OPSEC Working Group
Internet-Draft
Intended status: Informational
Expires: January 29, 2021

N. Cam-Winget
E. Wang
Cisco Systems, Inc.
R. Danyliw
Software Engineering Institute
R. DuToit
Broadcom

July 28, 2020

Impact of TLS 1.3 to Operational Network Security Practices
draft-ietf-opsec-ns-impact-02

Abstract

Network-based security solutions are used by enterprises, the public sector, internet-service providers, and cloud-service providers to both complement and enhance host-based security solutions. As TLS is a widely deployed protocol to secure communication, these network-based security solutions must necessarily interact with it. This document describes this interaction for current operational security practices and notes the impact of TLS 1.3 on them.

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 https://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 January 29, 2021.

Table of Contents

1. Introduction

Enterprises, public sector organizations, internet service providers and cloud service providers defend their networks and information systems from attacks that originate from inside and outside their networks. These organizations commonly employ security architectures that involve complementary technologies deployed on both endpoints and in the network; and collaborative watch-and-warning practices to realize this defense.

The design of these security architectures and associated practices entails numerous trade-offs. Typically, there is more than one technical approach to realize a particular mitigation, although comparable approaches may have different costs or side-effects. Network-based solutions are often attractive to network administrators because a single network device can:

- provide protection to many hosts and systems at once
- protect systems regardless of their type (e.g., fully patched desktop systems on a modern operating system; unpatched function-specific industrial control system)
- enforce policy on a system even if it is compromised, misconfigured, not under configuration control or had its endpoint protection disabled
- be managed (e.g. updates) and provisioned with resources (e.g. disk and computing) independent of the systems it is protecting
- by itself, a single system may not be able to detect and mitigate threats

In response to the adoption of new technologies, protocols and threats, these security architectures must evolve to remain effective. [RFC8404] documented a need to evolve with the effect of pervasive encryption on operations. This document takes a narrower focus by documenting the interaction of existing network-based security practices with TLS 1.2 [RFC5246] (and earlier) traffic to implement security policy, detection or mitigation of threats; and the impact on these practices with improvements made in TLS 1.3 [RFC8446].

2. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP

14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

Specific operational practices are numbered as "OP-##", operational practice 1 (i.e., OP-1), 2 (i.e., OP-2), etc.

3. How TLS is used to enable Network-Based Security Solutions

Network-based security solutions come in many forms, most commonly as Firewalls, Web Proxies, Intrusion Detection Systems (IDS), Intrusion Prevention Systems (IPS) and Network Security Visibility and Analytics systems. They inspect the network traffic, and then based on their function, log their observation and/or act on the traffic to implement security policy. When these devices act on the network traffic, they are typically deployed inline as middleboxes (e.g. firewalls) or as explicit proxies (e.g. web proxies). If their function is only to observe, they can be deployed either as middleboxes or given access to the network traffic out-of-band (OOB), through the network fabric (e.g., network tap or span port).

Depending on their function, network-based security devices use different degrees of visibility into the TLS traffic. Some operational practices require only access to the unencrypted protocol headers and associated meta-data of the TLS traffic. Other practices require full visibility into the encrypted session (payload).

The practices that inspect only the unencrypted headers and meta-data of TLS, require no special capabilities beyond access to the TLS packets. However, to inspect the encrypted payload of TLS traffic requires a TLS proxy.

A TLS proxy provides visibility and inspection to effectuate security controls without changing the state machine of the TLS Server and TLS Client, or the user experience. The TLS Proxy operates as a transparent hop at the TLS layer in both middlebox and explicit proxy deployments. For the web proxy case, after the client sends an HTTP CONNECT to request a tunnel to the server, the web proxy may insert a TLS Proxy function to proxy the TLS session without awareness by the client or server. The TLS operation afterwards remains the same as a middlebox.

