Eliminating Obsolete Transport Layer Security (TLS) Protocol Configurations

Executive summary

The National Security Agency (NSA) emphatically recommends replacing obsolete protocol configurations with ones that utilize strong encryption and authentication to protect all sensitive information. Over time, new attacks against Transport Layer Security (TLS) and the algorithms it uses have been discovered. Network connections employing obsolete protocols are at an elevated risk of exploitation by adversaries.

Sensitive and valuable data requires strong protections within electronic systems and transmissions. TLS and Secure Sockets Layer (SSL) were developed as protocols to create private, secure channels between a server and client using encryption and authentication. While the standards and most products have been updated, implementations often have not kept up.

The accompanying, full-length guidance helps network administrators and security analysts make a plan on how to weed out obsolete TLS configurations in the environment by detecting, prioritizing, remediating, and then blocking obsolete TLS versions, cipher suites, and finally key exchange methods. This will also help organizations prepare for cryptographic agility to always stay ahead of malicious actors' abilities and protect important information.

Using obsolete encryption provides a false sense of security because it may look as though sensitive data is protected, even though it really is not. The NSA previously released urgent guidance indicating obsolete and otherwise weak TLS protocol implementations were being observed, and threat intelligence stating that "nation-state and sufficiently resourced actors are able to exploit these weak communications." However, obsolete TLS configurations are still in use in U.S. Government systems. Obsolete configurations provide adversaries access to sensitive operational traffic using a variety of techniques, such as passive decryption and modification of traffic through man-in-the-middle attacks.

The full version of this guidance details recommended detection and remediation strategies, as there are many ways to detect obsolete TLS configurations and different organizations may need to modify their remediation approaches to minimize network impacts. Detecting systems that use or allow obsolete TLS configurations is the first step that will help prioritize remediation efforts and provide the knowledge needed for risk determinations for allowing, blocking, or remediating. Additional guidance for detecting obsolete TLS traffic, including network signatures, links to helpful tools, and sample configurations, is available at github.com/nsacyber/Mitigating-Obsolete-TLS.

All publicly accessible federal websites and services are required to use secure connections. Additionally, National Institute of Standards and Technology guidance and Committee on National Security Systems policy prohibit U.S. Government and National Security Systems (NSS), respectively, from using obsolete protocol configurations. This guidance helps NSS, Department of Defense (DoD), and Defense Industrial Base (DIB) cybersecurity leaders make informed decisions to enhance their cybersecurity posture. Since these risks affect all networks, all network owners and operators should consider taking these actions to reduce their risk exposure and make their systems harder targets for malicious threat actors.

Contact

Cybersecurity Inquiries: 410-854-4200, Cybersecurity_Requests@nsa.gov

Media Inquiries: 443-634-0721, MediaRelations@nsa.gov

U/OO/197443-20 | PP-20-1302 | JAN 2021 Ver 1.0

Introduction

Sensitive and valuable data requires strong protections within electronic systems and transmissions. Protected transmissions use a private, secure channel between a server and a client to communicate. Transport Layer Security (TLS) and Secure Sockets Layer (SSL)[2] were developed as protocols to create these protected channels using encryption and authentication. Over time, new attacks against TLS and the algorithms it uses have been discovered. The standards and most products have been updated, but implementations often have not kept up. Network connections employing obsolete protocols are at an elevated risk of exploitation by adversaries. As a result, all systems should avoid using obsolete configurations for TLS and SSL protocols. According to the Office of Management and Budget (OMB) memorandum M-15-13, "all publicly accessible federal websites and web services are required to only provide service through secure connections" [1].

The National Security Agency previously released urgent guidance (ORN U/OO/800922-17) indicating that obsolete and otherwise weak TLS protocol implementations were being observed, and threat intelligence stating that "nation-state and sufficiently resourced actors are able to exploit these weak communications." However, internal analysis indicates that obsolete TLS configurations are still in use in U.S. Government systems. Obsolete configurations provide adversaries access to sensitive operational traffic using a variety of techniques, such as passive decryption and modification of traffic through man-in-the-middle attacks [2].

