Communications Security Establishment
Centre de la sécurité des télécommunications

CANADIAN CENTRE FOR CYBER SECURITY

Guidance on Securely Configuring Network Protocols

(Version 2)

PRACTITIONER

FOREWORD

This document, *ITSP.40.062 Guidance on Securely Configuring Network Protocols*, is an UNCLASSIFIED publication issued by the Canadian Centre for Cyber Security (Cyber Centre) and provides an update to the previously published version.

We recommend that you also read *ITSP.40.111 Cryptographic Algorithms for UNCLASSIFIED, PROTECTED A, and PROTECTED B Information*. The configurations in this document comply with the cryptographic requirements in ITSP.40.111 [1].

EFFECTIVE DATE

This publication takes effect on September 21, 2020.

REVISION HISTORY

Revision	Amendments	Date
1	First release.	August 2, 2016
2	Updated version (version 2).	October 13, 2020

OVERVIEW

This document identifies and describes acceptable security protocols, and their appropriate methods of use, that organizations can implement to protect sensitive information. For GC departments and agencies, the guidance in this document applies to UNCLASSIFIED, PROTECTED A, and PROTECTED B information.

An organization's ability to securely transmit sensitive data and information is fundamental to the delivery of its programs and services. Using cryptographic security protocols ensures the confidentiality, integrity, and availability of information and helps provide protection against certain cyber intrusion threats.

Data confidentiality, integrity, availability, stakeholder authentication and accountability, and non-repudiation are all benefits of properly configured security protocols. Various protocols may be required to satisfy your organization's specific security requirements, and each protocol should be selected and implemented to ensure all requirements are met.

For more information on securely configuring network protocols, contact us:

Canadian Centre for Cyber Security Contact Centre
contact@cyber.gc.ca
(613) 949-7048 or 1-833-CYBER-88

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

Organizations rely on information technology (IT) systems to achieve business objectives. These interconnected systems can be the targets of serious threats and cyber attacks that jeopardize the availability, the confidentiality, and the integrity of information assets. Compromised networks, systems, or information can have adverse effects on business activities and may result in data breaches and financial loss.

This document provides guidance on the following topics:

- Securely configuring network protocols to protect sensitive information[1];
- Approved algorithms that the Cyber Centre recommends for use with these network protocols; and
- Standards and National Institute of Standards and Technology (NIST) special publications that provide additional information on these network protocols.

This document aids technology practitioners in choosing and using appropriate security protocols for protecting sensitive information (UNCLASSIFIED, PROTECTED A, and PROTECTED B information) and complements the Treasury Board of Canada Secretariat (TBS) *Guideline on Defining Authentication Requirements* [2]. This document also provides cryptographic guidance for IT solutions at the UNCLASSIFIED, PROTECTED A, and PROTECTED B levels.[2] Organizations are responsible for determining their security objectives and requirements as part of their risk management framework.

1.1 IT SECURITY RISK MANAGEMENT PROCESS

When implementing security protocols, practitioners should consider the IT security risk management activities described in *ITSG-33 IT Security Risk Management: A Lifecycle Approach* [3]. ITSG-33 addresses two levels of IT security risk management activities (departmental-level activities and information system-level activities) and includes a catalogue of security controls (i.e. standardized security requirements to protect the confidentiality, integrity, and availability of IT assets). See Figure 1 for an overview of the IT security risk management activity levels.

Additionally, organizations should consider the following activity areas: Define, Deploy, Monitor and Assess. See Annex 1 of ITSG-33 [3] for more information on these activities.

Departmental-level activities (or organizational-level activities for non-GC organizations) are included in departmental or organizational security programs to plan, manage, assess, and improve the management of IT security risks.

[1] For a GC department or agency, this guidance can be applied to UNCLASSIFIED, PROTECTED A, and PROTECTED B systems and information. Systems operating in PROTECTED C or classified domains may require additional design considerations that are not within the scope of this document.

[2] Systems operating in PROTECTED C or classified domains may require additional design considerations that are not within the scope of this document.

Information system-level activities are included in an information system's lifecycle through the information system security implementation process (ISSIP). When implementing network security protocols, you should consider all the steps in the ISSIP. See Annex 2 of ITSG-33 [3] for more information.

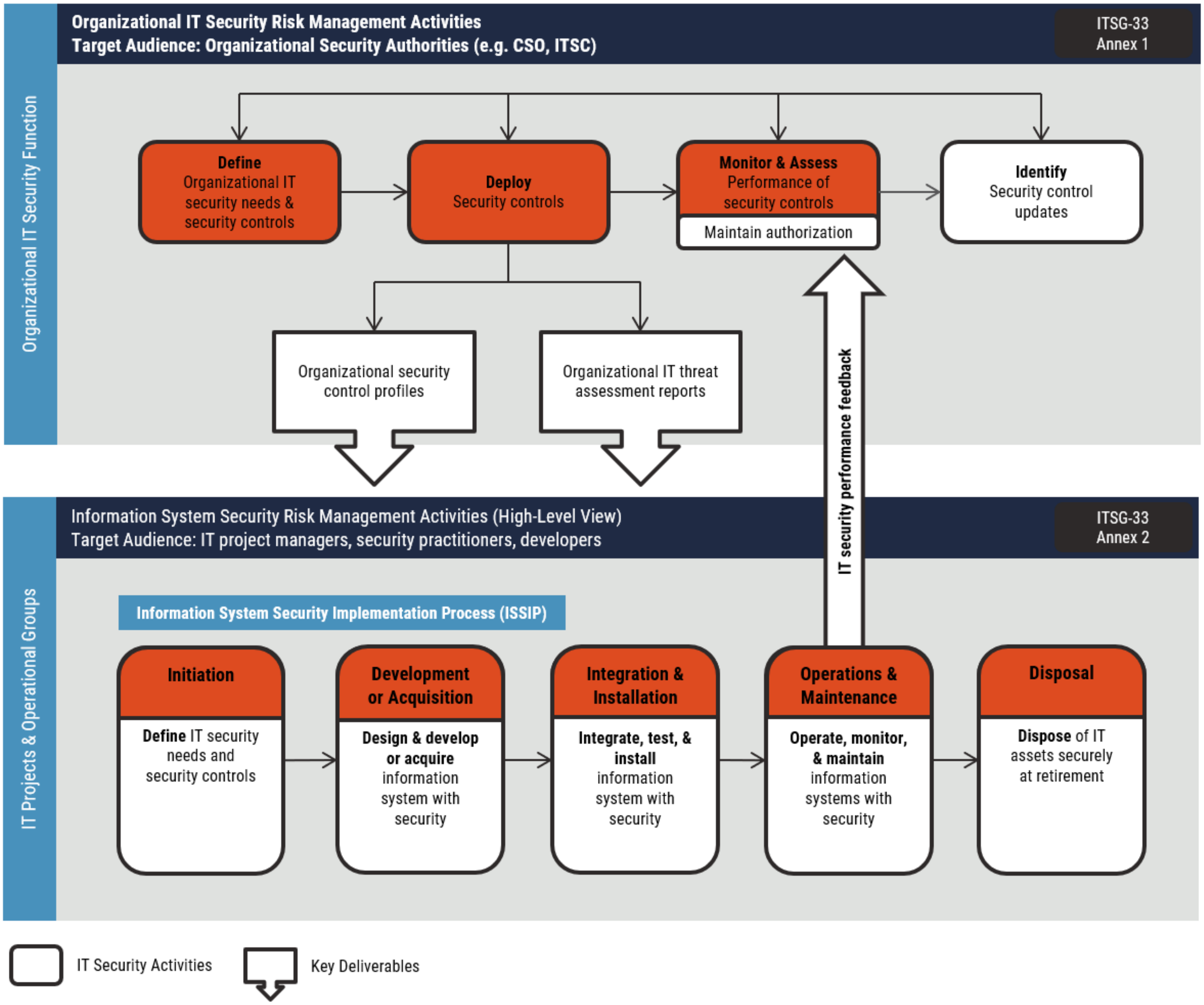
Organizational IT Security Function
Organizational IT Security Risk Management Activities
Target Audience: Organizational Security Authorities (e.g. CSO, ITSC)
ITSG-33 Annex 1
Define Organizational IT security needs & security controls
Deploy Security controls
Monitor & Assess Performance of security controls
Maintain authorization
Identify Security control updates
Organizational security control profiles
Organizational IT threat assessment reports
IT security performance feedback
IT Projects & Operational Groups
Information System Security Risk Management Activities (High-Level View)
Target Audience: IT project managers, security practitioners, developers
ITSG-33 Annex 2
Information System Security Implementation Process (ISSIP)
Initiation
Define IT security needs and security controls
Development or Acquisition
Design & develop or acquire information system with security
Integration & Installation
Integrate, test, & install information system with security
Operations & Maintenance
Operate, monitor, & maintain information systems with security
Disposal
Dispose of IT assets securely at retirement
IT Security Activities
Key Deliverables

