most companies (including PayPal) continued to support SSL 3.0 until the 2014 POODLE attack. SSL 3.0 support was disabled shortly after.

[ii] **TLS:** Transport Layer Security is the current standard in securing communications over the web. TLS was designed in 1999 with updates in 2006, 2008, and 2011, which resulted in TLS 1.0, 1.1, 1.2, and 1.3. For the purpose of this commentary, we are talking about ending support for TLS 1.0 and 1.1. This is partially due to the 2015 changes in PCI DSS 3.1 (Payment Card Industry Data Security Standards). For further information, see Transport Layer Security.

[iii] **SSL/TSL attacks:** The following attacks and deprecations have occurred since 2011:

2011: Beast. Beast was an attack on SSL/TLS encryption that revealed man-in-the-middle (MITM) decryption of HTTPS protected session cookies and vulnerabilities in the CBC block cipher and TLS 1.0. This attack resulted in promoting the use of TLS 1.1 and 1.2 plus deprecating CBC and using the RC4 stream cipher for improved security.

2013: Crime. Compression Ratio Info-leak Made Easy (CRIME) was a client-side attack that exploited secret web cookies over connections using the HTTPS and SPDY protocols that also use data compression. It can be mitigated by implementing the no-compression feature in the TLS 1.2 protocol.

2014: POODLE. Padding Oracle On Downgraded Legacy Encryption (POODLE) is a MITM attack that exploits older clients' and outdated browsers' tendencies to fall back from TLS 1.0 and 1.1 to SSL 3.0. This vulnerability resulted in most large companies, including PayPal, deciding to stop supporting SSL 3.0. TLS 1.0 and higher become the supported standard.

2013-2015: RC4. In 2011, RC4 was the stream cipher that mitigated Beast because browsers were not able to prevent the attack alone. By 2013, weaknesses in RC4 were being discovered and browsers were then capable of protecting against Beast-like

attacks. By 2015, the IETF declared the deprecation of RC4, stating significant security flaws. This case is significant because it illustrates the constantly evolving security landscape and the need to continually evaluate the state of what is considered secure.

[iv] **From Apple's App Transport Security:** "App Transport Security (ATS) enforces best practices in the secure connections between an app and its back end. ATS prevents accidental disclosure, provides secure default behavior, and is easy to adopt; it is also on by default in iOS 9 and OS X v10.11. You should adopt ATS as soon as possible, regardless of whether you're creating a new app or updating an existing one. If you're developing a new app, you should use HTTPS exclusively. If you have an existing app, you should use HTTPS as much as you can right now, and create a plan for migrating the rest of your app as soon as possible. In addition, your communication through higher-level APIs needs to be encrypted using TLS version 1.2 with forward secrecy. If you try to make a connection that doesn't follow this requirement, an error is thrown. If your app needs to make a request to an insecure domain, you have to specify this domain in your app's Info.plist file."

[v] **[Council] PCI Security Standards.** "Migrating from SSL and Early TLS." April 2015. PCI Security Standards.

[vi] **[Thomas] Tim.** "A First Look at PCI DSS 3.1." 22 April 2015. PCI Compliance Guide.