Using obsolete encryption provides a false sense of security because it seems as though sensitive data is protected, even though it really is not. National Institute of Standards and Technology (NIST) special publication guidance, SP 800-52rev2 (2019), and Committee on National Security Systems (CNSS) policy, CNSSP 15 (2016), prohibit U.S. Government and National Security Systems from using obsolete protocols [3] [4]. National Security Systems are required to use the algorithms in the NSA-Approved Commercial National Security Algorithm (CNSA) Suite (see Annex B of CNSSP 15). All other systems are recommended to use CNSA Suite algorithms as well. Non-NSS U.S. Government systems are required to use the algorithms specified by NIST in SP 800-52rev2. NSA strongly recommends detecting and remediating obsolete protocols and, instead, utilizing strong encryption and authentication to protect all sensitive information.

This guidance document provides detection strategies that can aid network security analysts in identifying continued use of obsolete TLS protocol versions, cipher suites, and key exchanges. By doing this, administrators should be alerted to non-compliant deployments so they can expeditiously be updated or disabled. Most network security devices able to detect such traffic can also be configured to block the traffic to eliminate exposure. Additional guidance for detecting obsolete TLS traffic, including network signatures, links to helpful tools, and sample configurations, is available at github.com/nsacyber/Mitigating-Obsolete-TLS.

Audience

The primary audiences for this guidance are National Security System (NSS), Department of Defense (DoD), and Defense Industrial Base (DIB) cybersecurity leaders, system administrators, and network security analysts. By using the following guidance, government network owners can make informed decisions to enhance their cybersecurity posture. Since these risks affect all networks, all network owners and operators should consider taking these actions to reduce their risk exposure and make their systems harder targets for malicious threat actors.

[2] SSL is the predecessor to TLS. Future references to TLS within this document are meant to indicate all versions of TLS and its predecessors of SSL 2.0 and SSL 3.0

Obsolete TLS versions

Sensitive data requires robust protection. TLS provides confidentiality, integrity, and often authenticity protections to data while it is in transit over a network. This occurs by providing a secured channel between a server and client to communicate for a session. Over time, new versions of the TLS protocol are developed and some of the previous versions become obsolete for numerous technical reasons or vulnerabilities, and therefore should no longer be used to sufficiently protect data.

NSA recommends that only TLS 1.2 or TLS 1.3 be used[3]; and that SSL 2.0, SSL 3.0, TLS 1.0, and TLS 1.1 not be used [5]. See NIST SP 800-52 Revision 2 Appendix F for related requirements and guidance for non-NSS U.S. Government systems.

Obsolete cipher suites

Within TLS 1.2 and TLS 1.3, NSA further recommends that cryptographic parameters meet the algorithm requirements in CNSSP 15, referred to as Commercial National Security Algorithms. In TLS 1.2, the term "cipher suite(s)" refers to the negotiated and agreed upon set of cryptographic algorithms for the TLS transmission. A list of cipher suites are offered by the TLS client, and a negotiated cipher suite from that list is selected by the TLS server. Cipher suites in TLS 1.2 consist of an encryption algorithm[4], an authentication mechanism[5], a key exchange[6] algorithm and a key derivation[7] mechanism[8]. A cipher suite is identified as obsolete when one or more of the mechanisms is weak.

Especially weak encryption algorithms in TLS 1.2 are designated as NULL, RC2, RC4, DES, IDEA, and TDES/3DES; cipher suites using these algorithms should not be used[9]. TLS 1.3 removes these cipher suites, but implementations that support both TLS 1.3 and TLS 1.2 should be checked for obsolete cipher suites.

Obsolete key exchange mechanisms

Especially weak key exchange mechanisms indicated by the cipher suite include those designated as EXPORT or ANON; cipher suites using these key exchange mechanisms should not be used. Even if the cipher suite used in a TLS session is acceptable, a key exchange mechanism may use weak keys that allow exploitation. TLS key exchange methods include RSA key transport and DH or ECDH key establishment. DH and ECDH include static as well as ephemeral mechanisms. NSA recommends RSA key transport and ephemeral DH (DHE) or ECDH (ECDHE) mechanisms, with RSA or DHE key exchange using at least 3072-bit keys and ECDHE key exchanges using the ***secp384r1*** elliptic curve***.*** For RSA key transport and DH/DHE key exchange, keys less than 2048 bits should not be used, and ECDH/ECDHE using custom curves should not be used.[10]