To proxy a TLS session, a TLS Proxy must be able to present a valid X.509 certificate to the TLS client to appear as a valid TLS Server; similarly, the client must be able to validate the X.509 certificate using the appropriate trust anchor for that TLS connection. To achieve this, a deployment must properly provision their systems (TLS Proxies and TLS clients). A TLS Proxy is unable to proxy a PSK based session unless it is on-path and has proxied the session leading to

the PSK. TLS client authentication requires additional provisioning for X.509 certificate on the TLS Server side. It does not have impact on the deployment scenarios though.

4. Changes in TLS 1.3 Relevant to Security Operations

TLS 1.3 introduces a number of protocol design changes to improve security and privacy. However, these enhancements impact current network security operational practices that rely on the protocol behavior of earlier TLS versions.

4.1. Perfect Forward Secrecy (PFS)

TLS 1.2 (and earlier versions) supports static RSA and Diffie-Hellman (DH) cipher suites, which enables the server's private key to be shared with a TLS proxy. [RFC7525] initiated the recommendation of using AEAD cipher suites and specifically decoupling the cipher suite negotiation based on the RSA key transport; this followed with TLS 1.3 explicitly removing support for these cipher suites in favor of supporting only ephemeral mode Diffie-Hellman to provide perfect forward secrecy (PFS). As a result of this enhancement, it would no longer be possible for a server to share a key with the middlebox in advance, which in turn implies that the middlebox cannot gain access to the TLS session data.ss

4.2. Encrypted Server Certificate

TLS 1.2 (and earlier versions) sends the ClientHello, ServerHello and Certificate messages in clear-text. In TLS 1.3, the Certificate message is encrypted whereby hiding the server identity from any intermediary. As a result of this enhancement, it would no longer be possible to observe the server certificate without inspection the encrypted TLS payload.

TLS proxies which implement a selective decryption policy will need to alter their behavior to accommodate TLS 1.3. In TLS 1.2 (and earlier), the proxy could observe the TLS handshake till seeing the clear text server certificate to make the decryption policy decision. For example, a proxy may not be permitted to decrypt certain types of traffic such as those going to a banking and health care service. However, in TLS 1.3, the TLS proxy must participate in both handshakes (i.e., client-to-proxy; and proxy-to-server) in order to view the server certificate. This change will impose a slight increase in load per connection on the proxy.

5. Network Security Operational Practices

Specific network security operational practices applied to TLS 1.2 (and earlier) are described in subsequent sub-sections. They are categorized into the following deployment scenarios:

1. Passive TLS inspection, where the network-based security function is inspecting either the inbound or outbound TLS header or meta-data traffic

2. Outbound TLS Proxy, where a TLS proxy mediates a TLS session originating from a client inside the enterprise administrative domain (and in the same administrative domain as the proxy) towards an entity on the outside

3. Inbound TLS Proxy, where a TLS proxy mediates a TLS session from a client outside the enterprise administrative domain towards an entity on the inside (and in the same administrative domain as the proxy)

Each deployment scenario describes current operational practices. For each operational practice, possible deployment modes (e.g., inline, out-of-band), a description of the practice, and the impact of TLS 1.3 is categorized and explained. The categorized impacts to practices when migrating to TLS 1.3 are as follows:

- no impact - no change in capability or performance is expected with this practice

- no capability impact - no change in capability is expected; but there may be a performance or implementation change required for this practice

- reduced effectiveness - this practice will not be as effective on TLS 1.3 traffic

- alternative approach required - this practice will not work with TLS 1.3 traffic

It should be noted that [ECH] will further reduce the effectiveness (passive inspection) or prevent certain practices (outbound proxy) from being deployed. More study is required in this area.

5.1. Passive TLS Inspection

Passive TLS inspection is the deployment scenario where a network security device passively inspects inbound or outbound TLS traffic to make visibility inferences or take policy actions. The network

security device examines only the unencrypted TLS protocol headers and does not have access to the encrypted content of the payload.

