Common Criteria Certification
BSI-DSZ-CC-xyz BSI-CC-PP-00zz

Funktionale Spezifikation

MAUVECORP MAUVEVPN CLIENT
Version 2.11

MauveCorp
Fliederweg 98
50020 Köln
certification@mauvecorp.com

Dokumentversion 1.0-SNAPSHOT
[Commit 7d63d7c / master]
2020-03-02

Inhaltsverzeichnis

Tabellenverzeichnis

Abbildungsverzeichnis

1. Einführung

Dieses Dokument enthält die notwendigen Informationen zu Evaluation der Vertrauenswürdigkeitskomponente ADV_FSP.4 für die Evaluation des MauveVPN Client.

Das Dokument beschreibt zunächst die physikalischen und logischen Schnittstellen, welche die Sicherheitsfunktionalität des TOE (TSF) betreffen. Im Anschluss werden alle Sicherheitsfunktionen des TOE ausführlich beschrieben. Es wird dokumentiert, über welche Schnittstelle die Sicherheitsfunktionen aktiviert werden.

Anmerkungen zur Notation

Tabelle 1.1 auf der nächsten Seite listet die in diesem Dokument verwendeten typographischen Auszeichnungen und ihre Verwendung auf. Oftmals ist die Abgrenzung zwischen den in der Tabelle aufgeführten Kategorien nicht ganz leicht: So kommt es gelegentlich vor, dass auf verschiedenen Abstraktionsebenen dieselben Begriffe verwendet werden. Es ist in solchen Fällen nicht auf den ersten Blick zu erkennen, auf welcher Abstraktionsebene der Begriff gerade verwendet wird. Beispielsweise kann ein Begriff gleichzeitig ein Konfigurationsparameter aus der Spezifikation und gleichzeitig der Name einer Variable im Code sein. Durch eine möglichst hohe typographische Konsistenz soll der jeweilige Kontext verdeutlicht werden. Die Hinweise in Tabelle 1.1 sollen helfen, die Begrifflichkeiten einzuordnen und voneinander abzugrenzen.

Module und Subsysteme werden durch doppelte Doppelpunkte in der Form Subsystem::Modul notiert. Wird in einem solchen Kontext auf eine Schnittstelle verwiesen, wird der Name der Schnittstelle durch zwei Schrägstriche vom Namen des Moduls getrennt: Subsystem::Modul//Schnittstelle.

Die ihm Text zitierten Namen von Code-Elementen – besonders die Komponenten im Java-Umfeld – können zum Teil lang werden. In solchen Fällen werden die Namen der besseren Lesbarkeit halber mit Trennstrichen versehen, die in dieser Form nicht im Code sichtbar sind. Diese minimale Abweichung wird in Kauf genommen, um dem Text ein harmonisches Gesamtbild zu verleihen, das nicht durch übermäßigen Weißraum gestört wird.

Typographische Auszeichnung	Verwendung
Schlüsselwörter	*Schlüsselwörter* sind solche Begriffe, die z. B. direkt aus der Spezifikation entnommen sind, dies können Konfigurationsparameter und ihre Werte sein. Aber auch andere Begriffe, die im Kontext des TOE eine hervorgehobene Bedeutung haben, sind so ausgezeichnet.
`Code-Elemente`	`Code-Elemente` sind solche Begriffe, die unmittelbar aus dem Source-Code des TOE entnommen sind. Dies können beispielsweise Java-Bundles, Klassen- oder Methodennamen sein, aber auch deren Parameter oder logische Strukturen in einer der verwendeten Programmiersprachen.
Dateinamen	*Dateinamen* beziehen sich auf Namen oder Namensteile von Elementen im Dateisystem.
Sicherheitsbezogene Begriffe	Begriffe, die in direktem Bezug zum Rahmenwerk der Common Criteria stehen, sind als Sicherheitsbezogene Begriffe gesetzt. Dies sind SFR, die Namen der SF und TSFI, aber auch die Namen der Subsysteme, Module und Schnittstellen, die den TOE konstituieren. Weiterhin sind Namen der vom TOE verwendeten Zertifikate und sonstigen Schlüsselmaterials in dieser Form gesetzt.

