Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 30, 2015

V. Dukhovni
Two Sigma

October 27, 2014

Opportunistic Security: Some Protection Most of the Time
draft-dukhovni-opportunistic-security-05

Abstract

This document defines the concept "Opportunistic Security" in the context of communications protocols. Protocol designs based on Opportunistic Security use encryption even when authentication is not available, and use authentication when possible, thereby removing barriers to the widespread use of encryption on the Internet.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

1.1. Background

Historically, Internet security protocols have emphasized comprehensive "all or nothing" cryptographic protection against both passive and active attacks. With each peer, such a protocol achieves either full protection or else total failure to communicate (hard fail). As a result, operators often disable these security protocols when users have difficulty connecting, thereby degrading all communications to cleartext transmission.

Protection against active attacks requires authentication. The ability to authenticate any potential peer on the Internet requires an authentication mechanism that encompasses all such peers. No IETF standard for authentication scales as needed and has been deployed widely enough to meet this requirement.

The Public Key Infrastructure (PKI) model employed by browsers to authenticate web servers (often called the "Web PKI") imposes cost and management burdens that have limited its use. With so many Certification Authorities (CAs), not all of which everyone is willing to trust, the communicating parties don't always agree on a mutually trusted CA. Without a mutually trusted CA, authentication fails, leading to communications failure in protocols that mandate authentication. These issues are compounded by operational difficulties. For example, a common problem is for site operators to forget to perform timely renewal of expiring certificates. In Web PKI interactive applications, security warnings are all too frequent, and end-users learn to actively ignore security problems, or site administrators decide that the maintenance cost is not worth the benefit so they provide a cleartext-only service to their users.

The trust-on-first-use (TOFU) authentication approach assumes that an unauthenticated public key obtained on first contact (and retained for future use) will be good enough to secure future communication. TOFU-based protocols do not protect against an attacker who can hijack the first contact communication and require more care from the end-user when systems update their cryptographic keys. TOFU can make it difficult to distinguish routine key management from a malicious attack.

DNS-Based Authentication of Named Entities (DANE [RFC6698]) defines a way to distribute public keys bound to DNS names. It can provide an alternative to the Web PKI. DANE needs to be used in conjunction with DNSSEC [RFC4033]. At the time of writing, DNSSEC is not sufficiently widely deployed to allow DANE to authenticate all potential peers. Protocols that mandate authenticated communication cannot yet generally do so via DANE (at the time of writing).

The lack of a global key management system means that for many protocols, only a minority of communications sessions can be predictably authenticated. When protocols only offer a choice between authenticated-and-encrypted communication, or no protection, the result is that most traffic is sent in cleartext. The fact that most traffic is not encrypted makes pervasive monitoring easier by making it cost-effective, or at least not cost-prohibitive (see [RFC7258] for more detail).

For encryption to be used more broadly, authentication needs to be optional. The use of encryption defends against pervasive monitoring and other passive attacks. Even unauthenticated, encrypted communication (defined below) is preferable to cleartext.

1.2. A New Perspective

This document describes a change of perspective. Until now, the protocol designer has viewed protection against both passive and active attacks as the default, and anything short of that as "degraded security" or a "fallback". The new viewpoint is that without specific knowledge of peer capabilities (or explicit configuration or direct request of the application), the default protection is no protection, and anything more than that is an improvement.

"Opportunistic Security" (OS) is defined as the use of cleartext as the baseline communication security policy, with encryption and authentication negotiated and applied to the communication when available.

Cleartext, not comprehensive protection, is the default baseline. An OS protocol is not falling back from comprehensive protection when that protection is not supported by all peers; rather, OS protocols aim to use the maximum protection that is available. (At some point in time for a particular application or protocol all but a negligible fraction of peers might support encryption. At that time, the baseline security might be raised from cleartext to always require encryption, and only authentication would have to be done opportunistically.)

