KEEPING THE INTERNET
OPEN • INNOVATIVE • FREE
www.cdt.org
CENTER FOR DEMOCRACY
& TECHNOLOGY

No Half Measures:
Digital Marketing Properties Must Adopt Encryption Best Practices

By Greg Norcie and Joseph Lorenzo Hall,
Center for Democracy & Technology
5 May 2015 (v1.0)

I. Introduction: Data is Increasingly Encrypted

In the wake of Edward Snowden's revelations, data on the web is increasingly encrypted in transit. This is typically accomplished via the usual Hypertext Transfer Protocol[1] (HTTP) but over Transport Layer Security (TLS)[2] for what is called "HTTP over TLS," commonly known as HTTPS. Sites that utilize TLS begin with "https://" (with the S standing for secure), and "HTTPS" is often used as a synonym for TLS and/or its predecessor, the Secure Socket Layer (SSL) protocol. However, as we note later in this paper, all versions of the SSL protocol are flawed, and should not be enabled on any modern web server. For the purposes of this paper, any mention of HTTPS should be assumed to be referring to an HTTP transaction over TLS.

There is increasing consensus that HTTPS is critical for a trusted web. The Internet Engineering Task Force's Internet Architecture Board released a statement on Internet Confidentiality[3]. This statement acknowledged that in the wake of Edward Snowden's revelations about pervasive passive monitoring of the web, encryption should be deployed throughout the protocol stack. Similarly the W3C Technical Architecture Group has thrown its support behind HTTPS being a "baseline requirement"[4] for web interactions. And the United States CIO has proposed that in the future, all government websites will be accessible only via HTTPS[5] ("HTTPS-Only"). While this is not yet a binding US government standard, several government sites such as Whitehouse.gov[6] and FTC.gov[7] have already enabled HTTPS-Only and when this standard is approved, all US government web sites will transition to HTTPS-Only.

[1] RFC 216, "Hypertext Transfer Protocol HTTP/1.1," *available at:* https://tools.ietf.org/html/rfc2616
[2] RFC 5246, "The Transport Layer Security (TLS) Protocol Version 1.2," *available at:* https://tools.ietf.org/html/rfc5246
[3] "Statement on Internet Confidentiality," Internet Architecture Board (November 14, 2014), *available at:* https://www.iab.org/2014/11/14/iab-statement-on-internet-confidentiality/
[4] W3C Technical Architecture Group, "Securing the Web," W3C TAG Finding (January 22, 2015), *available at:* http://www.w3.org/2001/tag/doc/web-https
[5] "The HTTPS-Only Standard Proposal," US Office of the Chief Information Officer, *available at:* https://https.cio.gov/
[6] Whitehouse Official Twitter, *available at:* https://twitter.com/WHWeb/status/575509388133335041
[7] Ashkan Soltani, "FTC.gov is now HTTPS by default," Tech@FTC Blog (March 6, 2015), *available at:* https://www.ftc.gov/news-events/blogs/techftc/2015/03/ftcgov-now-https-default

II. Mixed Content Hurts Security

Mixed content is harmful. A website's security can only be as strong as its weakest link. Just as one "weak link" in a chain will cause the entire chain to snap in two, a single flaw in a computer system can grant an attacker access to said system. Thus, the attacker has the advantage: while they can attack over and over, the defender may only fail once. Thus, it is very important that systems be designed to be as strong as possible. This means, for example, that if an otherwise secure HTTPS enabled site loads external Javascript over HTTP, said Javascript can be hijacked and used to inject malicious software.

Unfortunately, as Michael Kranch and Joseph Bonneau point out in a recent research paper[8], one major source of insecurity – mixed content – generally arises from web analytics and advertising resources.

At the high level, "mixed content" refers to the mixing of secure and insecure content on a single web page.