Tabelle 1.1.: Typographische Konventionen

2. Physische Schnittstellen

2.1. PS.LAN

Über die Netzwerkschnittstellen können Clientsysteme oder andere Systeme im LAN mit dem TOE kommunizieren. Die logische Schnittstelle LS.LAN wird über diese physische Schnittstelle bereitgestellt.

Obwohl der TOE ein reiner Software-TOE ist, spielt auch die Hardware, auf der der TOE läuft, eine wichtige Rolle für die Sicherheit. Aus diesem Grund betrachten wir hier kurz einige Eigenschaften der physischen Schnittstellen, die im MauveVPN Client verbaut sind. Die Beschreibung gilt gleichermaßen für PS.LAN wie für PS.WAN.

2.2. PS.WAN

Eine analog zu PS.LAN ausgeführte Ethernet-Schnittstelle zu Datennetzen (WAN)…

2.3. PS.LED

Der TOE ist mit LED ausgestattet, die im Takt blinken. Über die physische Schnittstelle wird die logische Schnittstelle LS.LED erreicht, über die der Nutzer Informationen über den Zustand des TOE abrufen kann.

3. Logische Schnittstellen

In diesem Kapitel werden die logischen Schnittstellen des TOE beschrieben.

In den Unterabschnitten dieses Kapitel gibt es zu jeder Schnittstelle eine grafische Darstellung über die Protokolle, die auf der Schnittstelle zum Einsatz kommen. Dabei sind diejenigen Protokolle, die zu einer TSFI gehören, orange markiert. Gelb markierte Protokolle gehören nicht zu einer TSFI. Gepunktet markierte Protokolle sind Schnittstellen für non-TSF Anteile des TOE – also non-TSFI. Sie werden trotzdem aufgeführt, da sie als Außenschnittstelle des TOE beschreibungsrelevant sind.

3.1. LS.LAN

LS.LAN ist eine logische Schnittstelle zu den Clientsystemen, die physisch über das LAN (PS.LAN) erreichbar sind.

Die Schnittstelle umfasst die folgenden Protokolle, Tabelle 3.1 listet diese Protokolle und die dafür verwendeten Portnummern. Abbildung 3.1 stellt die Protokolle dar, welche die sicherheitsrelevanten Aspekte des TSFI ausmachen (vgl. die einleitende Bemerkung zu Kapitel 3).

Ethernet für Netzzugang,

IP für Routing,

TCP und UDP für die Transportschicht,

DHCP für IP-Adressvergabe im LAN (Client),

TLS für die Absicherung der Kommunikation der anderen logischen Schnittstellen im LAN.

HTTP_Mgmt für den Zugriff auf die Management-Schnittstelle

Tabelle 3.1 listet die Protokolle und verwendeten Ports detailreicher auf. Abbildung 3.1 zeigt die Protokolle der Schnittstelle LS.LAN in Relation zueinander und bezogen auf das TCP/IP Schichtenmodell. Der Protokollstapel ist aus Platzgründen auf zwei Abbildungen aufgeteilt.

<table>
<tr><td rowspan="2">Anwendung</td><td rowspan="2">—</td><td>HTTP
Mgmt</td></tr>
<tr><td>TLS</td></tr>
<tr><td>Transport</td><td>UDP</td><td>TCP</td></tr>
<tr><td>Internet</td><td colspan="2">IPv4</td></tr>
<tr><td>Netzzugang</td><td colspan="2">Ethernet</td></tr>
</table>

Abbildung 3.1.: Protokolle auf LS.LAN für die sicherheitsfunktionalen Anteile

Dienst	In/Out	Protokoll	via	Quellport	Zielport	TSFI	Anmerkung
Basisprotokolle	–	IEEE802.3	–	–	–	LS.LAN.Ether	
	–	IPv4	IEEE802.3	–	–	LS.LAN.IP	
	–	TCP	IPv4	–	–	LS.LAN.TCP	
	–	UDP	IPv4	–	–	LS.LAN.UDP	
Administration	In	TLS	TCP	bel.	9443	LS.LAN.TLS	
	In	HTTP	TLS	bel.	9443	LS.LAN.HTTP_MGMT	

Tabelle 3.1.: Übersicht der Protokoll- und Portnummern für IP/TCP/UDP auf LS.LAN

3.1.1. LS.LAN.Ether

3.1.1.1. Zweck und Nutzungsbedingungen

Diese Schnittstelle dient als *Netzzugangsschicht* zum Ethernet-Netzwerk.

