Pomcor's Comments on Cybersecurity, Innovations and the Internet Economy

Francisco Corella, PhD
fcorella@pomcor.com

Karen Lewison, MD
kplewison@pomcor.com

July 31, 2011

We appreciate the opportunity to comment on the report entitled Cybersecurity, Innovations and the Internet Economy authored by the Internet Policy Task Force of the Department of Commerce ("the Report"). Our comments are related to:

Policy Recommendation C3: *Through its continued research efforts, the Department of Commerce should begin to specifically promote research and development of technologies that help protect I3S from cyber threats.*

and more specifically to the question:

> *What areas of research are most crucial for the I3S? In particular, what R&D efforts could be used to help the supply chain for I3S and for small and medium-sized businesses?*

We suggest R&D efforts that could greatly strengthen Internet security by improving and broadening the scope of the Transport Layer Security (TLS) protocol, formerly known as Secure Sockets Layer (SSL).

In 1994, the World Wide Web was mostly a collection of linked documents. E-commerce as we know it today did not exist. In 2011, e-commerce sales are projected to reach $197 billion in the US alone. What made e-commerce possible was the introduction in 1994 of Netscape's Secure Sockets Lawyer (SSL) protocol.

The Report states in page 64:

> *However, SSL does not ensure that a user reaches the intended site, so it is not applicable against attacks that redirect users. In other words, SSL site validation is effective, but only if a user reaches the correct destination first.*

This is incorrect. In 1994 it was relatively easy for an attacker to "redirect" a connection from an intended site to the attacker's site, e.g. by compromising the domain name system (DNS) or tampering with Internet routing; today it is even easier to do so, e.g. by spoofing a wireless access point. But TLS protects against such attacks because it provides server authentication. If the user reaches the attacker's site, rather than the intended site, over a TLS connection, the attacker's server will not be able to demonstrate knowledge of a private key associated with a public key bound to the domain name of the intended site by a valid certificate signed by a certificate authority trusted by the user's browser. This will cause the browser to issue a strong warning to the user.[1]

SSL and its successor TLS have been able to provide network security on the Internet despite that fact that the Internet lacked, and, as pointed out by the Report, still lacks, IP security, DNS security, and routing security. Without the ability to set up a secure connection to an authenticated server it would not have been possible to transmit credit card data securely to its intended destination, and e-commerce would not have been possible.

SSL was originally intended to provide secure connections for Hyper-Text Transport Protocol (HTTP) messages, i.e. for the Web. One measure of its success is the fact that TLS is now being used to provide underlying security for many other protocols on the Internet, including: ACAP, COPS, FTP, IMAP, NETCONF, NNTP, POP3, SDP, SMTP and SNMP.

But SSL has not changed much since 1995 (when security issues found in SSL version 2 were fixed in version 3). Protocol development was taken over by the IETF and SSL was renamed TLS; SSL version 3 was succeeded by TLS versions 1.0, 1.1 and 1.2; but only minor changes were made in each version. TLS is far from having reached its full potential, and much work remains to be done.

SSL was designed to provide both server and client authentication with PKI certificates. Server authentication was essential, and an imperfect but practical PKI infrastructure emerged to support it. But SSL/TLS was then the victim of its own success. A secure connection to an authenticated server made it possible for the client to authenticate cheaply by transmitting shared secrets; specifically, by sending a password to authenticate a login request, and an authentication cookie to authenticate subsequent HTTP requests during the login session. Once that was possible, little effort was made in the private sector to take advantage of the stronger security provided by SSL/TLS client authentication.

[1] Perhaps the drafters of the Report were thinking about the fact that double-redirection protocols such as OpenID or OAuth facilitate phishing attacks where a malicious relying party redirects the browser to a malicious identity provider, using HTTP redirection or Javascript submission of a form. In this case, of course, the browser issues no warning, since the intended site, as far as the browser can tell, is the malicious identity provider. But such attacks must be blamed on OpenID and OAuth rather than the browser or TLS.