To achieve widespread adoption, OS must support incremental deployment. Incremental deployment implies that security capabilities will vary from peer to peer, perhaps for a very long time. OS protocols will attempt to establish encrypted communication whenever both parties are capable of such, and authenticated communication if that is also possible. Thus, use of an OS protocol may yield communication that is authenticated and encrypted, unauthenticated but encrypted, or cleartext. This last outcome will occur if not all parties to a communication support encryption (or if an active attack makes it appear that this is the case).

When less than complete protection is negotiated, there is no need to prompt the user with "your security may be degraded, please click OK" dialogs. The negotiated protection is as good as can be expected. Even if not comprehensive, it is often better than the traditional outcome of either "no protection" or "communications failure".

OS is not intended as a substitute for authenticated, encrypted communication when such communication is already mandated by policy (that is, by configuration or direct request of the application) or is otherwise required to access a particular resource. In essence, OS is employed when one might otherwise settle for cleartext. OS protocols never preempt explicit security policies. A security administrator may specify security policies that override OS. For example, a policy might require authenticated, encrypted communication, in contrast to the default OS security policy.

In this document, the word "opportunistic" carries a positive connotation. Based on advertised peer capabilities, an OS protocol uses as much protection as possible. The adjective "opportunistic" applies to the adaptive choice of security mechanisms peer by peer. Once that choice is made for a given peer, OS looks rather similar to other designs that happen to use the same set of mechanisms.

The remainder of this document provides definitions of important terms, sets out the OS design principles, and provides an example of an OS design in the context of communication between mail relays.

2. Terminology

Trust on First Use (TOFU): In a protocol, TOFU calls for accepting and storing a public key or credential associated with an asserted identity, without authenticating that assertion. Subsequent communication that is authenticated using the cached key or credential is secure against an MiTM attack, if such an attack did not succeed during the vulnerable initial communication. The SSH protocol [RFC4251] in its commonly deployed form makes use of TOFU. The phrase "leap of faith" (LoF, [RFC4949]) is sometimes used as a synonym.

Authenticated, encrypted communication: Encrypted communication using a session establishment method in which at least the initiator (or client) authenticates the identity of the acceptor (or server). This is required to protect against both passive and active attacks. Mutual authentication, in which the server also authenticates the client, plays a role in mitigating active attacks when the client and server roles change in the course of a single session.

Unauthenticated, encrypted communication: Encrypted communication using a session establishment method that does not authenticate the identities of the peers. In typical usage, this means that the initiator (client) has not verified the identity of the target (server), making MiTM attacks possible.

Perfect Forward Secrecy (PFS): As defined in [RFC4949].

Man-in-the-Middle (MiTM) attack: As defined in [RFC4949].

3. Opportunistic Security Design Principles

OS provides a near-term approach to counter passive attacks by removing barriers to the widespread use of encryption. OS offers an incremental path to authenticated, encrypted communication in the future, as suitable authentication technologies are deployed. OS promotes the following design principles:

Coexist with explicit policy: Explicit security policies preempt OS. Opportunistic security never displaces or preempts explicit policy. Many applications and types of data are too sensitive to use OS, and more traditional security designs are appropriate in such cases.

Prioritize communication: The primary goal of OS is to not impede communication while maximizing the deployment of usable security. OS protocols need to be deployable incrementally, with each peer

configured independently by its administrator or user. With OS, communication is still possible even when some peers support encryption or authentication and others do not.

Maximize security peer by peer: OS protocols use encryption when it is mutually supported. OS protocols enforce peer authentication when an authenticated out-of-band channel is available to provide the requisite keys or credentials. In general, communication should be at least encrypted. OS should employ Perfect Forward Secrecy (PFS) wherever possible in order to protect previously recorded encrypted communication from decryption even after a compromise of long-term keys.

No misrepresentation of security: Unauthenticated, encrypted communication must not be misrepresented to users or in application logs of non-interactive applications as equivalent to authenticated, encrypted communication.