Über das Interface aufgerufene Sicherheitsfunktionen
Sicherheitsfunktionen, die über das Interface LS.LAN.Ether aufgerufen werden: (keine)

3.1.1.2. Parameter

Die Schnittstelle implementiert das Ethernet-Protokoll nach [IEEE802.3].

3.1.2. LS.LAN.IP

3.1.2.1. Zweck und Nutzungsbedingungen

Auf der *Internetschicht* verhält sich der TOE als IP-Router und unterstützt das Internet-Protokoll in der Version 4. Zusätzlich wird das ICMP-Protokoll unterstützt, welches Teil von IPv4 ist.

Über das Interface aufgerufene Sicherheitsfunktionen
Sicherheitsfunktionen, die über das Interface LS.LAN.IP aufgerufen werden: (keine)

3.1.2.2. Parameter

Die Implementierung von IPv4 entspricht den Vorgaben aus RFC 791 [RFC 791], RFC 1812 [RFC 1812] und den Aktualisierungen in RFC 2644 [RFC 2644]. ICMP ist in RFC 792 [RFC 792] spezifiziert. Die Implementierung von IPv4 wird durch den Linux-Kernel bereitgestellt.

3.1.3. LS.LAN.TCP

3.1.3.1. Zweck und Nutzungsbedingungen

Auf der *Transportschicht* unterstützt der TOE das Transmission Control Protocol (TCP).

Über das Interface aufgerufene Sicherheitsfunktionen
Sicherheitsfunktionen, die über das Interface LS.LAN.TCP aufgerufen werden: (keine)

3.1.3.2. Parameter

Die Implementierung von TCP entspricht den Vorgaben aus RFC 793 [RFC 793]. Die Implementierung von TCP wird durch den Linux-Kernel bereitgestellt.

3.1.4. LS.LAN.UDP

3.1.4.1. Zweck und Nutzungsbedingungen

Der TOE unterstützt auf der *Transportschicht* zusätzlich das User Datagram Protocol (UDP).

Über das Interface aufgerufene Sicherheitsfunktionen
Sicherheitsfunktionen, die über das Interface LS.LAN.UDP aufgerufen werden: (keine)

3.1.4.2. Parameter

Die Implementierung von UDP entspricht den Vorgaben aus RFC 768 [RFC 768]. Die Implementierung von UDP wird durch den Linux-Kernel bereitgestellt.

3.1.5. LS.LAN.TLS

3.1.5.1. Zweck und Nutzungsbedingungen

LS.LAN.TLS wird verwendet, um Verbindungen zu anderen Systemen im LAN abzusichern. TLS stellt Vertraulichkeit und Integrität der Verbindungen sicher. Eine Übersicht über die TLS Verbindungen und deren Konfiguration des TOE ist in Tabelle Tabelle B.4 zu finden.

Über das Interface aufgerufene Sicherheitsfunktionen
Sicherheitsfunktionen, die über das Interface LS.LAN.TLS aufgerufen werden: SF.CryptographicServices, SF.TLS

3.1.5.2. Parameter

Die Implementierung von TLS im TOE setzt [RFC 5246] um. Die Beschreibung der TLS-Implementierung sowie allen TLS-Verbindungen des TOE gemeinsamen Parameter und TOE-spezifische Anpassungen der TLS-Implementierung sind in Kapitel 4.6 dokumentiert.