Figure 1: IT Security Risk Management Process

1.2 RECOMMENDATIONS

Throughout this document, we make recommendations that fall within three categories: *Recommended*, *Sufficient*, and *Phase Out*. These three categories are explained further in Table 1.

Table 1: Three Recommendation Categories Used in this Document

Recommended	Sufficient	Phase Out
Configurations listed in the *Recommended* column have advantages over those in the *Sufficient* column. *Recommended* configurations should always be implemented if allowed by the remote connection profile.	Configurations listed in the *Sufficient* column are appropriate to be used as deemed necessary to support the profile of remote connections. *Sufficient* configurations should be applied when it is not possible to implement a *Recommended* profile.	Configurations listed in the *Phase Out* column are marked for transition according to guidance in *ITSP.40.111* [1] or due to protocol-specific concerns. If you have systems that use *Phase Out* selections, we recommend that you transition to *Recommended* or *Sufficient* alternatives as soon as possible.

Note: Systems do not need to be configured with all the selections listed in the *Recommended* or *Sufficient* columns. The chosen configurations will depend on an organization's remote connection profile. The protocol selections should be implemented to limit the network attack surface.

2 PUBLIC KEY INFRASTRUCTURE

Public Key Infrastructures (PKIs) support the management of public keys for security services in PKI-enabled protocols, including Transport Layer Security (TLS), Internet Protocol Security (IPsec), and Secure/Multipurpose Internet Mail Extensions (S/MIME).

PKI key management guidance is provided in *NIST Special Publication (SP) 800-57 Part 3 Rev 1 Recommendation for Key Management Part 3: Application-Specific Key Management Guidance* [4]. We recommend that you refer to section 2 of *NIST SP 800-57 Part 3 Rev 1* [4] for the guidance on installing and administering PKI.

Your implementations must not reuse public key pairs across multiple protocols within the PKI. For example, key pairs used in IKEv2 must not be reused for SSH.

You should format public key certificates in the X.509 version 3 certificate format, as specified in *Request for Comments (RFC) 5280 Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile* [5].

To support algorithm and key size agility, protocol implementations should support multiple certificates with their associated private keys. Public key certificates used for signing, key agreement, or key encipherment should be distinguished by the key usage extension, asserting one of the following bit-valued flags:

- digitalSignature;
- keyEncipherment; and
- keyAgreement.

To satisfy the cryptographic guidance provided in ITSP.40.111 [1], SHA-1 should not be used to generate or verify public key certificate digital signatures.

3 TRANSPORT LAYER SECURITY

TLS is a protocol developed to protect the confidentiality, integrity, and availability of Internet communications between server and client applications.

As specified in *RFC 8446 The Transport Layer Security (TLS) Protocol Version 1.3* [6], we *recommend* configuring TLS servers and clients to use TLS 1.3. Using TLS version 1.2, updated in RFC 8446 [6], is *sufficient* if it is required for wider compatibility, internal audit compliance, or threat monitoring systems. You should phase out versions of TLS older than 1.2 or any versions of Secure Sockets Layer (SSL).

Servers that use TLS to protect HTTP traffic (i.e. HTTPS) should support HTTP Strict Transport Security (HSTS), as specified in *RFC 6797 HTTP Strict Transport Security (HSTS)* [7].

An email server acting as a Message Transfer Agent (MTA) for Simple Mail Transfer Protocol (SMTP) should support the negotiation of TLS with other MTAs. SMTP traffic can be upgraded to TLS using STARTTLS, as specified in RFC 3207 SMTP Service Extension for Secure SMTP over Transport Layer Security [8]. To ensure the use of TLS for SMTP traffic, MTAs should support RFC 8461 SMTP MTA Strict Transport Security (MTA-STS) and configured to use the "enforce" policy mode [35] or RFC 7672 SMTP Security via Opportunistic DNS Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) [9].

Note: These opportunistic encryption techniques are only supported on a hop-by-hop basis. End-to-end message protection is provided by S/MIME (see section 6 of this document).

When TLS is used to protect the confidentiality of PROTECTED A or PROTECTED B information or the integrity of UNCLASSIFIED, PROTECTED A, or PROTECTED B information, you should use X.509 version 3 certificates to mutually authenticate between the server and the client.

3.1 TLS CIPHER SUITES

If the server or the client is configured to support TLS version 1.3, then the server or the client should be configured to support only the cipher suites listed in Table 2.