Recommended TLS configurations

All TLS implementations should be up-to-date and configured to meet CNSS and NIST guidance. Detecting systems that negotiate obsolete TLS versions or cipher suites or use weak keys is a first step that will help prioritize remediations. Once detected, an organization's servers and clients negotiating obsolete TLS sessions should be reconfigured to meet the requirements of CNSSP 15. If additional interoperability support is needed, configurations should use non-deprecated options from NIST SP 800-52r2 as necessary. Additional options commonly used by non-government servers may also need to be supported by clients; the risk of using options not approved by NIST should be assessed by the system owner. CNSSP 15 requirements for TLS are more fully explained by IETF in the "Commercial National Security Algorithm (CNSA)

[3] This applies to all protocols using TLS, such as HTTPS and LDAPS. Datagram Transport Layer Security (DTLS) is similar to TLS standards – NSA recommends only DTLS version 1.2 or above be used; DTLS 1.0 is obsolete.
[4] Encryption Algorithm is defined by NIST and CNSS as a "set of mathematically expressed rules for rendering data unintelligible by executing a series of conversions controlled by a key."
[5] Authentication Mechanism is defined by NIST and CNSS as "hardware or software-based mechanisms that force users to prove their identity before accessing data on a device."
[6] Key Exchange is defined by NIST and CNSS as a "process of exchanging public keys (and other information) in order to establish secure communications."
[7] Key Derivation is defined by NIST as "a process that derives keying material from a key or a shared secret."
[8] In TLS 1.3, the cipher suite extension only specifies the encryption algorithm and key derivation function; the key exchange and signature components are specified in other extensions of the client and server hello message.
[9] Use of DES and IDEA are not allowed in TLS 1.2 according to IETF RFC 5469 (2009), but many implementations support cipher suites using these algorithms.
[10] Use of custom public key parameters in key exchange messages is deprecated per RFC 8422 Section 5.1.1.

Suite Profile for TLS and DTLS 1.2 and 1.3" [6]. Guidance in this document includes recommended versions and cipher suites, as well as guidance on other TLS parameters, and provides recommendations for extension usage. Most clients, especially web browsers, will use compliant configurations just by upgrading the software to the latest version. In some cases, additional configurations to disable offering obsolete encryption may be needed. Specific instructions for configuring a server to be compliant should be available from the vendor. Additional configuration information is provided at github.com/nsacyber/Mitigating-Obsolete-TLS.

TLS also depends on proper use of certificates. Use of weak, compromised, unauthorized, or revoked certificates can lead to man-in-the-middle attacks, even for properly configured TLS implementations[11]. Server certificates should be issued by a reputable certification authority, should use approved signature schemes, and should be managed to ensure they accurately represent the server and are replaced prior to expiration. For U.S. Government owned servers and for clients representing U.S. Government employees, applicable certificate policies further restrict certificate usage. The Certification Authority - Browser Forum defines best practices for commercial servers. Obsolete or unauthorized certificates should be revoked. CNSS guidance for TLS certificates is included in the CNSA Suite Profile for TLS and DTLS 1.2 and 1.3 section 5.4 [6]; NIST recommendations for TLS certificates are in NIST SP 800-52r2 section 3.2 [3].

Detection strategy

Using network monitoring systems, signatures can be used to detect obsolete TLS, such as those provided at github.com/nsacyber/Mitigating-Obsolete-TLS. There are a number of open source tools that can monitor traffic using these signatures, as well as commercial services that can perform active scans to detect compliance. Whether the focus is to detect weak traffic, or to identify misconfigured servers or clients, the detection strategy will be similar.

Because there are many ways that obsolete TLS configurations may be exhibited in traffic, the following detection strategy is recommended. Signatures can be simplified using this strategy:

- First, identify clients offering and servers negotiating obsolete TLS versions. If a client offers, or a server negotiates SSL 2.0, SSL 3.0, or an obsolete TLS version, no further traffic analysis is required and remediation strategies should be employed.

- Next, for sessions using TLS 1.2, analysts should identify and remediate devices using obsolete cipher suites. Identify clients only offering and servers negotiating obsolete TLS cipher suites and update their configurations to be compliant. Note for TLS 1.3, neither NIST nor CNSS identify cipher suites that must not be used – however, CNSA compliant configurations should be followed [6].

- Finally, for sessions using TLS 1.2 or TLS 1.3 and recommended cipher suites, analysts should identify and remediate devices using weak key exchange methods.