3.1.6. LS.LAN.HTTP_MGMT

3.1.6.1. Zweck und Nutzungsbedingungen

Der TOE implementiert einen HTTP-Server, um die Konfiguration des TOE über die JSON-Schnittstelle der Webanwendung zu ermöglichen. Die Benutzung der Webanwendung wird im Administratorhandbuch beschrieben, das als Dokumenttyp AGD_ADM ebenfalls Teil der Sicherheitsdokumentation ist [AGD_ADM]. Der TOE liefert an der Schnittstelle LS.LAN.HTTP_MGMT den Code der Webanwendung an den Web-Browser des Administrators aus. Der Browser interpretiert den Code und führt ihn aus. Der im Browser ablaufende Code interagiert mit der JSON-API. Die Frontend-Elemente der Webanwendung (also die HTML-, CSS- und JavaScript-Elemente) selbst werden nicht näher beschrieben. Sie setzen keine Sicherheitsanforderungen um. Der Grund dafür ist, dass dieses Frontend auf einem Browser des Anwenders, also in der Umgebung des TOE, ausgeführt wird und zur Laufzeit nicht unter der Kontrolle der TSF steht. Der Nutzer hat volle Kontrolle über die Ausführungsplattform des Frontends und kann beliebig in den dort ablaufenden JavaScript Code eingreifen. Die Sicherheitsanforderungen der Managementschnittstelle werden folglich ausschließlich von den serverseitigen Komponenten erbracht.

Das Protokoll zur Übertragung von Konfigurationsparametern ist in Form einer JSON-API implementiert. Die serverseitigen Komponenten der Webanwendung nehmen die fachlichen Werte als API-Aufrufe entgegen. Darüber hinaus sorgen sie für die Authentisierung des Administrators und den Schutz vor XSS- und CSRF-Angriffen. Die konkrete Ausgestaltung dieser Schutzmaßnahmen wird in der Sicherheitsarchitektur beschrieben [ADV_ARC]. Die Prüfung des Authentisierungsstatus erfolgt fortlaufend, also bei jedem Request.

Über das Interface aufgerufene Sicherheitsfunktionen

Sicherheitsfunktionen, die über das Interface LS.LAN.HTTP_MGMT aufgerufen werden: SF.SelfProtection, SF.Administration

3.1.6.2. Parameter

Der HTTP-Server implementiert RFC 2616 [RFC 2616]. Die Verbindung ist über TLS abgesichert, vgl. die Beschreibung zu LS.LAN.TLS und Abschnitt 4.6.

Die Konfigurationswerte werden als JSON-Objekte an die Schnittstelle übergeben. Das Transportprotokoll ist HTTP, wobei die Schnittstelle als REST API zu verwenden ist. Es wird streng unterschieden zwischen Requests, mit denen statische Elemente angefordert werden, und solchen, die zu schützende Daten wie Session-IDs oder Benutzerdaten enthalten. Letztere werden ausschließlich über POST-Requests angefordert, sodass keine zu schützenden Elemente in der Browser-Historie verbleiben. Die Parameter und die Verwendung der Schnittstelle werden separat in einer separaten Dokumentation beschrieben. JSON wird in RFC 7159 [RFC 7159] definiert.

3.2. LS.WAN

In diesem Abschnitt werden die Protokolle beschrieben, die der TOE über seine Schnittstelle LS.WAN abwickelt. Tabelle 3.2.5.1 listet diese Protokolle und die dafür verwendeten Portnummern. Abbildung 3.2 stellt die Protokolle dar, welche die sicherheitsrelevanten Aspekte des TSFI ausmachen (vgl. die einleitende Bemerkung zu Kapitel 3).