An OS protocol first determines the capabilities of the peer with which it is attempting to communicate. Peer capabilities may be discovered by out-of-band or in-band means. (Out-of-band mechanisms include the use of DANE records or cached keys or credentials acquired via TOFU. In-band determination implies negotiation between peers.) The capability determination phase may indicate that the peer supports authenticated, encrypted communication; unauthenticated, encrypted communication; or only cleartext communication.

Encryption is used to mitigate the risk of passive monitoring attacks, while authentication is used to mitigate the risk of active man-in-the-middle (MiTM) attacks. When encryption capability is advertised over an insecure channel, MiTM downgrade attacks to cleartext may be possible. Since encryption without authentication only mitigates passive attacks, this risk is consistent with the expected level of protection. For authentication to protect against MiTM attacks the peer capability advertisements that convey support for authentication need to be over an out-of-band authenticated channel that is itself resistant to MiTM attack.

Opportunistic security protocols may hard-fail with peers for which a security capability fails to function as advertised. Security services that work reliably (when not under attack) are more likely to be deployed and enabled by default. It is vital that the capabilities advertised for an OS-compatible peer match the deployed reality. Otherwise, OS systems will detect such a broken deployment as an active attack and communication may fail. This might mean that advertised peer capabilities are further filtered to consider only those capabilities that are sufficiently operationally reliable.

Capabilities that can't be expected to work reliably should be treated by an OS protocol as "not present" or "undefined".

With unauthenticated, encrypted communication, OS protocols may employ more liberal settings than would be best-practice when security is mandated by policy. Some legacy systems support encryption, but implement only outdated algorithms or protocol versions. Compatibility with these systems avoids the need to resort to cleartext fallback.

For greater assurance of channel security, an OS protocol may enforce more stringent cryptographic parameters when the session is authenticated. For example, the set of enabled Transport Layer Security (TLS [RFC5246]) cipher suites might exclude deprecated algorithms that would be tolerated with unauthenticated, encrypted communication.

OS protocols should produce authenticated, encrypted communication when authentication of the peer is "expected". Here, "expected" means a determination via a downgrade-resistant method that authentication of that peer is expected to work. Downgrade-resistant methods include: validated DANE DNS records, existing TOFU identity information, and manual configuration. Such use of authentication is "opportunistic", in that it is performed when possible, on a per-session basis.

When communicating with a peer that supports encryption but not authentication, any authentication checks enabled by default must be disabled or configured to soft-fail in order to avoid unnecessary communications failure or needless downgrade to cleartext.

Cleartext is supported for backwards compatibility with systems already deployed. Even when cleartext needs to be supported, protocol designs based on Opportunistic Security prefer to encrypt, employing cleartext only with peers that do not appear to be encryption capable.

4. Example: Opportunistic TLS in SMTP

Most Message Transfer Agents (MTAs, [RFC5598]) support the STARTTLS ([RFC3207]) ESMTP extension. MTAs acting as SMTP ([RFC5321]) clients generally support cleartext transmission of email. They negotiate TLS encryption when the SMTP server announces STARTTLS support. Since the initial ESMTP negotiation is not cryptographically protected, the STARTTLS advertisement is vulnerable to MiTM downgrade attacks.

Recent reports from a number of large providers (e.g., [fb-starttls] and [goog-starttls]) suggest that the majority of SMTP email transmission on the Internet is now encrypted, and the trend is toward increasing adoption.

Various MTAs that advertise STARTTLS exhibit interoperability problems in their implementations. As a work-around, it is common for a client MTA to fall back to cleartext when the TLS handshake fails, or when TLS fails during message transmission. This is a reasonable trade-off, since STARTTLS only protects against passive attacks. In the absence of an active attack TLS failures are generally one of the known interoperability problems.

