Network Working Group
Internet-Draft
Intended status: Informational
Expires: October 10, 2014

S. Kent
BBN Technologies

April 8, 2014

Opportunistic Security as a Countermeasure to Pervasive Monitoring
draft-kent-opportunistic-security-01

Abstract

This document was prepared as part of the IETF response to concerns about "pervasive monitoring" (PM) as articulated in [I-D.farrell-perpass-attack]. It begins by describing the current criteria (discussed at the STRINT workshop [STRINT]) for addressing concerns about PM. It then examines terminology that has been used in IETF standards (and in academic publications) to describe encryption and key management techniques, with a focus on authentication vs. anonymity. Based on this analysis, it propose a new term, "opportunistic security" to describe a goal for IETF security protocols, one countermeasure to pervasive monitoring.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on October 10, 2014.

Table of Contents

1. Removing Impediments to Using Encryption in the Internet

Recent discussions in the IETF about countering pervasive monitoring (PM) have focused on increasing the use of encryption. In many contexts, it is perceived that requiring authentication as part of establishing an encrypted session is the major impediment to more widespread use of encryption. Many IETF security protocols commonly call for such authentication as part of establishing an encrypted session. Thus much of the current flurry of activity focuses on removing this impediment.

The term "opportunistic encryption" has been used frequently to refer to newly proposed techniques for encouraging more widespread use of encryption. However, this term has not always been used consistently, and the term already has a precise meaning in the IETF [RFC4322]. The next section of this document examines terminology relevant to the topic, and suggests use of a new term: "opportunistic security", a compromise based on the many terms that have been offered. It also proposes a definition for this term, based on principles adopted during the STRINT Workshop.

Opportunistic Security (for realtime communication) is defined as a set of mechanisms for a security protocol that exhibit the following characteristics:

1. It is invisible to users, and, more broadly, to applications that initiate sessions. (Lack of visibility is considered critical, so that users do not become confused by the variability in the set of security services they are being afforded. Similarly, an application that has not mandated explicit use of security

protocol benefits from opportunistic security, but OUGHT NOT [RFC6919] make decisions on how to behave based on the success or failure of OS mechanisms).

2. Opportunistic security is not intended to be a substitute for authenticated, integrity-protected encryption when that set of security services can be provided to a user. For example, if a user can establish a server-authenticated (see definition later) TLS session with a financial institution today, the user should continue to do so. This suggests that users (applications) MUST still be able to explicitly invoke security protocols.

3. Opportunistic security will make use of perfect forward secrecy (PFS) for key agreement. (PFS is desired because it affords protection against a range of attacks that go beyond simple, passive wiretapping. IKE [RFC5996] has offered this capability, though it does not appear to be commonly used.)

4. Crypto-based authentication is a desired, though not mandatory, feature. (Authentication comes in many flavors, as discussed in Section 2. Authentication is desirable because it offers protection against a range of active attacks, including MiTM, that could cause a user to communicate with a party impersonating the intended communication target. Some security protocols mandate two-way crypto-based authentication, by default, e.g., IKE. TLS [RFC5246] and SSH [RFC4253] typically make use of 1-way (server-based) crypto-based authentication, although they also support client-based crypto-based authentication. Because the success or failure of opportunistic security is not to be signaled to the initiator of a session, the user will not know whether the target of the session has been authenticated.

5. Detection of a man-in-the-middle (MiTM) attacks is a desired, thought not mandatory, feature. If crypto-based authentication cannot be achieved, it may still be possible to detect a MiTM. MiTM detection is considered a lower priority than (crypto-based) authentication, and is to be pursued only if the former security service is not available (or fails).

6. In some contexts, human-perceptible delays in session/connection establishment might discourage use of OS. In such contexts, authentication and MiTM detection SHOULD take place after an encrypted session is established. This ordering implies that some data may be transmitted prior to authentication or detection of a MiTM. In a context where delays imposed by performing authentication are not considered onerous, authentication MAY be attempted prior to enabling transmission. In such contexts all

application traffic will be afforded authentication (and, typically, integrity) if available.

