IAB
Internet-Draft
Intended status: Informational
Expires: April 30, 2017

T. Hardie, Ed.

October 27, 2016

Confidentiality in the Face of Pervasive Surveillance
draft-iab-privsec-confidentiality-mitigations-08

Abstract

The IAB has published [RFC7624] in response to several revelations of pervasive attack on Internet communications. This document surveys the most plausible mitigations to those threats currently available to the designers of Internet protocols.

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 April 30, 2017.

Table of Contents

1. Introduction

To ensure that the Internet can be trusted by users, it is necessary for the Internet technical community to address the vulnerabilities exploited in the attacks document in [RFC7258] and the threats described in [RFC7624]. The goal of this document is to describe more precisely the mitigations available for those threats and to lay out the interactions among them should they be deployed in combination.

2. Terminology

This document makes extensive use of standard security and privacy terminology; see [RFC4949] and [RFC6973]. Terms used from [RFC6973] include Eavesdropper, Observer, Initiator, Intermediary, Recipient, Attack (in a privacy context), Correlation, Fingerprint, Traffic Analysis, and Identifiability (and related terms). In addition, we use a few terms that are specific to the attacks discussed in [RFC7624]. Note especially that "passive" and "active" below do not refer to the effort used to mount the attack; a "passive attack" is any attack that accesses a flow but does not modify it, while an "active attack" is any attack that modifies a flow. Some passive attacks involve active interception and modifications of devices, rather than simple access to the medium.

3. Available Mitigations

Given the threat model laid out in [RFC7624], how should the Internet technical community respond to pervasive attack? The cost and risk considerations discussed in it provide a guide to responses. Namely, responses to passive attack should close off avenues for those attacks that are safe, scalable, and cheap, forcing the attacker to mount attacks that expose it to higher cost and risk. Protocols and security measures protecting against active attacks must also limit the impact of compromise and malfeasance by avoiding systems which grant universal credentials.

In this section, we discuss a collection of high-level approaches to mitigating pervasive attacks. These approaches are not meant to be exhaustive, but rather to provide general guidance to protocol designers in creating protocols that are resistant to pervasive attack.

Many of these are basic tools which already exist. As Edward Snowden put it, "properly implemented strong crypto systems are one of the few things you can rely on". The task for the Internet community is to ensure that applications are able to use the strong crypto and other mitigations already available- and that these are properly implemented and commonly turned on. Some of this work will require architectural changes to applications, e.g., in order to limit the information that is exposed to servers. In many other cases, however, the need is simply to make the best use we can of the cryptographic tools we have.

Attack Class	High-level mitigations
Passive observation	Encryption for confidentiality
Passive inference	Path differentiation
Active	Authentication, monitoring
Metadata Analysis	Data Minimization
Static key exfiltration	Encryption with per-session state (PFS)
Dynamic key exfiltration	Transparency, validation of end systems
Content exfiltration	Object encryption, distributed systems

Table 1: Table of Mitigations

3.1. Encryption

The traditional mitigation to passive attack is to render content unintelligible to the attacker by applying encryption, for example, by using TLS or IPsec [RFC5246][RFC4301]. Even without authentication, encryption will prevent a passive attacker from being able to read the encrypted content. Exploiting unauthenticated encryption requires an active attack (man in the middle); with authentication, a key exfiltration attack is required. For cryptographic systems providing forward secrecy, even exfiltration of long-term keys will not compromise data captured under session keys used before the exfiltration.

3.1.1. Forward Secrecy

An encrypted, authenticated session is safe from content-monitoring attacks in which neither end collaborates with the attacker, but can still be subverted by the endpoints. The most common ciphersuites used for HTTPS today, for example, are based on using RSA encryption in such a way that if an attacker has the private key, the attacker can derive the session keys from passive observation of a session. These ciphersuites are thus vulnerable to a static key exfiltration attack - if the attacker obtains the server's private key once, then they can decrypt all past and future sessions for that server.

Static key exfiltration attacks are prevented by including ephemeral, per-session secret information in the keys used for a session. Most IETF security protocols include modes of operation that have this property. These modes are known in the literature under the heading "perfect forward secrecy" (PFS) because even if an adversary has all of the secrets for one session, the next session will use new, different secrets and the attacker will not be able to decrypt it. The Internet Key Exchange (IKE) protocol used by IPsec supports PFS by default [RFC4306], and TLS supports PFS via the use of specific ciphersuites [RFC5246].