No practical method was developed to let a user register and later log in to an ordinary Web site using TLS client authentication. Eventually, the proliferation of passwords and the problem of password reuse led the industry to develop a series of identity solutions based on a double-redirection mechanism, ranging from Microsoft Password to OpenID and OAuth, where a relying party redirects the browser to an online identity provider, which asks the user for a password and redirects the browser back to the relying party. Unfortunately these identity solutions facilitate phishing attacks. Furthermore they invade the user's privacy, since the identity provider is informed of every login of the user to a relying party.

No method was developed, either, to authenticate HTTP requests during a login session using TLS client authentication and public key cryptography, instead of using an authentication cookie. This has had negative consequences for security and privacy. With respect to security, there has been over the years a long string of cookie-related vulnerabilities. With respect to privacy, the use of cookies for client authentication has been used as a justification for the existence of cookies, and sometimes as an excuse for their use to track the activities of users on the Web.

Two obstacles hindered the deployment of TLS client authentication in the private sector: the lack of a light-weight, automated method of issuing a credentials to browsers;[2] and the lack of a method by which a relying party can ask the browser to present credentials other than passwords. Not enough effort was made to overcome these obstacles.

SSL was very successful in transferring a fundamental scientific advance, public key cryptography, from the research labs to the Internet. Since then its successor, TLS, has incorporated further advances such as elliptic curve cryptography, the Advanced Encryption Standard (AES) and the NSA-designed family of cryptographic hash functions known as SHA-2. But these further advances are only incremental ones. More fundamental advances such as zero-knowledge proofs of knowledge and other privacy-enhancing technologies have not yet been incorporated into TLS.

Therefore, in answer to the question:

> *What areas of research are most crucial for the I3S?*

we suggest that one such area should encompass further development and more

[2] Certificate issuance has been a cumbersome manual process. Several automated protocols have been devised (CMP, CMC, SCEP) but they are not light-weight and they have not been used for issuing TLS certificates on the Web at large. The HTML5 specification being developed by the World Wide Web Consortium includes a <keygen> tag that can be used to build a light-weight protocol; it is already supported by some browsers and it is currently being used to issue WebID credentials. Neither <keygen> nor any of the above protocols support next-generation, privacy-enhanced credentials such as Idemix anonymous credentials or U-Prove tokens.

effective use of the TLS protocol. In particular, the above considerations motivate research and development efforts aimed at:

1. Integrating advanced cryptographic technologies into TLS to facilitate the use of privacy-enhanced credentials.

2. Developing light-weight methods by which Web sites can issue both PKI certificates and privacy-enhanced credentials to Web browsers automatically.[3]

3. Developing methods of tracking login sessions using TLS client authentication with suitable public key credentials instead of authentication cookies.

4. Developing methods by which relying parties can request submission of one or more identity claims supported by public key certificates and/or privacy-enhanced credentials.

The above list is not exclusive. Additional research and development efforts could be aimed, for example, at:

5. Protecting the confidentiality of the information contained in the TLS client certicate by transmitting the certificate encrypted, rather than in the clear as it is transmitted now.

6. Reducing the performance impact of TLS when used in bandwidth-constrained networks, or over high-latency links such as connections via geostationary satellites.

7. Improving and standardizing the methods used by browsers to manage credentials stored in a file system, smart card, or trusted platform module, and to protect, backup, and sync those credentials.

8. Developing easier and more efficient methods of revoking credentials and checking whether a credential is still valid or has been revoked.

In answer to the question:

> *What R&D efforts could be used to help the supply chain for I3S and for small and medium-sized businesses?*

we suggest that the deliverables of the above research and development efforts should include open source libraries and modules suitable for integration into the supply chain of browsers, relying parties, and identity providers.

The suggested research and development efforts would benefit the Identity Ecosystem envisioned by the National Strategy for Trusted Identities in Cyberspace (NSTIC), and the many Internet protocols that rely on the TLS protocol for their security.

[3] We suggest that TLS could be extended to encompass the execution of credential-issuance protocols in addition to credential-presentation protocols.