7. Because opportunistic security entails a new key management paradigm, it requires new capabilities by peers. Thus, until OS is universally adopted, an attempt to execute opportunistic security may fail, and the session will fallback to plaintext communication. Since an attempt to use opportunistic security is not communicated to users or applications, falling back to the status quo, i.e., plaintext communication, is a reasonable strategy.

2. Why not "Opportunistic Encryption"?

The term "opportunistic encryption" has become very widely used to describe a range of key management (for encryption) techniques in the IETF since the second half of 2013. However, it is not a new term. The term was coined by H. Spencer and D. Redelmeier in 2001 [FreeSwanOE], and entered into the IETF vocabulary by Michael Richardson in "Opportunistic Encryption using the Internet Key Exchange (IKE)" an Informational RFC [RFC4322]. In this RFC the term is defined as:

> ... the process of encrypting a session with authenticated knowledge of who the other party is without prearrangement.

This definition above is a bit opaque. The introduction to [RFC4322] provides a clearer description of the term, by stating the following goal:

> The objective of opportunistic encryption is to allow encryption without any pre-arrangement specific to the pair of systems involved.

Later the RFC notes:

> Opportunistic encryption creates tunnels between nodes that are essentially strangers. This is done without any prior bilateral arrangement.

The reference to "prior bilateral arrangement" is relevant to IPsec but not to most other IETF security protocols. If every pair of communicating entities were required to make prior bilateral arrangements to enable encryption between them, a substantial impediment would exist to widespread use of encryption. However, other IETF security protocols define ways to enable encryption that do not require prior bilateral arrangements. Some of these protocols require that the target of a communication make available a public

key, for use by any initiator of a communication; an example of a prior unilateral arrangement. The essential difference between IPsec and most other IETF security protocols is that IPsec intrinsically incorporates access control; other IETF security protocols do not.

The definition provided in [RFC4322] is specific to the IPsec [RFC4301] context and ought not be used to describe the goals noted in Section 1 above, as a countermeasure to PM. Because IPsec implements access controls, it requires explicit specification (by each peer) of how to process all traffic that crosses an "IPsec boundary" (inbound and outbound). Traffic is either discarded, permitted to pass w/o IPsec protection, or protected using IPsec. The goal of opportunistic encryption (as per [RFC4322]) is to enable IPsec protected communication without a priori configuration of access control database entries at each peer (hence, bilateral). Opportunistic encryption still calls for each party to identify the other, using IKE v2 [RFC5996] (equivalently, IKE v1 [RFC2409]) authentication mechanisms, so it is not an unauthenticated key management approach. Also note that [RFC4322] describes opportunistic encryption relative to IKE, as it should; IPsec implements encryption using ESP [RFC4303]. ESP usually provides data integrity and authentication, as well as confidentiality, thus the phrase opportunistic encryption is unduly narrow relative to the anti-PM goal.

[RFC4322] also defines anonymous encryption:

> Anonymous encryption: the process of encrypting a session without any knowledge of who the other parties are. No authentication of identities is done.

Thus, in [RFC4322], the term anonymous encryption refers to encrypted communication where neither party is authenticated to the other. Also note that the definition above refers to "the process of encrypting a session ..." In fact, it is the key management process that causes an encrypted session to be authenticated, or not, based on credentials such as public keys, public key certificates, etc.

An examination of about 70 papers published in ACM, IEEE, and other security conference proceedings identified numerous uses of the terms opportunistic and anonymous encryption. Most, though not all, of the papers used the terms opportunistic encryption and anonymous encryption as defined in [RFC4322], but in some papers the terminology was unclear or inconsistent with the [RFC4322] definition.