3.1.2. Covert Channel Reduction

Dynamic key exfiltration cannot be prevented by protocol means. By definition, any secrets that are used in the protocol will be transmitted to the attacker and used to decrypt what the protocol encrypts. Likewise, no technical means will stop a willing collaborator from sharing keys with an attacker. However, this attack model also covers "unwitting collaborators", whose technical resources are collaborating with the attacker without their owners' knowledge. This could happen, for example, if flaws are built into products or if malware is injected later on.

Standards can also define protocols that provide greater or lesser opportunity for dynamic key exfiltration. Collaborators engaging in key exfiltration through a standard protocol will need to use covert channels in the protocol to leak information that can be used by the attacker to recover the key. Such use of covert channels has been demonstrated for SSL, TLS, and SSH. Any protocol bits that can be freely set by the collaborator can be used as a covert channel, including, for example, TCP options or unencrypted traffic sent before a STARTTLS message in SMTP or XMPP. Protocol designers should consider what covert channels their protocols expose, and how those channels can be exploited to exfiltrate key information.

3.2. Auditing Authentication

As with traditional, limited active attacks, a basic mitigation to pervasive active attack is to enable the endpoints of a communication to authenticate each other over the encrypted channel. However, attackers that can mount pervasive active attacks can often subvert the authorities on which authentication systems rely.

Thus, in order to make authentication systems more resilient to pervasive attack, it is beneficial to monitor these authorities to detect misbehavior that could enable active attack. For example, DANE and Certificate Transparency both provide mechanisms for detecting when a CA has issued a certificate for a domain name

without the authorization of the holder of that domain name [RFC6962][RFC6698]. Other systems may use external notaries to detect certificate authority mismatch (e.g. Convergence [Convergence]).

3.3. Metadata Minimization

The additional capabilities of a pervasive passive attacker, however, require some changes in how protocol designers evaluate what information is encrypted. In addition to directly collecting unencrypted data, a pervasive passive attacker can also make inferences about the content of encrypted messages based on what is observable. For example, if a user typically visits a particular set of web sites, then a pervasive passive attacker observing all of the user's behavior can track the user based on the hosts the user communicates with, even if the user changes IP addresses, and even if all of the connections are encrypted.

Thus, in designing protocols to be resistant to pervasive passive attacks, protocol designers should consider what information is left unencrypted in the protocol, and how that information might be correlated with other traffic. Some of the data left unencrypted may be considered "metadata" within the context of a single protocol, as it provides adjunct information used for delivery or display, rather than the data directly created or consumed by protocol users. This does not mean it is not useful to attackers, however, and when this metadata is not protected by encryption it may leak substantial amounts of information. Data minimization strategies should thus be applied to any data left unencrypted, whether it be payload or metadata. Information that cannot be encrypted or omited should be be dissociated from other information. For example, the TOR overlay routing network [TOR] anonymizes IP addresses by using multi-hop onion routing.

3.3.1. Length Hiding

One fundamental limitation of encryption is that it exposes the size of the plaintext that protects. A passive attacker can use this to obtain information about the plaintext [CLINIC]. Protocols that use encryption can provide the ability to pad plaintext. This enables control over the size of ciphertext by endpoints, which can be used to reduce the information available to passive attackers.

3.4. Anonymization

While encryption and authentication protect the security of individual sessions, these sessions may still leak information, such as IP addresses or server names, that a pervasive attacker can use to

correlate sessions and derive additional information about the target. Thus, pervasive attack highlights the need for anonymization technologies, which make correlation more difficult. Typical approaches to anonymization against traffic analysis include:

- Aggregation: Routing sessions for many endpoints through a common mid-point (e.g, an HTTP proxy). The midpoint appears as the origin of the communication when traffic analysis is conducted from points after it, so individual sources cannot be distinguished. If traffic analysis is being conducted prior to the mid-point, all flows appear to be destined to the same point, which leaks very little information. Even when traffic analysis is being performed both before and after the mid-point, simultaneous connections may make it difficult to correlate the traffic going into and out of the mid-point. For this to be effective as a mitigation, traffic to the mid-point must be encrypted and traffic from the mid-point should be.

- Onion routing: Routing a session through several mid-points, rather than directly end-to-end, with encryption that guarantees that each node can only see the previous and next hops. This ensures that the source and destination of a communication are never revealed simultaneously. Note, however, that onion routing anonymity guarantees depend on an attacker being unable to control many of the routing nodes [TorPaper].