Table 2: Recommended Cipher Suites for TLS Version 1.3

Recommended	Sufficient
TLS_AES_256_GCM_SHA384 TLS_AES_128_GCM_SHA256 TLS_AES_128_CCM_SHA256	TLS_AES_128_CCM_8_SHA256

If TLS 1.2 support is required, a TLS server or client should be configured to support only the TLS 1.2 cipher suites listed in Table 3.

Table 3: Recommended Cipher Suites for TLS 1.2

Recommended	Sufficient	Phase Out
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	TLS_DHE_RSA_WITH_AES_256_GCM_SHA384	TLS_RSA_WITH_AES_256_GCM_SHA384
TLS_ECDHE_ECDSA_WITH_AES_256_CCM	TLS_DHE_DSS_WITH_AES_256_GCM_SHA384	TLS_RSA_WITH_AES_128_GCM_SHA256
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	TLS_DHE_RSA_WITH_AES_256_CCM	TLS_RSA_WITH_AES_256_CBC_SHA256
TLS_ECDHE_ECDSA_WITH_AES_128_CCM	TLS_DHE_RSA_WITH_AES_128_GCM_SHA256	TLS_RSA_WITH_AES_256_CBC_SHA
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	TLS_DHE_DSS_WITH_AES_128_GCM_SHA256	TLS_RSA_WITH_AES_128_CBC_SHA256
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	TLS_DHE_RSA_WITH_AES_128_CCM	TLS_RSA_WITH_AES_128_CBC_SHA
	TLS_ECDHE_ECDSA_WITH_AES_256_CCM_8	TLS_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA
	TLS_ECDHE_ECDSA_WITH_AES_128_CCM_8	TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA
	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA
	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA
	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA
	TLS_DHE_RSA_WITH_AES_256_CBC_SHA256	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
	TLS_DHE_RSA_WITH_AES_128_CBC_SHA256	TLS_DHE_RSA_WITH_AES_256_CBC_SHA
		TLS_DHE_RSA_WITH_AES_128_CBC_SHA

TLS servers and clients may use any or all the listed cipher suites according to the deployment profile. However, if an Internet-facing deployment requires cipher suites listed in the *Phase Out* column, we recommend you transition away from these as soon as possible. Your internal enterprise or data centre deployments of TLS may continue to use cipher suites with RSA key transport if required for audit compliance or threat monitoring systems.

Cipher suites do not specify a key size for the public key algorithm. TLS servers and clients should ensure that the server and client ephemeral key pairs that are used to establish the master secret satisfy the key length requirements specified in ITSP.40.111 [1]. Table 4 lists the Supported Groups that conform to ITSP.40.111 [1].

Table 4: Supported Groups that Conform to ITSP.40.111

Recommended	Sufficient	Phase Out
secp256r1	ffdhe3072	secp224r1
secp384r1	ffdhe4096	sect233r1
secp521r1	ffdhe6144	sect233k1
	ffdhe8192	ffdhe2048
	sect283k1	
	sect283r1	
	sect409k1	
	sect409r1	
	sect571k1	
	sect571r1	

Table 5 lists the Signature Algorithms that comply with ITSP.40.111 [1].

Table 5: Signature Algorithms that Comply with ITSP.40.111

Recommended	Sufficient	Phase Out
ecdsa_secp256r1_sha256	rsa_pkcs1_sha256	ecdsa_secp224r1_sha224
ecdsa_secp384r1_sha384	rsa_pkcs1_sha384	rsa_pkcs1_sha224
ecdsa_secp521r1_sha512	rsa_pkcs1_sha512	dsa_sha224
rsa_pss_pss_sha256		dsa_sha256
rsa_pss_pss_sha384		dsa_sha384
rsa_pss_pss_sha512		dsa_sha512
rsa_pss_rsae_sha256		
rsa_pss_rsae_sha384		
rsa_pss_rsae_sha512		

3.2 TLS EXTENSIONS

We recommend that TLS servers and clients support the following extensions:

- Certificate Signature Algorithms;
- Certificate Status Request;
- Cookie (TLS 1.3 only);
- Encrypt-then-MAC (TLS 1.2 only);
- Extended Master Secret (TLS 1.2 only);
- Key Share (TLS 1.3 only);
- Multiple Certificate Status (TLS 1.2 only);
- Pre-Shared Key (TLS 1.3 only);
- Pre-Shared Key Exchange Modes (TLS 1.3 only);
- Renegotiation Indication (TLS 1.2 only);
- Server Name Indication;
- Signature Algorithms;
- Signed Certificate Timestamps List;
- Supported Groups;
- Supported EC Point Formats (TLS 1.2 only);
- Supported Versions (TLS 1.3 only); and
- Trusted CA Indication (TLS 1.2 only).

Note: Do not enable extensions in your configurations that are not listed above.

3.3 CLIENT AND SERVER AUTHENTICATION

The client must validate the server certificate according to RFCs 5280 [5] and 8446 [6]. The revocation status of the certificate must be checked using a certificate revocation list (CRL) or the Online Certificate Status Protocol (OCSP). The client must check that the certificate contains a value in the Subject Alternative Name extension or in the Subject Distinguished Name field that matches the DNS or IP address requested.

If the client included the certificate signature algorithms extension, the client should verify that the certificate signature algorithm matches one of the proposed values. Otherwise, the client should verify that the certificate signature algorithm matches one of the proposed values in the signature algorithms extension.

Finally, the client should verify the public key length in the certificate satisfies the key length requirements specified in ITSP.40.111 [1].

If client authentication (also referred to as mutual authentication) is configured, the server must validate the client certificate according to RFCs 5280 [5] and 8446 [6]. The server must verify that the certificate validation path chains to a certificate authority (CA) that is trusted by the server to validate access to the requested resource. The revocation status of the certificate must be checked using a CRL or the OCSP. The server should check that the certificate contains a value in the Subject Alternative Name extension or in the Subject Distinguished Name field that matches an authorized client.

Finally, the server should verify that the public key length in the certificate satisfies the key length requirements specified in ITSP.40.111 [1].

3.4 OTHER TLS CONFIGURATION GUIDELINES

TLS clients and servers must be configured to disable TLS compression, which is done by negotiating the null compression method.

Due to the complication of mitigating replay attacks, we recommend that configurations do not support the 0-RTT mode of TLS version 1.3.

TLS 1.2 renegotiation without the Renegotiation Indication extension (see *RFC 5746 Transport Layer Security [TLS] Renegotiation Indication Extension* [10]) must be disabled. Furthermore, we recommend that TLS servers are configured to not accept client-initiated renegotiation at all in favour of establishing a new TLS connection.