Wikipedia [wikipedia] uses a somewhat different definition for opportunistic encryption. Wikipedia [wikipedia] provides the following definition:

> Opportunistic encryption refers to any system that, when connecting to another system, attempts to encrypt the communications channel otherwise falling back to unencrypted communications. This method requires no pre-arrangement between the two systems.

This definition shares some aspects of the [RFC4322] definition, but it is not equivalent; it makes no mention of authentication or access control, two essential aspects of opportunistic encryption as per [RFC4322]. The definition is similar to some of the goals listed in Section 1, but not to all of them. The article goes on to cite examples of what it considers to be opportunistic encryption (citing use of self-signed certificates in TLS), and in so doing contradicts the concise definition above. Given the questionable scholarship of the article, and its inconsistent use of the term with a range of examples, it does not merit consideration when choosing a term to describe the anti-PM mechanisms the IETF is developing.

Thus, the recent penchant for using the term opportunistic encryption to refer to mechanisms that yield unauthenticated sessions is inaccurate, even if popular. Although opportunistic encryption, as described in [RFC4322], did not see widespread use, the effort has resumed (as briefed in late 2013 [OErevisited]) and thus it makes sense to reserve the term for that well-defined context.

The adjective "opportunistic" has caught the imagination of many (at least in the IETF), so it seems desirable to retain that word when selecting a new phrase to describe the goals cited in Section 1. It was observed that these goals encompass more than just encryption. PFS is a key management feature, and the optional (crypto-based) authentication and MiTM detection features are security services [ISO.7498-2.1988]. Thus the term "opportunistic security" is proposed here as the (more accurate) term to replace opportunistic encryption.

3. Authentication, Key Management and Existing IETF Protocols

As noted above, many IETF security protocols incorporate (crypto-based) authentication as an intrinsic part of key management. IKE normally requires two-way (mutual) authentication of the peers that establish security associations. TLS normally affords server authentication (based on X.509 certificates and the so-called Web PKI), and offers optional support for client authentication based on use of certificates. SSH ([RFC4251], [RFC4252], [RFC4253]) makes use

of a trust on first use (TOFU) approach (aka a leap of faith, see below) for server authentication. Because SSH is most often employed in an enterprise context, reliance on this initial authentication mechanism (for severs) represents a reasonable risk-based design tradeoff. (User authentication in SSH is supported via a wide range of techniques, some of which are cryptographic-based.)

Although, as noted above, many IETF security protocols incorporate 1-way or 2-way crypto-based authentication as part of key management, most also offer options to enable creation of an encrypted session based on 1-way or 2-way unauthenticated key management. For example, TLS typically is used in a fashion that provides server, but not client, authenticated communication. TLS also supports "establishment of sessions" in which neither party (client or server) asserts an identity during the handshake protocol (based on Diffie-Hellman or ECDH key agreement). Thus TLS offers 2-way unauthenticated communication in addition to the common, server-authenticated communication. (The same analysis applies to DTLS [RFC6347].)

In the store-and-forward environment, encrypted S/MIME messages are usually signed. Moreover, the recipient of an S/MIME message is typically identified by a certificate, so the originator specifies to whom the message is directed. Thus, in common use, S/MIME provides 2-way authentication of traffic. However, S/MIME allows transmission of originator-anonymous encrypted messages. First, note that signing of a message by the originator is optional (see Section 3.3 of [RFC5751]). Also, an originator may employ a key agreement algorithm (e.g., Diffie-Hellman), to preserve originator anonymity. (Section 6.2.2 of [RFC5652] notes: "The originatorKey alternative inclues the algorithm identifier and sender's key agreement public key. This alternative permits originator anonymity since the public key is not certified.")