The TLS proxy deployment scenarios may also incorporate these practices.

5.1.1. **OP-1.** Acceptable Use Policy (AUP) Enforcement (via header inspection).

Deployment mode: inline

A firewall or web proxy restricts a client in the same administrative domain from accessing sites or services outside that domain per an acceptable use policy. The identification of the destination server is performed through the inspection of either the SNI field in the TLS ClientHello message from the client; or by extracting the server identity from the Common Name (CN) or Subject Alternative Name (SAN) fields of an X.509 certificate that is presented in the server's Certificate TLS message. This data is used for domain categorization or application identification.

This meta-data can also inform decryption eligibility decisions by a firewall, in OP-4. For instance, a firewall may bypass traffic decryption for a connection destined to a healthcare web service due to privacy compliance requirements.

TLS 1.3 considerations: reduced effectiveness. Per Section 4.2, domain categorization and application identification will be limited to IP address and SNI information (beyond additional correlation possible with other means such as DNS).

While an SNI is mandatory in TLS 1.3, there is no guarantee that the server responding is the one indicated in the SNI from the client. A SNI alone, without comparison of the server certificate, does not provide reliable information about the server that the client is attempting to reach. Where a client has been compromised by malware, it may present an innocuous SNI to bypass protective filters (e.g., to reach a command and control server), and this will be undetectable under TLS 1.3.

5.1.2. **OP-2.** Network Behavior Analytics

Deployment mode: inline and out-of-band

Network behavior analysis and machine learning engines in IDSs, IPSs and firewalls observe the cleartext fields of the TLS handshake (e.g., session cipher suites) and conducts traffic analysis by observing encrypted record sizes, packet rates and their inter-

arrival times, and similar outer connection behavior. They match encrypted connections against known application patterns; identify anomalies; and identify or block those without payload inspection. These analytics may also observe that malicious applications may deliberately manipulate certain TLS header fields, throttle packet rates, and vary payload sizes in order to circumvent detection.

Through traffic analysis, researchers have detected devastating pseudo-random number generator failures [TLS_VULNERABILITY], nonce failures [NONCE_FAIL], and deeply flawed random number generators in products in [WEAK_KEY] and [WEAK_K2].

TLS 1.3 considerations: reduced effectiveness. Per Section 4.2, any features relying on Certificate information will not be available.

5.1.3. **OP-3.** Crypto, Security and Security Policy Compliance (server)

Deployment: out-of-band

A network security device observes TLS handshake traffic to audit that TLS server configuration conforms to policy. This compliance monitoring commonly examines ciphersuites (e.g., use of weak ciphersuites) and certificate properties (e.g., no self-signed certificates, black or white list of certificate authorities, certificate expiration times).

TLS 1.3 considerations: reduced effectiveness. Per Section 4.2, only TLS ClientHello and ServerHello parameters can be audited. Certification information will not be visible.

5.1.4. **OP-4.** Crypto and Security Policy Compliance (client)

Deployment: inline

A network security device observes TLS handshake traffic to ensure that clients negotiating TLS connections have configurations (e.g., only make connections with TLS 1.2+) and server certificate (e.g., black-listed CAs) that adhere to policy. This is a variant of OP-3. It is commonly used in deployments where an organization may have reduced configuration control of end points (e.g., lab environments, Bring Your Own Device arrangements, and IoT).

TLS 1.3 considerations: reduced effectiveness. Per Section 4.2, only TLS ClientHello and ServerHello parameters can be audited. Certification information will not be visible.

5.2. Outbound TLS Proxy

Outbound TLS proxy is the deployment scenario where a security device that performs the TLS proxy function is in the same administrative domain as the TLS client, and the TLS server is located in an external zone such as the Internet or in another policy zone of the same administrative domain. Usually the goal is to protect the client endpoint and the organization by controlling application behaviors and enforcing an acceptable use policy for the organizational network. See Figure 1.