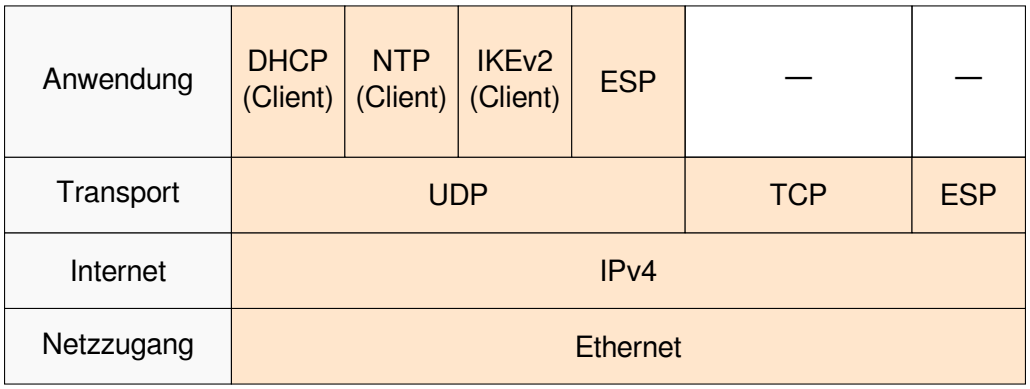

Abbildung 3.2.: Protokolle auf LS.WAN für die sicherheitsfunktionalen Anteile

3.2.1. LS.WAN.Ether

Die Implementierung von Ethernet an der WAN-Schnittstelle entspricht in allen Aspekten der Implementierung an der LAN-Schnittstelle, vgl. Abschnitt 3.1.1 auf Seite 12.

3.2.2. LS.WAN.IP

Die Implementierung von IPv4 und ICMP an der WAN-Schnittstelle entspricht in allen Aspekten der Implementierung an der LAN-Schnittstelle, vgl. Abschnitt 3.1.2 auf Seite 12.

3.2.3. LS.WAN.TCP

Die Implementierung von TCP an der WAN-Schnittstelle entspricht in allen Aspekten der Implementierung an der LAN-Schnittstelle, vgl. Abschnitt 3.1.3 auf Seite 12.

3.2.4. LS.WAN.UDP

Die Implementierung von UDP an der WAN-Schnittstelle entspricht in allen Aspekten der Implementierung an der LAN-Schnittstelle, vgl. Abschnitt 3.1.4 auf Seite 12.

3.2.5. LS.WAN.DHCP

3.2.5.1. Zweck und Nutzungsbedingungen

Der TOE bietet die Möglichkeit, auf der WAN-Schnittstelle als DHCP-Client zu agieren. Der TOE kann also seine IP-Adresse von einem DHCP-Server im WAN beziehen. Diese Funktion ist auf der Managementoberfläche (de-)aktivierbar.

Dienst	In/Out	Protokoll	via	Quellport	Zielport	TSFI	Anmerkung
Basisprotokolle	–	IEEE802.3	–	–	–	LS.WAN.Ether	
	–	IPv4	IEEE802.3	–	–	LS.WAN.IP	
	–	TCP	IPv4	–	–	LS.WAN.TCP	
	–	UDP	IPv4	–	–	LS.WAN.UDP	
IPSec	Out	IKEv2	UDP	dyn.	500	LS.WAN.IPSec	
	Out	IKEv2	UDP	dyn.	4500	LS.WAN.IPSec	bei UDP-Encapsulation
	–	ESP	IPv4	–	–	LS.WAN.IPSec	
	–	ESP	UDP	dyn.	4500	LS.WAN.IPSec	bei UDP-Encapsulation
Zeitdienst	Out	NTP	UDP	–	123	LS.WAN.NTP	
DHCP-Service	Out	DHCP	UDP	68	67	LS.WAN.DHCP	

Tabelle 3.2.: Übersicht der Protokoll- und Portnummern für IP/TCP/UDP auf LS.WAN

Über das Interface aufgerufene Sicherheitsfunktionen

Sicherheitsfunktionen, die über das Interface LS.WAN.DHCP aufgerufen werden[1]: SF.NetworkServices

3.2.5.2. Parameter

Das DHCP-Protokoll setzt auf dem UDP-Protokoll auf und ist in RFC 2131 [RFC 2131] beschrieben. Die Beschreibungen enthalten insbesondere das Format einer DHCP-Nachricht und die darin enthaltenen Felder, die verschiedenen Nachrichtentypen und den Ablauf der Kommunikation.

Das RFC 2132 [RFC 2132] dokumentiert zudem weitere DHCP-Optionen, die der DHCP-Server mit dem DHCP-Client austauschen kann.