- Multi-path: Routing different sessions via different paths (even if they originate from the same endpoint). This reduces the probability that the same attacker will be able to collect many sessions or associate them with the same individual. If, for example, a device has both a cellular and 802.11 interface, routing some traffic across the cellular network and other traffic over the 802.11 interface means that traffic analysis conducted only with one network will be incomplete. Even if conducted in both, it may be more difficult for the attacker to associate the traffic in each network with the other. For this to be effective as a mitigation, signalling protocols which gather and transmit data about multiple interfaces (such as SIP) must be encrypted to avoid the information being used in cross-correlation.

3.5. End-to-End Protection

Content exfiltration has some similarity to the dynamic exfiltration case, in that nothing can prevent a collaborator from revealing what they know, and the mitigations against becoming an unwitting collaborator apply. In this case, however, applications can limit what the collaborator is able to reveal. For example, the S/MIME and PGP systems for secure email both deny intermediate servers access to

certain parts of the message [RFC5750][RFC2015]. Even if a server were to provide an attacker with full access, the attacker would still not be able to read the protected parts of the message.

Mechanisms like S/MIME and PGP are often referred to as "end-to-end" security mechanisms, as opposed to "hop-by-hop" or "end-to-middle" mechanisms like the use of SMTP over TLS. These two different mechanisms address different types of attackers: Hop-by-hop mechanisms protect from attackers on the wire (passive or active), while end-to-end mechanisms protect against attackers within intermediate nodes as well as those on the wire. Even end-to-end mechanisms are not complete protection in themselves, as intermediate nodes can still access some metadata. For example:

- Two users messaging via Facebook over HTTPS are protected against passive and active attackers in the network between the users and Facebook. However, if Facebook is a collaborator in an exfiltration attack, their communications can still be monitored. They would need to encrypt their messages end-to-end in order to protect themselves against this risk.

- Two users exchanging PGP-protected email have protected the content of their exchange from network attackers and intermediate servers, but the header information (e.g., To and From addresses) is unnecessarily exposed to passive and active attackers that can see communications among the mail agents handling the email messages. These mail agents need to use hop-by-hop encryption and traffic analysis mitigation to address this risk.

Mechanisms such as S/MIME and PGP are also known as "object-based" security mechanisms (as opposed to "communications security" mechanisms), since they operate at the level of objects, rather than communications sessions. Such secure object can be safely handled by intermediaries in order to realize, for example, store and forward messaging. In the examples above, the encrypted instant messages or email messages would be the secure objects. Hop-to-hop security mechanisms may be susceptible to downgrade attacks (e.g., STARTTLS-secured SMTP has been downgraded by intermediate network nodes [WaPo-STARTTLS]) in which case end-to-end mechanisms are advised.

The mitigations to the content exfiltration case regard participants in the protocol as potential passive attackers themselves, and apply the mitigations discussed above with regard to passive attack. Information that is not necessary for these participants to fulfill their role in the protocol can be encrypted, and other information can be anonymized.

4. Interplay among Mechanisms

One of the key considerations in selecting mitigations is how to manage the interplay among different mechanisms. Care must be taken to avoid situations where a mitigation is rendered fruitless because of mechanisms which working at a different time scale or with a different aim.

The tools that we currently have have not generally been designed with all of these mitigations in mind, so they may need elaboration or adjustment to be completely suitable. Thus, managing the integration of one mitigation with the environment in which it is deployed is critical.

As an example, there is work in progress in IEEE 802 to standardize a method for the randomization of MAC Addresses. This work aims to enable the MAC address to vary as the device connects to different networks, or connects at different times. In theory, the randomization will mitigate tracking by MAC address. However, the randomization will be defeated if the adversary can link the randomized MAC address to other identifiers such as the interface identifier used in IPv6 addresses, the unique identifiers used in DHCP or DHCPv6, or unique identifiers used in various link-local discovery protocols.

For mitigations which rely on aggregation to separate the origin of traffic from its destination, care must be taken that the protocol mechanics do not expose origin IP through secondary means. [I-D.ietf-dnsop-edns-client-subnet] for example, documents a method to carry the IP address or subnet of a querying party through a recursive resolver to an authoritative resolver. Even with a truncated IP address, this mechanism increases the likelihood that a pervasive monitor would be able to associate query traffic and responses.

If a client wished to ensure that its traffic did not expose this data, it would need to require that its stub resolver emit any privacy-sensitive queries with a source NETMASK set to 0, as detailed in Section 5.1 of [I-D.ietf-dnsop-edns-client-subnet]. Given that setting this only occasionally might also be used a signal to observers, any client wishing to have any privacy sensitive traffic would, in essence have to emit this for every query. While this would succeed at providing the required privacy, given the mechanism proposed, it would also mean no split-DNS adjustments in response would be possible for the privacy sensitive client.