The administrator manages the TLS client to allow interception by the TLS proxy, usually by deploying a local Certificate Authority (CA) certificate on the TLS client. A typical scenario is an organization-managed client endpoint, such as a laptop or a mobile device that accesses the Internet through the organizational network. When a client attempts to access an external TLS server, the TLS proxy function typically presents a locally signed certificate from the local CA on behalf of the server; alternatively, the certificate generation function may be offloaded to an external Hardware Security Module (HSM) service with which that the TLS proxy must integrate.

It has to be noted that the method does not work if the TLS client does not support customized list of CAs, such as with certificate pinning. The impact is independent of TLS 1.3 deployment.

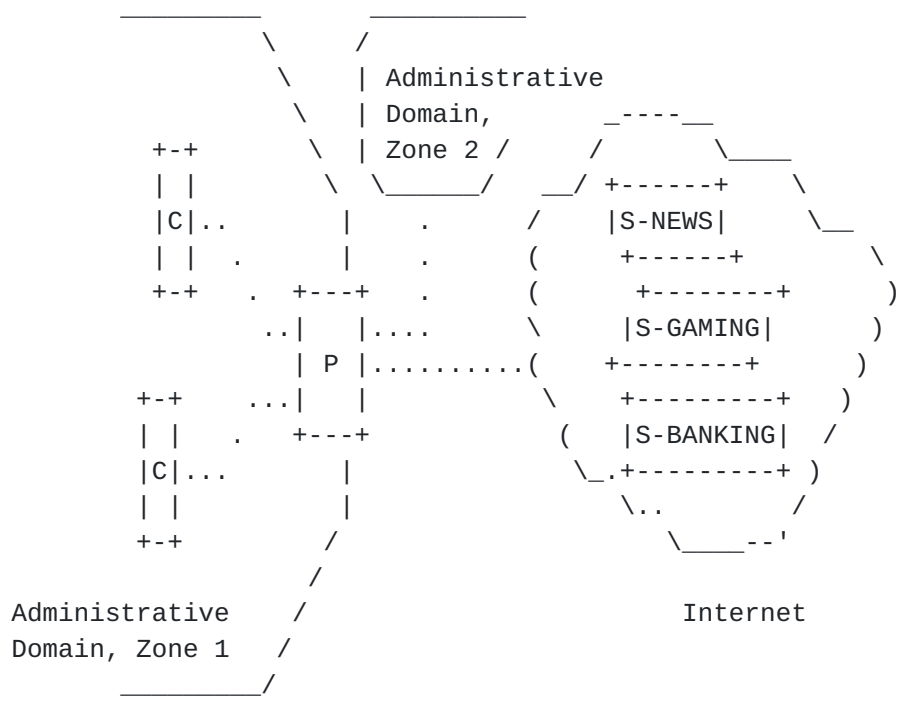

Figure 1: Outbound TLS proxy

5.2.1. OP-5: Acceptable Use Policy (AUP) Enforcement (via payload inspection)

Deployment: inline

A firewall or web proxy restricts a client in the same administrative domain from accessing sites or services outside that domain per an acceptable use policy. Similar in intent to OP-1, but the policy enforcement in this practice requires access to data in the TLS session (e.g., URL).

TLS 1.3 considerations: no capability impact. See Section 4.2 if a selective decryption policy is used.

5.2.2. OP-6: Data Loss Prevention Compliance

Deployment: inline

A firewall enforces a Data Loss Prevention (DLP) policy by monitoring the TLS sessions content of outbound communication for systems sending organizational proprietary content or other restricted information. Note that the firewall may be implemented and enforced either at the endpoint or by the network infrastructure.

TLS 1.3 considerations: no capability impact. See Section 4.2 if a selective decryption policy is used.