There are two types of mixed content: passive and active. Passive content generally refers to non-executable content with minimal risk, such as digital images, served over plain HTTP. While passive mixed content also presents issues, active mixed content such as JavaScript, CSS, fonts, and iframes presents a graver security risk. For example, votes cast in the recent state elections of New South Wales, Australia, were vulnerable to alteration due to the presence of 3rd party Javascript[9]. This outside server this Javascript was hosted on was vulnerable to the "FREAK attack"[10] – an attacker could force the voter's client to use a weaker, more easily broken cipher suite

Many websites rely on third party ads for revenue or analytics services to better understand their audience, and many of these services do not support HTTPS. However, even when all resources on a given website are loaded via HTTPS, there can still be security issues. As Kranch and Bonneau pointed out, even when these services do support HTTPS, they rarely support HSTS (and thus might load insecure content via a downgrade to plain HTTP). And even when a site supports HSTS, it rarely supports certificate pinning - thus a rogue certificate authority could attack the user. Finally, when

[8] Michael Kranch and Joseph Bonneau, "Upgrading HTTPS in Mid-Air: An Empirical Study of Strict Transport Security and Key Pinning," NDSS Symposium 2015, *available at:* http://www.jbonneau.com/doc/KB15-NDSS-hsts_pinning_survey.pdf

[9] Vanessa Teague and J. Alex Halderman, "Security flaw in New South Wales puts thousands of online votes at risk," Freedom to Tinker (March 22, 2015), *available at:* https://freedom-to-tinker.com/blog/teaguehalderman/ivote-vulnerability/

[10] Matthew Green, "Attack of the week: FREAK (or 'factoring the NSA for fun and profit')," A Few Thoughts on Cryptographic Engineering (March 3, 2015) *available at:* http://blog.cryptographyengineering.com/2015/03/attack-of-week-freak-or-factoring-nsa.html

a site supports HTTPS *and* certificate pinning, the sites often do not load *all* of their content via sites that use HSTS and pinned certificates[11].

The Interactive Advertising Bureau recently put out a blog post[12] exhorting the advertising industry to better support encryption, stating that 80% of of member ad delivery systems supported HTTPS. However, deployed HTTPS may not be as high as the Interactive Advertising Bureau estimates – for example, researchers at Citizen Lab found that only only 38% of advertisers in the Digital Advertising Alliance's opt-out page supported HTTPS.

III. Best Practices for TLS Implementations

This leads us to our next point: Simply enabling HTTPS is not enough – it must be done properly.

A. Use Strong Algorithms to Ensure Confidentiality

Advertising providers should also ensure that they use a strong cipher – as of this writing, this means 2048-bit RSA or 256-bit ECDSA keys[13]. It should be noted that while providers should not use smaller keys, larger keys may create performance issues and do not provide an appreciable increase in confidentiality and may impair user experience.

B. Use up to date software

Ad providers should enable the latest version of TLS for their ad networks. Ad providers should not support any version of SSL, since they are all vulnerable to a range of attacks.[14] Even the most recent version of SSL (v3.0) is vulnerable and SSL 3.0 uses the RC4 cipher, which is widely known to be insecure. SSL 3.0 allows for the use of a block cipher in CBC mode. The POODLE[15] attack can exploit the fact that SSL 3.0's CBC mode padding is non-deterministic to recover unencrypted message content. Using the POODLE attack, an attacker requires only 256 SSL 3.0 requests to reveal one byte of encrypted messages.

[11] See fn 8.

[12] Brendan Riordan-Butterworth, "Adopting Encryption: the need for HTTPS," IAB Blog (March 25, 2015) *available at:* http://www.iab.net/iablog/2015/03/adopting-encryption-the-need-for-https.html

[13] Ivan Ristic, "SSL/TLS Deployment Best Practices," *available at:* https://www.feistyduck.com/library/openssl-cookbook/online/apA-ssl-tls-deployment-best-practices.html

[14] See: "Deprecating Secure Sockets Layer Version 3.0," IETF Internet-Draft (May 14, 2015), *available at:* https://tools.ietf.org/html/draft-thomson-sslv3-diediedie-00.

[15] Bodo Möller, "This Poodle Bites: Exploiting the SSL 3.0 Fallback," Google Security Blog (October 14, 2014) *available at:* http://googleonlinesecurity.blogspot.co.uk/2014/10/this-poodle-bites-exploiting-ssl-30.html

Current TLS/HTTPS best practice instructs the following[16]:

- The RC4 cipher is insecure and should not be used.
- Likewise the SHA-1 hashing algorithm is thought to be soon insecure and should not not be used in hashes within TLS certificates.
- TLS compression should be disabled to guard against the CRIME attack.
- Finally, ad providers should make sure their TLS deployment prefer cipher suites that provide perfect forward secrecy, so that even if a particular message is able to be decrypted, past messages are not also able to be decrypted.