If support for session resumption is desired, we recommend that you use the session identifier method in TLS 1.2 or session resumption via pre-shared keys (PSKs) in TLS 1.3. You should use PSKs with a (EC)DHE key exchange to provide forward secrecy.

4 INTERNET PROTOCOL SECURITY

You can use a combination of the protocol pair Internet Key Exchange Protocol Version 2 (IKEv2) and IPsec to create secure data tunneling at the network layer. The IKEv2 protocol establishes secure key material that can be used in the IPsec protocol to secure the data that is exchanged between peers.

4.1 IKEv2

IKEv2 is specified in *RFC 7296 Internet Key Exchange Protocol Version 2 (IKEv2)* [11].

Note: IKEv1 should no longer be used.

4.1.1 AUTHENTICATION

When IKEv2 is used to set up an IPSec security association (SA) to protect the confidentiality of PROTECTED A or PROTECTED B information or the integrity of UNCLASSIFIED, PROTECTED A, or PROTECTED B information, digital signatures should be used for authentication. Pre-shared keys should not be used for authentication.

Table 6 lists the authentication schemes that comply with ITSP.40.111 [1].

Table 6: Recommended IKEv2 Authentication Schemes

Recommended	Sufficient	Phase Out
ECDSA with SHA-256 on the P-256 curve ECDSA with SHA-384 on the P-384 curve ECDSA with SHA-512 on the P-521 curve RSASSA-PSS with bit length 3072 and SHA-384 RSASSA-PSS with bit length 4096 and SHA-384	RSASSA-PKCS1-v1.5 with bit length 3072 and SHA-384 RSASSA-PKCS1-v1.5 with bit length 4096 and SHA-384	RSASSA-PSS with bit length 2048 and SHA-256 RSASSA-PKCS1-v1.5 with bit length 2048 and SHA-256

4.1.2 MESSAGE ENCRYPTION

Table 7 lists the IKEv2 message encryption algorithms that comply with ITSP.40.111 [1].

Table 7: Recommended IKEv2 Message Encryption Algorithms

Recommended	Sufficient	Phase Out
ENCR_AES_GCM_16 ENCR_AES_CCM_16	ENCR_AES_GCM_12 ENCR_AES_CCM_12 ENCR_AES_CBC ENCR_AES_CTR	ENCR_3DES ENCR_CAST

We recommend using Advanced Encryption Standard (AES) in Galois/Counter Mode (GCM) to encrypt IKEv2 messages. If GCM or CCM is not supported, use an integrity protection mechanism from subsection 4.1.6.

4.1.3 KEY EXCHANGE

Table 8 lists the IKEv2 key exchange groups that comply with ITSP.40.111 [1].

Table 8: Recommended IKEv2 Key Exchange Groups

Recommended	Sufficient	Phase Out
256-bit Random ECP Group 384-bit Random ECP Group 521-bit Random ECP Group	3072-bit MODP Group 4096-bit MODP Group 6144-bit MODP Group 8192-bit MODP Group	2048-bit MODP Group 2048-bit MODP Group with 224-bit Prime Order Subgroup 2048-bit MODP Group with 256-bit Prime Order Subgroup 224-bit Random ECP Group

Implementations must check that received public values are between 1 and p-1 and, in the case of Elliptic-Curve Diffie-Hellman (ECDH), satisfy the elliptic curve equation.

We recommend that every key exchange uses a freshly generated ephemeral ECDH/DH key pair.

4.1.4 PSEUDO-RANDOM FUNCTIONS FOR KEY GENERATION

IKEv2 uses a pseudo-random function (PRF) to generate key material. Table 9 lists PRFs that comply with ITSP.40.111 [1].

Table 9: Sufficient PRF for IKEv2 Key Generation

Sufficient
PRF_HMAC_SHA2_256
PRF_HMAC_SHA2_384
PRF_HMAC_SHA2_512
PRF_AES128_CMAC

4.1.5 IKEv2 INTEGRITY PROTECTION

When not using an authenticated encryption (AEAD) algorithm (such as GCM) for message encryption, an additional integrity protection mechanism is required. Table 10 lists the integrity protection mechanisms that comply with ITSP.40.111 [1].

Table 10: Sufficient and Phase Out Integrity Protection Mechanisms

Sufficient	Phase Out
AUTH_HMAC_SHA2_256_128 AUTH_HMAC_SHA2_384_192 AUTH_HMAC_SHA2_512_256 AUTH_AES_128_GMAC AUTH_AES_192_GMAC AUTH_AES_256_GMAC AUTH_AES_CMAC_96	AUTH_HMAC_SHA1_160

4.1.6 EXTENSIBLE AUTHENTICATION PROTOCOL

RFC 7396 JSON Merge Patch [13] specifies that Extensible Authentication Protocol (EAP) in IKEv2 can be used if it is used with the IKEv2 responder public key-based authentication. *RFC 5998 An Extension for EAP-Only Authentication in IKEv2* [14] lists the methods that can be used in IKEv2 to provide mutual authentication and that do not require responder public key-based authentication.

While many authentication methods are listed as safe EAP methods in RFC 5998 [14], we recommend that you use methods that support channel binding. We also recommend that you maintain the use of responder public key-based authentication.

4.1.7 DDOS PROTECTION

IKEv2 is prone to distributed denial-of-service attacks (DDoS). In a DDoS attack, a threat actor overwhelms a responder with a huge number of SA requests that are sent from spoofed IP addresses, creating half-open SAs.

To protect against DDoS attacks, you should configure IKEv2 so that either the lifetime of half-open SAs or the upper limit to the maximum number of half-open SAs are allowed for a given IP before you take protective measures. You should implement the protection mechanisms described in *RFC 8019 Protecting IKEv2 Implementations from DDoS Attacks* [15].

You should not use IP fragmentation, as it is prone to DDoS attacks. Instead, use IKEv2 fragmentation and configure the size of the IKEv2 fragments.

RFC 7383 Internet Key Exchange Protocol Version 2 (IKEv2) Message Fragmentation [16] recommends a maximum datagram size of 1280 bytes for IPv6 traffic and 576 bytes for IPv4 traffic.

4.1.8 KEY AND AUTHENTICATION LIFETIMES

In the context of IKEv2, re-keying creates new key material for the IKE SA or a CHILD SA via the CREATE_CHILD_SA exchange. Re-authentication requires a complete IKE exchange and creates a new IKE SA. In this case, the old SAs are deleted.

We recommend that you ensure that the re-key period or key lifetime of a CHILD SA (including the Encapsulated Security Payload [ESP] SA) does not exceed 8 hours. The re-authentication period or authentication lifetime of the IKE SA should not exceed 24 hours.