5. IANA Considerations

This memo makes no request of IANA.

6. Security Considerations

This memorandum describes a series of mitigations to the attacks described in [RFC7258]. No such list could possibly be comprehensive, nor is the attack therein described the only possible attack.

7. Contributors {Contributors}

This document is derived in part from the work initially done on the Perpass mailing list and at the STRINT workshop. Work from Brian Trammell, Bruce Schneier, Christian Huitema, Cullen Jennings, Daniel Borkmann, Martin Thomson, and Richard Barnes is incorporated here, as are ideas and commentary from Jeff Hodges, Phillip Hallam-Baker, and Stephen Farrell.

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC4949] Shirey, R., "Internet Security Glossary, Version 2", FYI 36, RFC 4949, DOI 10.17487/RFC4949, August 2007, <http://www.rfc-editor.org/info/rfc4949>.

[RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, "Privacy Considerations for Internet Protocols", RFC 6973, DOI 10.17487/RFC6973, July 2013, <http://www.rfc-editor.org/info/rfc6973>.

[RFC7258] Farrell, S. and H. Tschofenig, "Pervasive Monitoring Is an Attack", BCP 188, RFC 7258, DOI 10.17487/RFC7258, May 2014, <http://www.rfc-editor.org/info/rfc7258>.

[RFC7624] Barnes, R., Schneier, B., Jennings, C., Hardie, T., Trammell, B., Huitema, C., and D. Borkmann, "Confidentiality in the Face of Pervasive Surveillance: A Threat Model and Problem Statement", RFC 7624, DOI 10.17487/RFC7624, August 2015, <http://www.rfc-editor.org/info/rfc7624>.

8.2. Informative References

[CLINIC] Miller, B., Huang, L., Joseph, A., and J. Tygar, "I Know Why You Went to the Clinic: Risks and Realization of HTTPS Traffic Analysis", March 2014.

[Convergence] M Marlinspike, ., "Convergence Project", August 2011, <http://convergenc.io>.

[I-D.ietf-dnsop-edns-client-subnet] Contavalli, C., Gaast, W., tale, t., and W. Kumari, "Client Subnet in DNS Queries", draft-ietf-dnsop-edns-client-subnet-08 (work in progress), April 2016.

[RFC2015] Elkins, M., "MIME Security with Pretty Good Privacy (PGP)", RFC 2015, DOI 10.17487/RFC2015, October 1996, <http://www.rfc-editor.org/info/rfc2015>.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <http://www.rfc-editor.org/info/rfc4301>.

[RFC4306] Kaufman, C., Ed., "Internet Key Exchange (IKEv2) Protocol", RFC 4306, DOI 10.17487/RFC4306, December 2005, <http://www.rfc-editor.org/info/rfc4306>.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, DOI 10.17487/RFC5246, August 2008, <http://www.rfc-editor.org/info/rfc5246>.

[RFC5750] Ramsdell, B. and S. Turner, "Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.2 Certificate Handling", RFC 5750, DOI 10.17487/RFC5750, January 2010, <http://www.rfc-editor.org/info/rfc5750>.

[RFC6698] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA", RFC 6698, DOI 10.17487/RFC6698, August 2012, <http://www.rfc-editor.org/info/rfc6698>.

[RFC6962] Laurie, B., Langley, A., and E. Kasper, "Certificate Transparency", RFC 6962, DOI 10.17487/RFC6962, June 2013, <http://www.rfc-editor.org/info/rfc6962>.

[STRINT] S Farrell, ., "Strint Workshop Report", April 2014, <https://www.w3.org/2014/strint/draft-iab-strint-report.html>.

[TOR] The Tor Project, "Tor", 2013, <https://www.torproject.org/>.

[TorPaper] Dingledine, R., Mathewson, N., and P. Syverson, "Tor: The Second-Generation Onion Router", 2004, <http://static.usenix.org/event/sec04/tech/full_papers/dingledine/dingledine.pdf>.

[WaPo-STARTTLS] Scola, N. and A. Soltani, "Mobile ISP Cricket was thwarting encrypted emails, researchers find", 2014, <https://www.washingtonpost.com/news/the-switch/wp/2014/10/28/mobile-isp-thwarted-customers-attempts-to-send-encrypted-e-mails-research-finds/>.

Author's Address

Ted Hardie (editor)

Email: ted.ietf@gmail.com