Alerts can identify the IP address of the organization's device responsible for the improper use. If the organization's device observed using obsolete TLS is a server, then it (or the web application firewall or other network device that accepts connections on its behalf) should be upgraded or configured to only negotiate recommended versions and cipher suites.It should also use recommended key exchange mechanisms, including recommended key sizes or elliptic curve groups. If the organization's device is a client, then it is offering an obsolete TLS version or cipher suite and may accept weak key exchange mechanisms. A client that is offering to use both recommended and obsolete cipher suites does not necessarily mean that the server would select the obsolete cipher suite to use (that would be detected by a server rule). Clients should be upgraded to use the most current browser or TLS library. It should be configured to only offer recommended versions and cipher suites, and not accept sessions using weak key exchanges negotiated by a server.

Comprehensive analysis of organizational servers can also be performed by attempting to initiate weak TLS sessions using custom test clients and seeing if the server agrees to utilize obsolete cryptography. This is useful for methodically determining compliance.

[11] Weak certificates are ones that have keys or other properties that do not meet or exceed current policies and best practices. Compromised certificates are ones that an unauthorized entity has access to the private keys. Unauthorized certificates are ones that should not have been issued according to the issuer's policies or procedures. Revoked certificates are ones that the issuer states are no longer to be trusted.

Remediation

Network monitoring devices can be configured to alert analysts to servers and/or clients that negotiate obsolete TLS or can be used to block weak TLS traffic. The choice to alert and/or block will depend on the organization. To minimize mission impact, organizations should use a phased approach to detecting and fixing clients and servers until an acceptable number have been remediated before implementing blocking rules. The following tables can help to prioritize responses for each topic discussed in this document. In addition to addressing obsolete TLS configurations, organizations should plan to update all other servers and/or clients to support CNSSP-15 recommended algorithms.

The following table indicates the prioritization and urgency for immediate remediation of obsolete TLS versions.

Version	Additional Monitoring	Traffic Response	Asset Response
SSL 2.0	N/A	Immediate block	Disable service until reconfigured to only support TLS 1.2 and TLS 1.3.
SSL 3.0	N/A	Immediate block	Disable service until reconfigured to only support TLS 1.2 and TLS 1.3.
TLS 1.0	N/A	Immediate block	Immediately reconfigure to only support TLS 1.2 and TLS 1.3.
TLS 1.1	If not blocked, detect obsolete cipher suites	Blocking recommended	Reconfigure to only support TLS 1.2 and TLS 1.3 as soon as possible.

Table 1: Prioritization of remediation of obsolete TLS versions

For detecting obsolete cipher suites based on a key phrase listed in the *EncryptionAlgorithm*[12] of the name, the recommended priority order is laid out in the following table.

Cipher Suite	Additional Monitoring	Traffic Response	Asset Response
NULL	N/A	Immediate block	Disable or quarantine until reconfigured.
RC2	N/A	Immediate block	Disable or quarantine until reconfigured.
RC4	N/A	Immediate block	Disable or quarantine until reconfigured.
DES	N/A	Immediate block	Disable or quarantine until reconfigured.
IDEA	N/A	Immediate block	Disable or quarantine until reconfigured.
TDES/3DES	N/A	Immediate block	Disable or quarantine until reconfigured.

Table 2: Prioritization of remediation of obsolete TLS cipher suites

For each cipher suite not blocked due to an obsolete encryption algorithm, determine if the cipher suite uses obsolete key exchange methods based on the table below.

Key Exchange Method	Additional Monitoring	Traffic Response	Server Response	Client Response
ANON	N/A	Immediate block	Disable or quarantine until reconfigured.	Disable or quarantine until reconfigured.
EXPORT	N/A	Immediate block	Disable or quarantine until reconfigured.	Disable or quarantine until reconfigured.
RSA with keys < 1024 bits (common sizes 512, 768)	N/A	Immediate block	Disable or quarantine until reconfigured. Install enterprise approved certificate.	Review configuration to ensure it is up to date. Reconfigure as necessary.
DHE with keys < 1024 bits (common sizes 512, 768)	N/A	Immediate block	Disable or quarantine until reconfigured.	Review configuration to ensure it is up to date. Reconfigure as necessary.
ECDHE with custom curves	N/A	Immediate block	Disable or quarantine until reconfigured.	Reconfigure to only offer recommended curves.
RSA with keys between 1024 and 2048 bits (common sizes 1024, 1536)	N/A	Detect/Block[13]	Reconfigure or update to support 3072 bit RSA, DHE with 3072 bits, and/or ECDHE with p384. Install enterprise	Reconfigure or update to support 3072 bit RSA, DHE with 3072 bits, and/or ECDHE with p384.