4.1.9 SESSION RESUMPTION

RFC 5723 Internet Key Exchange Protocol Version 2 (IKEv2) Session Resumption [17] offers a means for peers to reconnect a broken connection by using a previously established IKE SA.

If session resumption is used, the ticket-by-reference method is recommended, under the condition that the peers can be trusted to maintain the security of stored SA information. We also recommend that you limit the lifetime of a ticket to no more than the re-keying time.

4.2 IPsec

IPsec is a suite of network protocols developed to protect the confidentiality, integrity, and availability of Internet communications between network hosts, gateways, and devices. IPsec also provides access control, replay protection, and traffic analysis protection.

IPsec hosts, gateways, and devices should be configured as specified in *RFC 4301 Security Architecture for the Internet Protocol* [18], *RFC 4303 IP Encapsulating Security Payload (ESP)* [19,] and *RFC 7321 Cryptographic Algorithm Implementation Requirements and Usage Guidance for Encapsulating Security Payload (ESP) and Authentication Header (AH)* [20].

IPsec key management guidance is provided in *NIST SP 800-57 Part 3 Rev 1* [4]. Refer to section 3 of *NIST SP 800-57 Part 3 Rev 1* [4] for guidance on installing and administering IPsec.

4.2.1 KEY GENERATION

An IPsec SA specifies the key material used to encrypt and provide integrity protection for the traffic protected under a specific IPsec session. An IPsec SA must be established by a prior IKEv2 exchange as specified above.

4.2.2 DATA AND INTEGRITY PROTECTION

You should use digital signatures for authentication when IPsec is used to protect the confidentiality of PROTECTED A or PROTECTED B information or the integrity of UNCLASSIFIED, PROTECTED A, or PROTECTED B information. You should not use PSKs for authentication.

IPsec should use ESP protocol in tunnel mode to protect the confidentiality, integrity, and availability of the packets and packet headers. Do not use the Authentication Header (AH) protocol. AH protocol cannot protect confidentiality.

Table 11 lists the ESP packet encryption algorithms that comply with ITSP.40.111 [1].

Table 11: Recommended ESP Packet Encryption Algorithms

Recommended	Sufficient	Phase Out
ENCR_AES_GCM_16 ENCR_AES_CCM_16	ENCR_AES_GCM_12 ENCR_AES_CCM_12 ENCR_AES_CBC ENCR_AES_CTR	ENCR_3DES ENCR_CAST

We recommend that you use AES in GCM for the encryption of ESP packets, as described in RFC 4106 [18]. If GCM or CCM is not supported, an integrity protection mechanism must be configured. Table 12 lists the integrity protection mechanisms that comply with ITSP.40.111 [1].

Table 12: Sufficient and Phase Out Integrity Protection Mechanisms

Sufficient	Phase Out
AUTH_HMAC_SHA2_256_128 AUTH_HMAC_SHA2_384_192 AUTH_HMAC_SHA2_512_256 AUTH_AES_128_GMAC AUTH_AES_192_GMAC AUTH_AES_256_GMAC AUTH_AES_CMAC_96	AUTH_HMAC_SHA1_160

4.2.3 REPLAY PROTECTION

Replay protection for IPsec implementations should be used. If performance allows, use the recommended anti-replay window size of 128.

5 SECURE SHELL

Secure Shell (SSH) is a protocol developed to protect the confidentiality, integrity, and availability of remote access, file transfer, and point-to-point tunneling over the Internet.

SSH servers and clients should be configured to use SSH protocol version 2.0. SSH is a family of protocols that is specified in *RFC 4251 The Secure Shell (SSH) Protocol Architecture* [21], *RFC 4252 The Secure Shell (SSH) Authentication Protocol* [22], *RFC 4253 The Secure Shell (SSH) Transport Layer Protocol* [23], and *RFC 4254 The Secure Shell (SSH) Connection Protocol* [24].

SSH protocol version 1.0 has serious vulnerabilities. Administrators should verify that it is not running on their systems.

NIST SP 800-57 Part 3 Rev 1 [4] provides SSH key management guidance. Refer to section 10 of *NIST SP 800-57 Part 3 Rev 1* [4] for guidance on installing and administering SSH.

5.1 SSH AUTHENTICATION

SSH offers both server-only and server-client mutual authentication.

You should use server-client mutual authentication. In this case, the server is first authenticated via the Transport Layer Protocol, followed by client authentication via the SSH Authentication protocol.

Server authentication is performed with public key cryptography. Client authentication to the server can use various mechanisms. Client authentication that is based on public keys or Kerberos is preferred rather than the various forms of password authentication. You should not use SSH host-based authentication; it is vulnerable to IP address spoofing.

If using public key authentication, you should use public key certificates that are managed by a PKI framework for both server and client authentication.

A PKI framework provides digital signing of keys by a trusted source. The framework also provides key management functions, such as revocation CRLs, key lifetime controls, and key usage restrictions. *RFC 6187 x509.v3 certificates for Secure Shell Authentication* [25] specifies the use of x509.v3 certificates in SSH.

Since SSH keys are typically system-level keys, keys should be generated upon session initialization to ensure uniqueness across devices and virtual machine images.

5.2 SSH PORT FORWARDING

With SSH port forwarding, a host can access an insecure network service on a machine residing behind a server that acts as an SSH VPN gateway. Port forwarding should be disabled for interactive user accounts. For devices that require SSH tunneling, the traffic should be secured with a second tunnel (e.g. IPSec).

5.3 SSH ROOT ACCESS

You should disable remote root user account logins.

5.4 SSH PARAMETER SELECTION

This section details the cryptographic algorithms recommend for SSH that satisfy the cryptographic guidance of ITSP.40.111 [1] and align with *NIST SP 800-57 Part 3 Rev 1* [4]. We recommend that you refer to subsection 10.2.1 of *NIST SP 800-57 Part 3 Rev 1* for cryptographic guidance on the SSH Transport Layer Protocol.

5.4.1 ENCRYPTION ALGORITHM SELECTION

Do not use Cipher Block Chaining (CBC) mode in SSH. **CBC mode is vulnerable to plain text recovery attacks.** *RFC 4344 The Secure Shell (SSH) Transport Layer Encryption Modes* [26] recommends using Counter (CTR) mode in SSH in place of CBC mode. Even better, authenticated encryption with associated data (AEAD) algorithms (such as AES GCM) protect both authenticity and confidentiality. Therefore, when you use AEAD algorithms, you do not need to use a separate MAC algorithm.

Table 13 lists the SSH encryption algorithms that satisfy the cryptographic guidance provided in ITSP.40.111 [1].