3.2.6. LS.WAN.NTP

3.2.6.1. Zweck und Nutzungsbedingungen

Die Schnittstelle wird genutzt, um die Systemzeit mit Zeitservern aus dem Internet zu synchronisieren. (FPT_STM.1)

Über das Interface aufgerufene Sicherheitsfunktionen

Sicherheitsfunktionen, die über das Interface LS.WAN.NTP aufgerufen werden: SF.NetworkServices

3.2.6.2. Parameter

Die Protokollbeschreibung ist in RFC 5905 [RFC 5905] dokumentiert. Der Zielport für den Client ist UDP-Port 123.

3.2.7. LS.WAN.IPSec

3.2.7.1. Zweck und Nutzungsbedingungen

Der TOE verwendet die WAN-Schnittstelle für den Aufbau des VPN-Kanals Der TOE bietet dafür IPsec als VPN-Client an der WAN-Schnittstelle an.

Über das Interface aufgerufene Sicherheitsfunktionen

Sicherheitsfunktionen, die über das Interface LS.WAN.IPSec aufgerufen werden: SF.CryptographicServices, SF.VPN

3.2.7.2. Parameter

IPSec berücksichtigt die Vorgaben aus RFC 4301 [RFC 4301]. Der TOE verwendet für IPSec zwei Protokolle. Zum Schlüsselaustausch wird das Protokoll Internet Key Exchange (IKE) in der Version 2 nach RFC 7296 [RFC 7296] am UDP-Port 500 verwendet. Für die Übertragung der Nutzdaten wird das Encapsulation Security Payload (ESP) nach RFC 4303 [RFC 4303] (IP-Protokollnummer 50, bzw. UDP-Port 4500) umgesetzt.

[1] Diese Zuordnung lässt sich nicht durch die gemeinsame Zuordnung eines SFR nachvollziehen.

3.2.7.3. Internet Key Exchange Version 2 (IKEv2)

IKE selbst ist nicht für die Übertragung von Nutzdaten verantwortlich, sondern lediglich für die Verteilung von geeignetem Schlüsselmaterial und die Authentifizierung der zwei Parteien. Die Übertragung der eigentlichen Nutzdaten ist die Aufgabe des ESP-Protokolls (vgl. Abschnitt 3.2.7.4).

Der TOE als Initiator der VPN-Verbindungen schlägt zu Beginn des IKE-Protokolls die zu verwendenden Algorithmen vor.

3.2.7.4. Encrypted Security Payload (ESP)

Das ESP-Protokoll gewährleistet Authentizität, Integrität und Vertraulichkeit der übermittelten Daten. Dabei werden getrennte Algorithmen für Verschlüsselung und Integritätssicherung verwendet.

Die in IKE für ESP ausgehandelte SA inklusive der benötigen Schlüssel und Algorithmen wird nun verwendet, um die IP-Pakete, die über den Tunnel verschickt werden sollen, zu sichern. Dazu wird die Struktur des ESP-Headers verwendet, wie in Abbildung 3.3 dargestellt (s. a. [RFC 4303, Figure 2])