[12] Cipher Suite naming convention for TLS 1.2 is TLS_*KeyExchangeAlgorithm*_WITH_*EncyrptionAlgorithm_MessageAuthenticationAlgorithm*.
[13] Blocking is recommended, but if doing so will significantly impact mission, it may be necessary to coordinate a schedule to initiate blocking after mission critical servers are updated.

Key Exchange Method	Additional Monitoring	Traffic Response	Server Response	Client Response
			approved certificate to support RSA[14].	
DH/DHE with keys between 1024 and 2048 bits (common sizes 1024, 1536)	N/A	Detect/Block[13]	Reconfigure or update to support DHE with 3072 bits, and/or ECDHE with p384[14].	Reconfigure or update to support 3072 bit RSA, DHE with 3072 bits, and/or ECDHE with p384.

Table 3: Prioritization for Remediation of Key Exchange Methods

Configure deficient or out-of-date devices to meet the recommendations in [6].

Obsolete TLS provides a false sense of security

Organizations encrypt network traffic to protect data in transit. However, using obsolete TLS configurations provides a false sense of security since it looks like the data is protected, even though it really is not. Make a plan to weed out obsolete TLS configurations in the environment by detecting, remediating, and then blocking obsolete TLS versions, cipher suites, and finally key exchange methods. Prepare for cryptographic agility to always stay ahead of malicious actors' abilities and protect important information.

Works cited

[1] Executive Office of the President [Barack Obama]. "M-15-13: Policy to Require Secure Connections across Federal Websites and Web Services." White House, Jun. 2015. whitehouse.gov/sites/whitehouse.gov/files/omb/memoranda/2015/m-15-13.pdf.

[2] National Security Agency. "U/OO/800922-17: Cybersecurity Operational Risk Notice: Network Security Devices Utilizing Vulnerable Weak Signature Algorithms in TLS." NSA, 2017. www.nsa.gov/Portals/70/documents/what-we-do/cybersecurity/professional-resources/orn-deprecated-signature-algorithms.pdf.

[3] National Institute for Standards and Technology. "NIST SP 800-52 Rev. 2: Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations." NIST, 2019. nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-52R2.pdf.

[4] United States, Committee on National Security Systems. "CNSSP 15: Use of Public Standards for Secure Information Sharing." CNSS, 2016. www.cnss.gov/CNSS/issuances/Policies.cfm.

[5] Moriarty K., Farrell S. "Deprecating TLSv1.0 and TLSv1.1", IETF, Nov 2020, tools.ietf.org/pdf/draft-ietf-tls-oldversions-deprecate-09.pdf.

[6] Cooley, D. "Commercial National Security Algorithm (CNSA) Suite Profile for TLS and DTLS 1.2 and 1.3." IETF and NSA. Sep. 2020. tools.ietf.org/pdf/draft-cooley-cnsa-dtls-tls-profile-06.pdf.

Disclaimer of endorsement

Purpose

This Information Sheet is being issued to provide guidance on detecting and remediating obsolete TLS versions and cipher suites. The National Security Agency is responsible for developing, reviewing, and approving all standards, techniques, systems, and equipment related to the security of National Security Systems (NSS). NSS are subject to the minimum standards established in Committee on National Security Systems (CNSS) Policy 15 (2016), while the National Institute of Standards and Technology (NIST) issues guidance in special publications for all other U.S. Government information systems. Unless otherwise noted, protocol versions and cipher suites that have been identified as authorized were approved for use by NIST on non-NSS U.S. Government information systems in SP 800-52rev2 (2019). Nothing herein should be construed to alter or supersede guidance issued by CNSS for NSS or NIST for non-NSS systems.

This document was developed in furtherance of NSA's cybersecurity missions, including its responsibilities to identify and disseminate threats to National Security Systems, Department of Defense, and Defense Industrial Base information systems, and to develop and issue cybersecurity specifications and mitigations. This information may be shared broadly to reach all appropriate stakeholders.

Contact

Client Requirements / General Cybersecurity Inquiries: Cybersecurity Requirements Center, 410-854-4200, Cybersecurity_Requests@nsa.gov
Media Inquiries / Press Desk: Media Relations, 443-634-0721, MediaRelations@nsa.gov

[14] CNSSP 15 allows deployments using commercial technology solely for the protection of UNCLASSIFIED NSS data or for community of interest separation to continue to use RSA and Diffie-Hellman at the 2048 bit level and SHA-256 in the near term.