Table 13: Recommended SSH Encryption Algorithms

Recommended	Sufficient	Phase Out
AEAD_AES_128_GCM AEAD_AES_256_GCM	aes128-ctr aes192-ctr aes256-ctr	cast128-ctr 3des-ctr

The AEAD GCM encryption algorithms are vulnerable to nonce reuse. You should ensure that the (key, nonce) pair is unique for each encrypted message.

5.4.2 MAC ALGORITHM SELECTION

In addition to the AEAD algorithms specified above, Table 14 lists the SSH MAC algorithms that satisfy the cryptographic guidance provided in ITSP.40.111 [1].

Table 14: Sufficient and Phase Out SSH MAC Algorithms

Sufficient	Phase Out
hmac-sha2-256 hmac-sha2-512	hmac-sha1

5.4.3 KEY EXCHANGE ALGORITHM

Table 15 lists the SSH key exchange algorithms that satisfy the cryptographic guidance provided in ITSP.40.111 [1].

Table 15: Recommended SSH Key Exchange Algorithms

Recommended	Sufficient	Phase Out
ecdh-sha2-nistp256 ecdh-sha2-nistp384 ecdh-sha2-nistp521 ecmqv-sha2 gss-nistp256-sha256-* gss-nistp384-sha384-* gss-nistp521-sha512-*	diffie-hellman-group15-sha512 diffie-hellman-group16-sha512 diffie-hellman-group17-sha512 diffie-hellman-group18-sha512 gss-group15-sha512-* gss-group16-sha512-* gss-group17-sha512-* gss-group18-sha512-*	rsa2048-sha256 diffie-hellman-group14-sha256 gss-group14-sha256-*

The SSH protocol allows the session keys to be renewed by either the client or the server. Re-keying schedules are based on a time limit or a data volume, as described in RFC 4344 [26].

To avoid MAC collisions, RFC 4344 [26] recommends re-keying after receiving 2^{32} packets when a 32-bit sequence number is used.

5.4.4 PUBLIC KEY ALGORITHM

SSH optionally allows for authentication using public keys. Table 16 lists the SSH public key algorithms that satisfy the cryptographic guidance provided in ITSP.40.111 [1].

Table 16: Recommended SSH Public Key Algorithms

Recommended	Sufficient	Phase Out
ecdsa-sha2-nistp256 ecdsa-sha2-nistp384 ecdsa-sha2-nistp521 x509v3-ecdsa-sha2-nistp256 x509v3-ecdsa-sha2-nistp384 x509v3-ecdsa-sha2-nistp521	rsa-sha2-256 rsa-sha2-512 x509v3-rsa2048-sha256	x509v3-ecdsa-sha2-nistp224

6 SECURE/MULTI-PURPOSE INTERNET MAIL EXTENSIONS

S/MIME is a standard developed to protect the confidentiality, integrity, and availability of electronic messages over the Internet.

S/MIME 4.0 as specified in *RFC 8551 Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 4.0 Message Specification* [27] and *RFC 8550 Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 4.0 Certificate Handling* [28] should be used. S/MIME 4.0 includes support for AES-GCM.

RFC 5753 Use of Elliptic Curve Cryptography (ECC) Algorithms in Cryptographic Message Syntax (CMS) [29] provides guidance on the use of elliptic curve cryptography (ECC) in Cryptographic Message Syntax (CMS) for generating digital signatures and exchanging keys to encrypt or authenticate messages.

Software vendors should implement multi-part isolation with security considerations for dealing with HTML and multi-part/mixed messages, as discussed in RFC 8551 [27]. Until such multi-part isolation is supported, S/MIME clients must be configured to disable the loading of remote content or only display messages in plain text.

6.1 DIGEST ALGORITHMS

Digest algorithms are used in S/MIME for digesting the body of a message or as part of a signature algorithm. Table 15 lists the digest algorithms that satisfy the cryptographic guidance provided in ITSP.40.111 [1].

Table 17: Sufficient and Phase Out Digest Algorithms

Sufficient	Phase Out
SHA-256 SHA-384 SHA-512 SHA3-256 SHA3-384 SHA3-512	SHA-224 SHA3-224

Using SHA-1 to generate digital signatures does not satisfy the cryptographic guidance provided in ITSP.40.111 [1]. For S/MIME 3.2 or earlier versions, SHA-1 should not be used as a digest algorithm to sign messages.

6.2 SIGNATURE ALGORITHMS

Signature algorithms should be used with a digest algorithm. Table 18 lists the signature algorithms, which are paired with a digest algorithm from Section 6.1, that satisfy the cryptographic guidance provided in ITSP.40.111 [1].

Table 18: Recommended Signature Algorithm and Digest Algorithm Pairs

Recommended	Sufficient	Phase Out
ECDSA with NIST P-256 curve ECDSA with NIST P-384 curve ECDSA with NIST P-521 curve RSASSA PSS with 3072-bit or larger modulus	RSASSA PKCS1v1.5 with 3072-bit or larger modulus DSA with 3072-bit or larger group	ECDSA with NIST P-224 curve RSASSA PSS with 2048-bit modulus RSASSA PKCS1v1.5 with 2048-bit modulus DSA with 2048-bit group

We recommend using RSASSA-PSS (instead of PKCS #1 v1.5) as the encoding mechanism for RSA digital signatures. This applies to both X.509 certificates, as specified in *RFC 5756 Updates for RSAES-OAEP and RSASSA-PSS Algorithm Parameters* [30], and signed-data content types, as specified in *RFC 4056 Use of the RSASSA-PSS Signature Algorithm in Cryptographic Message Syntax (CMS)* [31]. If signing with multiple signature algorithms, you should use the multipleSignatures CMS attribute as specified in *RFC 5752 Multiple Signatures in Cryptographic Message Syntax (CMS)* [32].

Implementations of RSASSA-PSS should protect against possible hash algorithm substitution attacks. Implementations should check that the hash algorithm used to compute the digest of the message content is the same as the hash algorithm used to compute the digest of signed attributes.

6.3 KEY ENCRYPTION ALGORITHMS

Most key encryption algorithms for S/MIME require a key wrap algorithm to be specified as a parameter. Acceptable key wrap algorithms are specified in subsection 6.3.1 of this document. Table 19 lists the key encryption algorithms that satisfy the cryptographic guidance provided in ITSP.40.111 [1].