The originator of an S/MIME message directs an encrypted message to a specific recipient (or set of recipients), and typically makes use of a public key associated with the intended recipient to encrypt the content encryption key for the message. If the recipient is identified by a certificate, as is commonly the case, one would view the communication as recipient-authenticated. However, if the public key associated with a recipient is not conveyed via a validated certificate, then the recipient would not be (crypto) authenticated in the traditional sense. S/MIME calls for implementations to cache capabilities information about senders (section 2.7.2 of [RFC5751]), to facilitate this form of inband cryptographic data transfer. This represents an alternative way for a prospective recipient to convey public key info. (Note, this procedure is at odds with the definition of opportunistic encryption, as it calls for a priori,

per-peer configuration of data to enable later encrypted communication!)

4. Anonymous, Pseudonymous, and Unauthenticated

The discussion above used the terms "authenticated" and "unauthenticated" when describing various modes of key management. These different modes achieve different results with respect to identification of participants in a communication. We avoided the term "anonymous" in part because unauthenticated communication, in many contexts does not confer anonymity, per se. If a user does not employ a key management technique that authenticate his/her identity, the user may be required to employ some other form of authentication later in the communication. In such cases the user clearly is not anonymous. Also, the IP address and other characteristics of the user may be gleaned from a communication, independent of the use of explicit authentication mechanisms, including those associated with key management. Finally, we avoid using the term "unauthenticated encryption" because "authenticated encryption" is a well-defined term in the crypto community. Instead we use the terms "unauthenticated encrypted communication" and "authenticated encrypted communication" as appropriate.

Another reason to avoid the term "anonymous" here is because it is often confused with mechanisms that offer pseudonymous communication. Pseudonymity [merriam-webster] implies use of an identifier, but one that represents a "false name" for an entity. Use of pseudonyms is common in some Internet communication contexts. Many Gmail, Yahoo, and Hotmail mail addresses likely represent pseudonyms. A pseudonym is an attractive way to provide unauthenticated communication. A pseudonym typically makes use of the same syntax as a verified identity in authenticated communication, and thus protocols designed to make use of authenticated identities are compatible with use of pseudonyms, to first order.

"Traceable Anonymous Certificate", is an Experimental RFC [RFC5636] that describes a specific mechanism for a Certification Authority (CA) [RFC5280] to issue an X.509 certificate with a pseudonym. The goal of the mechanisms described in that RFC is to conceal a user's identity in PKI-based application contexts (for privacy), but to permit authorities to reveal the true identity (under controlled circumstances). This appears to be the only RFC that explicitly addresses pseudonymous key management; although it uses the term "pseudonym" extensively, it also uses the term "anonymous" more often, treating the two as synonyms.

Self-signed certificates [RFC6818] are often used with TLS in both browser and non-browser contexts. In the HTTPS (browser) context, a

self-signed certificate typically is accepted after a warning has been displayed to a user; the HTTPS ([RFC2818], [RFC6797]) requirement to match a server DNS name against a certificate Subject name does not apply when TLS is employed in non-browser contexts. The Subject name in a self-signed certificate is completely under the control of the entity that issued it, thus this is a trivial way to generate a pseudonymous certificate, without using the mechanisms specified in [RFC5636]. Thus support for pseudonymous encrypted communication is supported in web browsing, as a side effect of this deviation from [RFC2818]. (Some speculate that most self-signed certificates contain accurate user or device IDs; the certificates are used to avoid the costs associated with issuance of certificates by Web PKI CAs.)

Pseudonymous encrypted communication is the result of applying techniques to distribute keys when an authentication exchange is based on a pseudonym, e.g., a self-signed certificate containing a pseudonym. As with unauthenticated encrypted communication, pseudonymous encrypted communication may apply to one or both parties in an encrypted communication. One also can imagine mixed mode communications, e.g., in which unauthenticated encrypted communication is employed by one party and pseudonymous encrypted communication is employed by the other.