5.2.3. OP-7: Granular Network Segmentation

Deployment: inline

A firewall mediates the traffic between different policy zones in an organization. The access policies between these zones may be based on application names and categories rather than static IP addresses and TCP/UDP port numbers. Through a TLS proxy, the firewall can inspect URLs and other application parameters based on data in the TLS session.

TLS 1.3 considerations: no capability impact. See Section 4.2 if a selective decryption policy is used.

5.2.4. OP-8: Network-based Threat Protection (client)

Deployment: inline or out-of-band (depending on functionality)

Web proxies and firewalls protect end-users against a range of threats by inspecting the data in the TLS session with a variety of analytical techniques (e.g., signatures, heuristics, statistical

models, machine learning). This practice is a superset of OP-2. Common goals are to prevent malware from reaching the endpoint, preventing malware communication from a compromised host, restricting lateral network movement of an intruder and gathering insight into the behavior of threat activity on the network.

In certain deployments these technologies are also used to act as a last line of defense against software vulnerabilities on endpoints - either for 0-days for which there is no patch, or simply unpatched clients.

TLS 1.3 considerations: no capability impact. See Section 4.2 if a selective decryption policy is used.

5.2.5. OP-9: Protecting Challenging End Points

Deployment mode: inline

Web proxies, IPS and firewalls implement security policy and afford protection to devices for which it is not feasible to run an end-point solution (e.g., IoT); or that are end-of-life and will not receive patches. This is a specialized instance of OP-8 targeting these disadvantaged classes of devices.

These practices ensure that that older endpoints (and in some cases even new ones) are not permanently vulnerable to newly discovered vulnerabilities.

TLS 1.3 considerations: no capability impact. See Section 4.2 if a selective decryption policy is used.

5.2.6. OP-10: Content Injection

Deployment: inline

A firewall or web proxy restricts message manipulation or insertion, such as a block page or an interactive authentication portal redirect, into the encrypted flow for the client to see. This may be used in conjunction with OP-1, OP-5, and OP-7.

TLS 1.3 considerations: no capability impact. See Section 4.2 if a selective decryption policy is used.

5.3. Inbound TLS Proxy

Inbound TLS proxy is the deployment scenario where the TLS proxy is deployed in front of one or a set of servers or services. The network device that implements the TLS proxy function is located in

the same administrative domain as the server(s) or service(s) it is protecting. Usually it is not predictable or controllable as to which TLS client will initiate a connection. See Figure 2.

The TLS proxy is provisioned with the server's certificates and private keys so that it may either decrypt or terminate the TLS connection on behalf of the server. In some instances, the TLS proxy may periodically retrieve the private keys and associated certificates from an external secure distribution service, such as a HSM. Traffic between the TLS proxy and server may be encrypted or in the clear; the former configuration is typical of a perimeter firewall while the latter of a load-balancer.