Table 19: Recommended Key Encryption Algorithms

Recommended	Sufficient	Phase Out
dhSinglePass stdDH SHA256 KDF with the NIST P-256 curve dhSinglePass stdDH SHA384 KDF with the NIST P-384 curve dhSinglePass stdDH SHA512 KDF with the NIST P-521 curve	RSAES OAEP with a 3072-bit or larger modulus dhSinglePass cofactorDH SHA256 KDF with the NIST P-256 curve dhSinglePass cofactorDH SHA384 KDF with the NIST P-384 curve dhSinglePass cofactorDH SHA512 KDF with the NIST P-521 curve mqvSinglePass SHA256 KDF with the NIST P-256 curve mqvSinglePass SHA384 KDF with the NIST P-384 curve mqvSinglePass SHA512 KDF with the NIST P-521 curve	dhSinglePass stdDH SHA224 KDF with the NIST P-224 curve dhSinglePass cofactorDH SHA224 KDF with the NIST P-224 curve RSA KEM with a 2048-bit modulus or larger RSAES OAEP with a 2048-bit modulus RSAES PKCS1v1.5 with a 2048-bit or larger modulus

We recommend the use of standard Elliptic Curve Diffie-Hellman, as specified in *RFC 5753 Use of Elliptic Curve Cryptography (ECC) Algorithms in Cryptographic Message Syntax (CMS)* [33].

If you are using RSA encryption, RSAES-OAEP should be implemented, as specified in *RFC 3560 Use of the RSAES-OAEP Key Transport Algorithm in Cryptographic Message Syntax (CMS)* [34] and *RFC 5756 Updates for RSAES-OAEP and RSASSA-PSS Algorithm Parameters* [30], to meet the cryptographic guidance of ITSP.40.111 [1].

Careful checking or random filling mitigations should be implemented, as described in *RFC 3218 Preventing the Million Message Attack on Cryptographic Message Syntax* [36], if you have S/MIME implementations that allow the decryption of PKCS #1 v1.5 encoding.

6.3.1 KEY WRAP ALGORITHMS

Table 20 lists the key wrap algorithms that can be used with an appropriate key encryption algorithm to satisfy the cryptographic guidance provided in ITSP.40.111 [1].

Table 20: Recommended Key Wrap Algorithms

Recommended	Phase Out
AES-128 Wrap AES-192 Wrap AES-256 Wrap AES-128 Wrap Pad AES-192 Wrap Pad AES-256 Wrap Pad	3DES Wrap CAST5 CMS Key Wrap with a key length of 128 bits

6.4 CONTENT ENCRYPTION ALGORITHMS

The following encryption algorithms are appropriate for S/MIME content encryption and satisfy the cryptographic guidance provided in ITSP.40.111 [1].

Table 21: Content Encryption Algorithms

Recommended	Phase Out
AES-128 GCM AES-192 GCM AES-256 GCM	AES-128 CBC AES-192 CBC AES-256 CBC

7 COMMERCIAL TECHNOLOGIES ASSURANCE PROGRAMS

When implementing PKI, TLS, IPsec, SSH and S/MIME, the implementation assurance guidance in Section 11 of ITSP.40.111 [1] should be followed.

8 SUMMARY

Your organization can implement cryptographic security protocols to provide the security mechanisms to protect the confidentiality, integrity, and availability of information. As a first step, you should determine your organizational security requirements before determining which protocols to implement. Although your organization will have its own specific security requirements, various protocols can be used. You should select and implement each protocol in a manner that supports and meets these specific requirements.

8.1 CYBER CENTRE CONTACT INFORMATION

If you would like more information on securely configuring network protocols, contact us by phone or email:

Contact Centre
contact@cyber.gc.ca
(613) 949-7048 or 1-833-CYBER-88

9 SUPPORTING CONTENT

9.1 LIST OF ABBREVIATIONS

Term	Definition
AEAD	Authenticated Encryption with Associated Data
AES	Advanced Encryption Standard
AH	Authentication Header
CA	Certificate Authority
CBC	Cipher Block Chaining
CMS	Cryptographic Message Syntax
CMVP	Cryptographic Module Validation Program
CRL	Certificate Revocation List
DANE	DNS-Based Authentication of Named Entities
DDoS	Distributed Denial of Service
DH	Diffie-Hellman
DTLS	Datagram Transport Layer Security
ECC	Elliptic Curve Cryptography
ECDH	Elliptic-Curve Diffie-Hellman
ECDHE	Ephemeral Elliptic Curve Diffie-Hellman
ECDSA	Elliptic Curve Digital Signature Algorithm
ECP	Elliptic Curve Groups modulo a Prime
ESP	Encapsulating Security Payload
GC	Government of Canada
GCM	Galois/Counter Mode
HMAC	Keyed-Hash Message Authentication Code
HSTS	HTTP Strict Transport Security
IKE	Internet Key Exchange
IPsec	Internet Protocol Security
IT	Information Technology
MAC	Message Authentication Code
MTA	Message Transfer Agent
PFS	Perfect Forward Secrecy
PKI	Public Key Infrastructure

Term	Definition
PRF	Pseudo-Random Function
NIST	National Institute of Standards and Technology
RFC	Request for Comments
RSA	Rivest Shamir Adleman
SA	Security Association
SHA	Secure Hash Algorithm
SSH	Secure Shell
S/MIME	Secure/Multipurpose Internet Mail Extensions
SMTP	Simple Mail Transfer Protocol
SP	Special Publication
SSL	Secure Socket Layer
TBS	Treasury Board of Canada Secretariat
TLS	Transport Layer Security

9.2 GLOSSARY

Term	Definition
Authentication	A process or measure used to verify a user's identity.
Authenticity	The state of being genuine and being able to be verified and trusted; confidence in the validity of a transmission, a message, or message originator.
Availability	The ability for the right people to access the right information or systems when needed. Availability is applied to information assets, software, and hardware (infrastructure and its components).
Classified Information	A Government of Canada label for specific types of sensitive data that, if compromised, could cause harm to the national interest (e.g. national defence, relationships with other countries, economic interests).
Confidentiality	The ability to protect sensitive information from being accessed by unauthorized people.
Cryptography	The study of techniques used to make plain information unreadable, as well as to convert it back to a readable form.
DDoS Attack	An attack in which multiple compromised systems are used to attack a single target. The flood of incoming messages to the target system forces it to shut down and denies service to legitimate users.
Decryption	A process that converts encrypted voice or data information into plain form by reversing the encryption process.
Digital Signature	A cryptologic mechanism used to validate an item's (e.g. document, software) authenticity and integrity.
Encryption	Converting information from one form to another to hide its content and prevent unauthorized access.

Term	Definition
Forward Secrecy	A property of key establishment protocols where the compromise of the long-term private key will not allow an adversary to re-compute previously derived keys or sessions.
Integrity	The ability to protect information from being modified or deleted unintentionally when it's not supposed to be. Integrity helps determine that information is what it claims to be. Integrity also applies to business processes, software application logic, hardware, and personnel.
Key Management	The procedures and mechanisms for generating, disseminating, replacing, storing, archiving, and destroying cryptographic keys.
Replay Attack	A form of network attack in which a valid data transmission is maliciously or fraudulently repeated or delayed.