C. When Possible, Utilize the Latest HTTPS Technologies

In addition to the above baseline standards, advertisers should whenever possible utilize new technologies to guard against the latest threats. For example, to guard against man in the middle attacks and rouge certificate authority incidents such as the one that happened to Google[17], ad providers should utilize HTTP Strict Transport Security (HSTS)[18] and certificate pinning[19].

1. **Use HTTP Strict Transport Security (HSTS) When Possible**
   Simply put, HSTS is a header response request from the server that the browser obeys. When enabled with HSTS a site will refuse any connections over plain HTTP - HTTPS is required, and plain HTTP elements will not load.

   However, HSTS is not a magic bullet - for example, HSTS is not applied by default to subdomains, and failing to set the "`includeSubDomains`" directive in a HSTS policy can lead to mixed content issues.
2. **Support Certificate Pinning When Possible**
   Certificate Pinning[20] allows a domain to specify which certificate authorities are authorized to issue certificates for them.[21] Recall that there are hundreds of certificate authorities operated and controlled by a multitude of countries. Certificate authorities can be compromised, either

[16] See fn 13.

[17] Adam Langley, "Maintaining digital certificate security," Google Security Blog (March 23, 2015), *available at:* http://googleonlinesecurity.blogspot.com/2015/03/maintaining-digital-certificate-security.html

[18] RFC 6797, "HTTP Strict Transport Security (HSTS)," *available at:* https://tools.ietf.org/html/rfc6797

[19] Monica Chew, "Firefox 32 supports Public Key Pinning," Monica at Mozilla (August 26, 2015), *available at:* http://monica-at-mozilla.blogspot.com/2014/08/firefox-32-supports-public-key-pinning.html

[20] RFC 7469, "Public Key Pinning Extension for HTTP," *available at:* https://datatracker.ietf.org/doc/rfc7469/

[21] see fn 19 for a detailed explanation of how certificate pinning works.

through technical means or governmental pressure. Rogue certificates have been issued in the past[22], and will continue to be a threat in the future.

It should be noted that care must be taken when enabling certificate pinning, since improperly configured pinning can render a website inaccessible. That is, by misconfiguring certificate pinning – by, for example, including only one intermediate certificate hash that doesn't correspond to a valid deployed certificate – a client/browser the browser may never encounter a case where it finds a valid certificate, locking the client out of the website for ever (or, at least, until the browser is reinstalled, which is a severe remediation).

3. **Support the Latest TLS Version When Possible**
   Advertisers should always endeavor to use the latest version of TLS (1.2 at the time of this writing).

IV. Economic and Regulatory Effects of Failure to Adopt

As mentioned previously,[23] The Interactive Advertising Board recently put out a position piece that advertising networks must adopt HTTPS. Advertisers who fail to adapt may see customers shift to advertisers who do.

In addition to the threat of lost revenue, advertisers may in the future incur penalties from regulators if their lack of encryption leads to a privacy breach. We believe that as HTTPS becomes more widely used, it will eventually become a standard security practice. When that time comes, sites with improperly configured and/or nonexistent web encryption that results in harm to users (malware installation, identity theft, etc.) could face regulatory scrutiny. Specifically, failure to properly enable HTTPS may in the near future become contemplatable as an unfair business practice under the Federal Trade Commission's Section 5 authority, or may expose companies to liability under state data security statutes or international data protection laws.

V. Conclusion

In closing, while enabling HTTPS is an important first step, however advertisers must take steps to ensure their transition is done properly and that a secure state is maintained. Failing to support HTTPS (and to do so securely) may lead to security and/or privacy breaches. Advertisers who fail to adopt HTTPS may experience lost revenue as advertising and analytics clients seek more secure alternatives.

[22] See fn 17 "Maintaining digital certificate security."
[23] See fn 12.

Finally, as more sites enable HTTPS and organizations such as the Interactive Advertising Bureau, IETF IAB, and W3C TAG agree that enabling HTTPS is a best practice, it is possible that in the future advertisers who fail to *properly* enable HTTPS may be subject to regulatory scrutiny.