Some client MTAs employing STARTTLS abandon the TLS handshake when the server MTA fails authentication, and immediately start a cleartext connection. Other MTAs have been observed to accept unverified self-signed certificates, but not expired certificates; again falling back to cleartext. These and similar behaviors are NOT consistent with OS principles, since they needlessly fall back to cleartext when encryption is clearly possible.

Protection against active attacks for SMTP is described in [I-D.ietf-dane-smtp-with-dane]. That document introduces the terms "Opportunistic TLS" and "Opportunistic DANE TLS", and is consistent with the OS design principles defined in this document. With "Opportunistic DANE TLS", authenticated, encrypted communication is enforced with peers for which appropriate DANE records are present. For the remaining peers, "Opportunistic TLS" is employed as before.

5. Operational Considerations

OS protocol designs should minimize the possibility of failure of negotiated security mechanisms. OS protocols may need to employ "fallback", to work-around a failure of a security mechanisms that is found in practice to encounter interoperability problems. The choice to implement or enable fallback should only be made in response to significant operational obstacles.

When protection only against passive attacks is negotiated over a channel vulnerable to active downgrade attacks, and the use of encryption fails, a protocol might elect non-intrusive fallback to cleartext. Failure to encrypt may be more often a symptom of an interoperability problem than an active attack. In such a situation occasional fallback to cleartext may serve the greater good. Even though some traffic is sent in the clear, the alternative is to ask the administrator or user to manually work-around such interoperability problems. If the incidence of such problems is non-negligible, the user or administrator might find it more expedient to just disable Opportunistic Security.

6. Security Considerations

OS supports communication that is authenticated and encrypted, unauthenticated and encrypted, or cleartext. And yet the security provided to communicating peers is not reduced by the use of OS because the default OS policy employs the best security services available based on the capabilities of the peers, and because explicit security policies take precedence over the default OS policy. OS is an improvement over the status quo; it provides better security than the alternative of providing no security services when authentication is not possible (and not strictly required).

While the use of OS is preempted by a non-OS explicit policy, such a non-OS policy can be counter-productive when it demands more than many peers can in fact deliver. Non-OS policy should be used with care, lest users find it too restrictive and act to disable security entirely.

7. Acknowledgements

I would like to thank Dave Crocker, Peter Duchovni, Paul Hoffman, Benjamin Kaduk, Steve Kent, Scott Kitterman, Pete Resnick, Martin Thomson, Nico Williams, Paul Wouters and Stephen Farrell for their many helpful suggestions and support.

8. References

8.1. Normative References

[RFC3207] Hoffman, P., "SMTP Service Extension for Secure SMTP over Transport Layer Security", RFC 3207, February 2002.

[RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, March 2005.

[RFC4251] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Protocol Architecture", RFC 4251, January 2006.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

[RFC5321] Klensin, J., "Simple Mail Transfer Protocol", RFC 5321, October 2008.

[RFC6698] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA", RFC 6698, August 2012.

8.2. Informative References

[I-D.ietf-dane-smtp-with-dane] Dukhovni, V. and W. Hardaker, "SMTP security via opportunistic DANE TLS", draft-ietf-dane-smtp-with-dane-13 (work in progress), October 2014.

[RFC4949] Shirey, R., "Internet Security Glossary, Version 2", RFC 4949, August 2007.

[RFC5598] Crocker, D., "Internet Mail Architecture", RFC 5598, July 2009.

[RFC7258] Farrell, S. and H. Tschofenig, "Pervasive Monitoring Is an Attack", BCP 188, RFC 7258, May 2014.

[fb-starttls] Facebook, "The Current State of SMTP STARTTLS Deployment", May 2014, <https://www.facebook.com/notes/protect-the-graph/the-current-state-of-smtp-starttls-deployment/1453015901605223>.

[goog-starttls] Google, "Safer email - Transparency Report - Google", June 2014, <https://www.google.com/transparencyreport/saferemail/>.

Author's Address

Viktor Dukhovni
Two Sigma

Email: ietf-dane@dukhovni.org