9.3 REFERENCES

Number	Reference
1	Canadian Centre for Cyber Security. *ITSP.40.111 Cryptographic Algorithms for UNCLASSIFIED, PROTECTED A, and PROTECTED B Information*. August 2016.
2	Treasury Board of Canada Secretariat. *Guideline on Defining Authentication Requirements*. November 2008.
3	Canadian Centre for Cyber Security. *ITSG-33 IT Security Risk Management: A Lifecycle Approach.* December 2014.
4	National Institute of Standards and Technology. *Special Publication 800-57 Part 3 Rev 1 Recommendation for Key Management Part 3: Application-Specific Key Management Guidance.* January 2015.
5	Cooper, D., et al. *RFC 5280 Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*. Internet RFCs. ISSN 2070-1721. May 2008.
6	Rescola, E. *RFC 8446 The Transport Layer Security (TLS) Protocol Version 1.3.* Internet RFCs. ISSN 2070-1721. August 2018.
7	Hodges, J., C. Jackson, and A. Barth. *RFC 6797 HTTP Strict Transport Security (HSTS).* Internet RFCs. ISSN 2070-1721. November 2012.
8	Hoffman, P.*RFC 3207 SMTP Service Extension for Secure SMTP over Transport Layer Security.* Internet RFCs. ISSN 2070-1721. February 2002.
9	Dukhovni, V., et al. *RFC 7672 SMTP Security via Opportunistic DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS).* Internet RFCs. ISSN 2070-1721. October 2015.
10	Ray, M. and S. Dispensa. *RFC 5746 Transport Layer Security (TLS) Renegotiation Indication Extension.* Internet RFCs. ISSN 2070-1721. February 2010.
11	Kaufman, C., et al. *RFC 7296 Internet Key Exchange Protocol Version 2 (IKEv2)*. Internet RFCs ISSN 2070-1721. October 2014.
12	Kivinen, T., and J. Snyder. *RFC 7427 Signature Authentication in IKEv2.* Internet RFCs. ISSN 2070-1721. January 2015.
13	Hoffman, P. and J. Snell. *RFC 7396 JSON Merge Patch*. Internet RFCs. ISSN 2070-1721. October 2014.
14	Eronen, P. and H. Tschofeniq. *RFC 5998 An Extension of EAP-Only Authentication in IKEv2.* Internet RFCs. ISSN 2070-1721. September 2010.

Number	Reference
15	Nir, Y. and V. Smyslov. *RFC 8019 Protecting IKEv2 Implementations from Distributed Denial-of-Service Attacks*. Internet RFCs. ISSN 2070-1721. November 2016.
16	*RFC 7383 Internet Key Exchange Protocol Version 2 (IKEv2) Message Fragmentation*.
17	*RFC 5723 Internet Key Exchange Protocol Version 2 (IKEv2) Session Resumption.*
18	Kent, S., and K. Seo. *RFC 4301 Security Architecture for the Internet Protocol.* Internet RFCs. ISSN 2070-1721. December 2005.
19	Kent, S. *RFC 4303 IP Encapsulating Security Payload (ESP).* Internet RFCs. ISSN 2070-1721. December 2005.
20	McGrew, D., and P. Hoffman. *RFC 7321 Cryptographic Algorithm Implementation Requirements and Usage Guidance for Encapsulating Security Payload (ESP) and Authentication Header (AH).* Internet RFCs. ISSN 2070-1721. August 2014.
21	Ylonen, T., and C. Lonvick, Ed. *RFC 4251 The Secure Shell (SSH) Protocol Architecture.* Internet RFCs. ISSN 2070-1721. January 2006.
22	Ylonen, T., and C. Lonvick, Ed. *RFC 4252 The Secure Shell (SSH) Authentication Protocol.* Internet RFCs. ISSN 2070-1721. January 2006.
23	Ylonen, T., and C. Lonvick, Ed. *RFC 4253 The Secure Shell (SSH) Transport Layer Protocol.* Internet RFCs. ISSN 2070-1721. January 2006.
24	Ylonen, T., and C. Lonvick, Ed. *RFC 4254 The Secure Shell (SSH) Connection Protocol.* Internet RFCs. ISSN 2070-1721. January 2006.
25	Igoe, K., and D. Stebila. *RFC 6187 x509.v3 certificates for Secure Shell Authentication.* Internet RFCs. ISSN 2070-1721. March 2011.
26	Bellare, M., T. Kohno, and C. Namprempre. *RFC 4344 The Secure Shell (SSH) Transport Layer Encryption Modes*. Internet RFCs. ISSN 2072-1721. January 2006.
27	Schaad, J., B. Ramsdell, and S. Turner. *RFC 8551 Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 4.0 Message Specification.* Internet RFCs. ISSN 2070-1721. April 2019.
28	Schaad, J., B. Ramsdell, and S. Turner. *RFC 8550 Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 4.0 Certificate Handling.* Internet RFCs. ISSN 2070-1721. April 2019.
29	Turner, S., and D. Brown. *RFC 5753 Use of Elliptic Curve Cryptography (ECC) Algorithms in Cryptographic Message Syntax (CMS).* Internet RFCs. ISSN 2070-1721. January 2010.
30	Turner, S., et al. *RFC 5756 Updates for RSAES-OAEP and RSASSA-PSS Algorithm Parameters.* Internet RFCs. ISSN 2070-1721. January 2010.
31	Schaad, J. *RFC 4056 Use of the RSASSA-PSS Signature Algorithm in Cryptographic Message Syntax (CMS).* Internet RFCs. ISSN 2070-1721. June 2005.
32	Turner, S., and J. Schaad. *RFC 5752 Multiple Signatures in Cryptographic Message Syntax (CMS).* Internet RFCs. ISSN 2070-1721. January 2010.
33	Turner, S. and Brown, D. *RFC 5753 Use of Elliptic Curve Cryptography (ECC) Algorithms in Cryptographic Message Syntax (CMS).* Internet RFCs. ISSN 2070-1721. January 2010.
34	Housley, R. *RFC 3560 Use of the RSAES-OAEP Key Transport Algorithm in Cryptographic Message Syntax (CMS).* Internet RFCs. ISSN 2070-1721. July 2003.

Number	Reference
35	Margolis, D. Risher, M., Ramakrishnan, B., Brothman, A., Jones, J. *RFC 8461 SMTP MTA Strict Transport Security (MTA-STS).* ISSN: 2070-1721. September 2018.