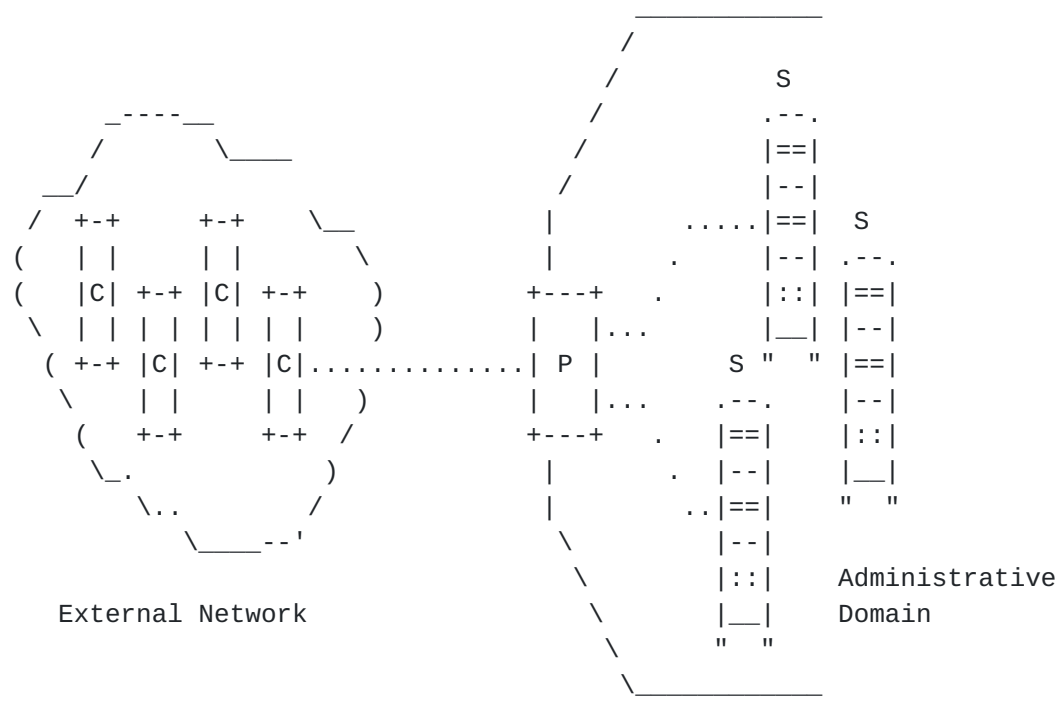

Figure 2: Inbound TLS proxy

5.3.1. OP-11: TLS offloading

Deployment mode: inline

Offloads crypto operations from the application server to a TLS Proxy. This is not a typical security function on its own, but it facilitates security control insertion downstream. As this is in the same administrative domain, it is presumed that a TLS Proxy can be provisioned with the appropriate keys when the TLS Server is configured or managed.

TLS 1.3 considerations: no impact.

5.3.2. OP-12. Content distribution and application load balancing

Deployment mode: inline

Load balancers deployed in front of services provide resiliency against denial of service attacks. TLS proxy functionality provides access to the cleartext application layer data to enable service-tailored load balancing. Similar to OP-11, it is presumed that a TLS Proxy can be provisioned with the appropriate keys when the TLS Server is configured or managed.

This practice may be combined with OP-11.

TLS 1.3 considerations: no impact.

5.3.3. OP-13: Network-based Threat Protection (server)

Deployment mode: inline and out-of-band

Web application firewalls (WAF) and firewalls protect servers and services against a range of threats by inspecting the data in the TLS session with a variety of analytical techniques (e.g., signatures, heuristics, statistical models, machine learning). This practice is identical in function to OP-8, but focused on threat prevention of inbound requests to servers and services.

TLS 1.3 considerations for inline deployment mode: no capability impact. Per Section 4.1, the network security device must explicitly terminate the TLS connection from the client.

TLS 1.3 considerations for out-of-band mode: alternative approach required. Per Section 4.1, active participation in the TLS exchange is required to inspect the session.

5.3.4. OP-14: Full Packet Capture

Deployment mode: inline and out-of-band

A network security device stores a copy of all decrypted traffic that meets a given filter. This traffic may be continuously captured in a rolling buffer for use in future forensic analysis, incident response, or computationally intensive retrospective analysis. This collection may also be selectively enabled to support application troubleshooting.

TLS 1.3 considerations for inline deployment mode: no capability impact. Per Section 4.1, the network security device must explicitly terminate the TLS connection from the client.

TLS 1.3 considerations for out-of-band mode: alternative approach required. Per Section 4.1, offline decryption is not possible.

5.3.5. OP-15: Application Layer Gateway (ALG)

Deployment mode: inline

To conduct protocol conformance checks and rewrite embedded IP addresses and TCP/UDP ports within the application layer payload for traffic traversing a NAT boundary. While not strictly a security function, this capability may typically be found in firewalls along with the NAT supporting functions.

TLS 1.3 considerations: no impact.