As noted earlier, the model for opportunistic security (for realtime communications) is to first establish an encrypted session, using key management that affords PFS, and then attempt to "upgrade" it to an authenticated communication. This is analogous to what IKE v2 [RFC5996] does. As experience with IKE has shown, this creates a DoS vulnerability, i.e., an attacker can cause the target of a session/ connection to expend resources performing key agreement operations prior to authenticating the initiator of the communication. Implementations of opportunistic security will have to address this concern. Opportunistic security designs also will have to address various flavors of downgrade attacks, since opportunistic security will allow unauthenticated or plaintext communication. Even though opportunistic security assumes that a user is not alerted to its use, it may be appropriate to alert a user to such attacks, or provide a means by which a system administrator can become aware of them. The details of how these concerns are addressed probably will be specific to the protocol context in which opportunistic security is implemented.

5. Terminology

The following definitions are derived from the Internet Security Glossary [RFC4949], where applicable.

Authenticated Encrypted Communication An encrypted communication session based on using a key management technique that provides crypto-based authentication, of one or both parties to the communication. The communication may be 1-way or 2-way authenticated.

Authentication The process of verifying a claim that a system or entity has a certain attribute value. In the IETF context, authentication typically refers to verification of an identity claim, relative to some identifier's space. Typical identifier spaces in the Internet include DNS names and [RFC0822] names.

(Data) Confidentiality The security service that prevents information becoming available to unauthorized entities. Encryption is the security mechanism typically used to implement confidentiality.

Content encryption key (CEK) A symmetric cryptographic key used to encrypt/decrypt the content of an S/MIME message. (Sometimes referred to as a message encryption key.)

(Data) Integrity The security service that enables a recipient of a message or a packet to determine if the data has been modified or destroyed in an unauthorized manner.

Key agreement algorithm A key establishment method based on asymmetric cryptography, in which a pair of entities engage in a public exchange of data (public keys and associated data), to generate the same shared secret value. (Thus both entities contribute secret values to the resulting key.) This value is later used to create symmetric keys used for encryption and/or integrity checking. Diffie-Hellman and Elliptic Curve Diffie-Hellman (ECDH) are the most common algorithms used for key agreement as specified in RFCs.

Key transport A key establishment method by which a secret (symmetric) key is generated by one entity and securely sent to another entity. (Thus only one entity contributes secret values to the resulting key.) Key transport may make use of either symmetric or asymmetric cryptographic algorithms. The RSA algorithm is most commonly cited in RFCs as a basis for a public key, key transport mechanism.

Man-in-the-Middle attack (MiTM) A form of active wiretapping attack in which an attacker intercepts and may selectively modify communicated data to masquerade as one of the entities involved in a communication. Masquerading enables the MiTM to violate the

confidentiality and/or the integrity of communicated data passing through it.

Opportunistic Encryption A key management technique that enables authenticated communication between parties, and that does not require a priori, bilateral arrangements. This term is defined only for IPsec.

Opportunistic Security The result of employing a key management technique that attempts to establish an encrypted communication automatically and invisibly to a user. Opportunistic security may attempt to upgrade an encrypted communication to provide authentication (one or two way), and/or to detect MiTM attacks. If opportunistic security is unable to create an encrypted communication, e.g., because the other communicant does not support opportunistic security, unencrypted (plaintext) communication results.

Perfect Forward Secrecy (PFS) For a key management protocol, the property that compromise of long-term keys does not compromise session/traffic/content keys that are derived from or distributed using the long-term keys.

Private key The secret component of a pair of cryptographic keys used for asymmetric cryptography.

Public key The publicly disclosed component of a pair of cryptographic keys used for asymmetric cryptography. The phrase "public key data" includes a public key and any additional parameters required to perform computation using the public key.

Pseudonymous Communication A key management technique that enables pseudonymous communication between parties, e.g., based on use of a self-signed certificate. Pseudonymous communication may be one-way or two-way, depending on details of the key management mechanism employed.

Session A realtime communication between entities.

Shared secret A value derived from a key agreement algorithm and used as an input to generate a content encryption key or traffic encryption key.

Symmetric cryptography A type of cryptography in which the algorithms employ the same key for encryption and decryption, and the key is not publicly disclosed.

