The Increasing Importance of Proof-of-Possession to the Web

Michael B. Jones

mbj@microsoft.com – http://self-issued.info/

August 14, 2014

W3C Workshop on Authentication, Hardware Tokens and Beyond

Abstract

A number of different initiatives and organizations are now defining new ways to use proof-of-possession in several kinds of Web protocols. These range from cookies that can't be stolen and reused, identity assertions only usable by a particular party, password-less login, to proof of eligibility to participate. While each of these developments is important in isolation, the pattern of all of them concurrently emerging now demonstrates the increasing importance of proof-of-possession to the Web.

1. Introduction

Proof-of-possession is a means of proving that a party sending a message is in possession of a particular cryptographic key. This is used as a proof that the correct party is sending the message, under the assumption that only that sender has possession the key.

A number of different initiatives and organizations are now defining new ways to use proof-of-possession in several kinds of Web protocols. These range from cookies that can't be stolen and reused, identity assertions only usable by a particular party, password-less login, to proof of eligibility to participate. While each of these developments is important in isolation, the pattern of all of them concurrently emerging now demonstrates the increasing importance of proof-of-possession to the Web.

2. Existing Uses of Proof-of-Possession

Proof-of-possession is used by most Internet users every day without them even knowing it. Proof-of-possession underlies the TLS [RFC 5246] security guarantees provided by HTTPS site certificates [RFC 5280]. This is almost certainly the most common use of proof-of-possession today.

3. Emerging Uses of Proof-of-Possession

3.1. Use for TLS Channel Binding

Dirk Balfanz has written a specification [Channel ID] describing using proof-of-possession for TLS channel binding. Browsers deploying this, as Chrome already does, can replace cookies that are bearer tokens and can be replayed by other browsers, if captured, with cookies that are channel bound, and tied to private key state never released by the browser.

The second benefit of channel binding is that channel ID values can be cryptographically incorporated into higher level protocols, for instance, federation protocols such as OpenID Connect [OpenID Connect], enabling tokens used to also be bound to a particular TLS channel.

3.2. OAuth 2.0 Uses

A number of specifications have been submitted to the IETF OAuth working group that enable the use of proof-of-possession in different OAuth scenarios. These supplement the existing OAuth 2.0 [RFC 6749] functionality in which access tokens and other protocol values are bearer tokens [RFC 6750].

Nat Sakimura wrote a specification [OAuth Code PoP] enabling proof-of-possession for OAuth authorization code values. This alleviates a security vulnerability in iOS and Android devices in which multiple applications can try to register for the same OAuth responses.

John Bradley wrote a specification [OAuth PoP Key Dist] that enables OAuth clients to demonstrate proof-of-possession of a key when accessing an OAuth protected resource, rather than just using a bearer token [RFC 6750].

I wrote a specification [OAuth JWT PoP] that defines a representation of a proof key in a JSON Web Token (JWT) [JWT]. This is already in production in some applications, including XBOX One.

Phil Hunt wrote a specification [OAuth PoP Architecture] describing the security characteristics of the use of proof-of-possession in specific OAuth 2.0 scenarios.

In summary, proof-of-possession is the next major area of new work for the IETF OAuth working group.

3.3. Use for Login

Another emerging use is logging in by proving possession of a private key, rather than through use of a password. This private key can be stored either in platform secure storage, such as a TPM, or on other secure devices, such as smart cards or other devices with secure hardware. The FIDO Alliance [FIDO] is in the process of creating a number of specifications for this use case.

3.4. Use to Prove Eligibility to Participate

An emerging use case in the W3C WebCrypto working group [WebCrypto Key Discovery] is using private keys held securely on a device. These keys can be used, for instance, to prove that the device holding the key is a legitimate participant in a particular online interaction.

A clear next step for the WebCrypto working group is enabling JavaScript applications to discover and use secure platform keys, including keys used for proof-of-possession.

4. Conclusions

While passwords and bearer tokens are commonly used on the Web today, their limitations are well known and their security vulnerabilities continue to result in breaches and compromises. The number of independent initiatives working on enabling proof-of-possession at present demonstrates the increasing importance of proof-of-possession for the Web, and gives some hope that the days of relying primarily on passwords and bearer tokens may be behind us within a few years.

References

[Channel ID] *Transport Layer Security (TLS) Channel IDs*, June 2013. http://tools.ietf.org/html/draft-balfanz-tls-channelid-01.

[FIDO] FIDO Alliance, July 2014. http://fidoalliance.org/.

[JWT] *JSON Web Token (JWT)*, July 2014. http://tools.ietf.org/html/draft-ietf-oauth-json-web-token.

[OAuth Code PoP] *OAuth Symmetric Proof of Possession for Code Extension*, April 2014. http://tools.ietf.org/html/draft-sakimura-oauth-tcse.

[OAuth JWT PoP] Proof-Of-Possession Semantics for JSON Web Tokens (JWTs), July 2014. http://tools.ietf.org/html/draft-ietf-oauth-proof-of-possession.

[OAuth PoP Architecture] *OAuth 2.0 Proof-of-Possession (PoP) Security Architecture*, July 2014. http://tools.ietf.org/html/draft-ietf-oauth-pop-architecture.

[OAuth PoP Key Dist] *OAuth 2.0 Proof-of-Possession: Authorization Server to Client Key Distribution*, July 2014. http://tools.ietf.org/html/draft-ietf-oauth-pop-key-distribution.

[OpenID Connect] *OpenID Connect Core 1.0*, February 2014. http://openid.net/specs/openid-connect-core-1_0.html.

[RFC 5246] *The Transport Layer Security (TLS) Protocol Version 1.2*, August 2008. http://tools.ietf.org/html/rfc5246.

[RFC 5280] Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, May 2008. http://tools.ietf.org/html/rfc5280.

[RFC 6749] *The OAuth 2.0 Authorization Framework*, October 2012. http://tools.ietf.org/html/rfc6749.

[RFC 6750] *The OAuth 2.0 Authorization Framework: Bearer Token Usage*, October 2012. http://tools.ietf.org/html/rfc6750.

[WebCrypto Key Discovery] *WebCrypto Key Discovery*, August 2013. http://www.w3.org/TR/webcrypto-key-discovery/.