6. Security Considerations

This document presents common and existing security monitoring and detection functionality and how it interacts with TLS. It further notes where existing practices will have to be adjusted to remain effective as these solutions transition to include TLS 1.3 improvements.

These operational practices involve both good faith and malicious client applications. The former category typically exhibits consistently identifiable behavior and does not actively prevent any transit inspection devices from performing application identification for visibility and control purposes. The latter category of applications actively attempts to circumvent network security controls by deliberately manipulating various protocol headers, injecting specific messages, and varying payload sizes in order to avoid identification or to masquerade as a different permitted application.

7. IANA Considerations

This document has no IANA actions.

8. Appendix A: Summary Impact to Operational Practices with TLS 1.3

Operational Practice	Impact with TLS 1.3
OP-1: AUP enforcement (headers only)	reduced effectiveness
OP-2: Behavior analytics (headers only)	reduced effectiveness
OP-3: Crypto compliance monitoring (server)	reduced effectiveness
OP-4: Crypto compliance monitoring (client)	reduced effectiveness
OP-5: AUP enforcement (payload)	no capability impact
OP-6: Data loss prevention compliance	no capability impact
OP-7: Granular network segmentation	no capability impact
OP-8: Network protection (client)	no capability impact
OP-9: Protecting challenging end points	no capability impact
OP-10: Content Injection	no capability impact
OP-11: TLS offloading	no impact
OP-12: Application load balancing	no impact
OP-13: inline: Network protection (server)	no operational impact
OP-13: oob: Network protection (server)	alternative required
OP-14: inline: Full packet capture	no operational impact
OP-14: oob: Full packet capture	alternative required
OP-15: Application layer gateway	no impact

9. References

9.1. 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, <https://www.rfc-editor.org/info/rfc2119>.

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

[RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <https://www.rfc-editor.org/info/rfc7525>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8404] Moriarty, K., Ed. and A. Morton, Ed., "Effects of Pervasive Encryption on Operators", RFC 8404, DOI 10.17487/RFC8404, July 2018, <https://www.rfc-editor.org/info/rfc8404>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

9.2. Informative References

[ECH] Rescorla, E., Oku, K., Sullivan, N., and C. Wood, "TLS Encrypted Client Hello", draft-ietf-tls-esni-07 (work in progress), June 2020.

[NONCE_FAIL] Jovanovic, P., "Nonce-disrespecting adversaries: Practical forgery attacks on GCM in TLS", 2016, <https://www.usenix.org/conference/woot16/workshop-program/presentation/bock>.

[TLS_VULNERABILITY] Shenefiel, C., "PRNG Failures and TLS Vulnerabilities in the Wild", 2017, <https://rwc.iacr.org/2017/Slides/david.mcgrew.pptx>.

[WEAK_K2] Heninger, N., "Weak Keys Remain Widespread in Network Devices", 2016, <https://www.cis.upenn.edu/~nadiah/papers/weak-keys/weak-keys.pdf>.

[WEAK_KEY] Halderman, A., "Mining your Ps and Qs: Detection of widespread weak keys in network devices", 2012, <https://www.usenix.org/conference/usenixsecurity12/technical-sessions/presentation/heninger>.

Acknowledgments

The authors thank Andrew Ossipov, Flemming Andreasen, Kirsty Paine, David McGrew, and Eric Vyncke for their contributions and valuable feedback.

Authors' Addresses

Nancy Cam-Winget
Cisco Systems, Inc.
3550 Cisco Way
San Jose, CA 95134
USA

EMail: ncamwing@cisco.com

Eric Wang
Cisco Systems, Inc.
3550 Cisco Way
San Jose, CA 95134
USA

EMail: ejwang@cisco.com

Roman Danyliw
Software Engineering Institute

EMail: rdd@cert.org

Roelof DuToit
Broadcom

EMail: roelof.dutoit@broadcom.com