Traffic (encryption) key (TEK) A symmetric key used to encrypt/ decrypt traffic carried via an association.

Trust on First Use (TOFU) In a protocol, TOFU typically consists of accepting an asserted identity, without authenticating that assertion, and caching a key or credential associated with the identity. Subsequent communication using the cached key/ credential is secure against a MiTM attack, if such an attack did not succeed during the (vulnerable) initial communication or if the MiTM is not present for all subsequent communications. The SSH protocol makes use of TOFU. The phrase "leap of faith" (LoF) is sometimes used as a synonym.

Unauthenticated Encrypted Communication An encrypted communication session based on using a key management technique that enables unauthenticated communication between parties. The communication may be 1-way or 2-way unauthenticated. If 1-way, the initiator (client) or the target (server) may be anonymous.

6. Acknowledgements

I want to thank David Mandelberg and Edric Barnes for their help in generating this document.

7. Security Considerations

[TBS]

8. References

[FreeSwanOE]
Spencer, H. and D. Redelmeier, "Opportunistic Encryption", May 2001.

[I-D.farrell-perpass-attack]
Farrell, S. and H. Tschofenig, "Pervasive Monitoring is an Attack", draft-farrell-perpass-attack-06 (work in progress), February 2014.

[ISO.7498-2.1988]
International Organization for Standardization, "Information Processing Systems - Open Systems Interconnection Reference Model - Security Architecture", ISO Standard 7498-2, 1988.

[OErevisited]
Wouters, P., "Opportunistic Encryption revisited", November 2013, <http://www.ietf.org/proceedings/88/slides/slides-88-saag-3.pdf>.

[RFC0822] Crocker, D., "Standard for the format of ARPA Internet text messages", STD 11, RFC 822, August 1982.

[RFC2409] Harkins, D. and D. Carrel, "The Internet Key Exchange (IKE)", RFC 2409, November 1998.

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, May 2000.

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

[RFC4252] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Authentication Protocol", RFC 4252, January 2006.

[RFC4253] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Transport Layer Protocol", RFC 4253, January 2006.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

[RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, December 2005.

[RFC4322] Richardson, M. and D. Redelmeier, "Opportunistic Encryption using the Internet Key Exchange (IKE)", RFC 4322, December 2005.

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

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

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, May 2008.

[RFC5636] Park, S., Park, H., Won, Y., Lee, J., and S. Kent, "Traceable Anonymous Certificate", RFC 5636, August 2009.

[RFC5652] Housley, R., "Cryptographic Message Syntax (CMS)", STD 70, RFC 5652, September 2009.

[RFC5751] Ramsdell, B. and S. Turner, "Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.2 Message Specification", RFC 5751, January 2010.

[RFC5996] Kaufman, C., Hoffman, P., Nir, Y., and P. Eronen, "Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 5996, September 2010.

[RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, January 2012.

[RFC6797] Hodges, J., Jackson, C., and A. Barth, "HTTP Strict Transport Security (HSTS)", RFC 6797, November 2012.

[RFC6818] Yee, P., "Updates to the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 6818, January 2013.

[RFC6919] Barnes, R., Kent, S., and E. Rescorla, "Further Key Words for Use in RFCs to Indicate Requirement Levels", RFC 6919, April 1 2013.

[STRINT] "A W3C/IAB workshop on Strengthening the Internet Against Pervasive Monitoring (STRINT)", March 2014, <https://www.w3.org/2014/strint/>.

[merriam-webster] "pseudonymity", March 2014, <http://www.merriam-webster.com/dictionary/pseudonymity>.

[wikipedia] "Opportunistic encryption", November 2013, <http://en.wikipedia.org/w/index.php?title=Opportunistic_encryption&oldid=581222222>.

Author's Address

Stephen Kent
BBN Technologies
10 Moulton St.
Camridge, MA 02138
US

Email: kent@bbn.com