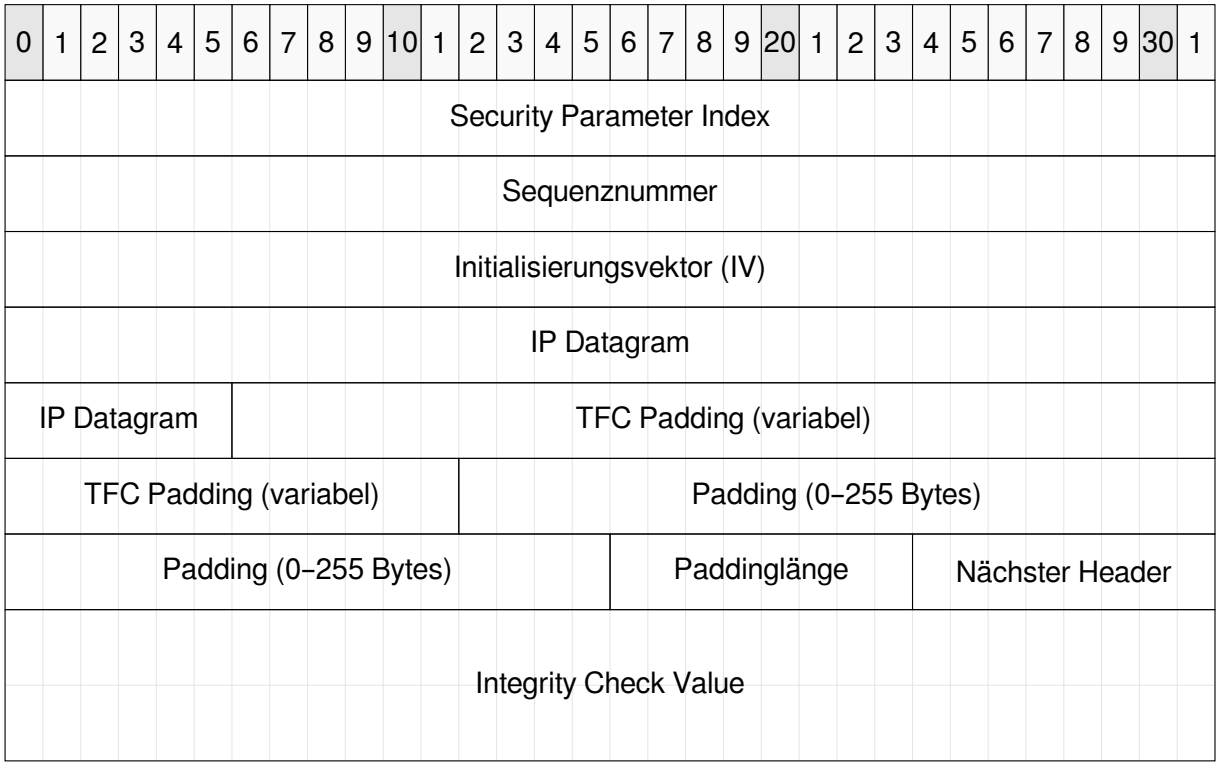

Abbildung 3.3.: ESP Header

3.3. LS.LED

Über diese Schnittstelle kann der TOE des Status auf PS.LED anzeigen. LS.LED wird vom Betriebssystem genutzt. Die Bedeutung der Anzeige ist in [AGD_ADM, Kapitel 6] beschrieben.

4. Sicherheitsfunktionen des TOE

4.1. VPN-Client (SF.VPN)

Schnittstellen der Sicherheitsfunktion

Der Aufruf der Sicherheitsfunktion SF.VPN erfolgt über: LS.WAN.IPSec.

Konfigurationsparameter der Sicherheitsfunktion

Die Konfiguration der Sicherheitsfunktion wird im Administratorhandbuch beschrieben [AGD_ADM, Abschnitt 7.4.3.3 VPN (Virtual Private Network)].

Beschreibung der Sicherheitsfunktion

Der TOE stellt einen VPN Service (Client) bereit, mit dem der TOE sichere Kanäle zwischen sich selbst und VPN-Konzentratoren aufbauen kann. Diese Kanäle werden logisch von anderen Kanälen getrennt gehalten und stellen eine Identifikation und Authentisierung der Endpunkte des Kanals sicher. Durch den Kanal werden die übertragenen Daten in Bezug auf ihre Integrität und Vertraulichkeit geschützt. Der Aufbau geschieht über die Schnittstelle LS.WAN.

4.1.1. Authentifizierung und Schlüsselaushandlung via IKE

Die Schlüsselvereinbarung wird mittels IKEv2-Protokoll nach RFC 4306 [RFC 4306] umgesetzt. Hierbei wird alleinig die Diffie-Hellman-Gruppe 14 nach RFC 3526 [RFC 3526] unterstützt. Der TOE verwendet für den Schlüsselaustausch einen DH-Exponent der Länge 384 Bit.

Bereits verwendete Diffie-Hellman-Schlüssel werden nicht erneut verwendet, um (Perfect) Forward Secrecy zu gewährleisten.

Initiator (TOE) und Responder (VPN-Konzentrator) werden im Rahmen des IKE-Protokolls gegenseitig zertifikatsbasiert authentifiziert (FTP_ITC.1/VPN, FCS_CKM.2/IKE) und handeln Schlüssel für die Verschlüsselung und Integritätssicherung der Nutzdaten aus. Die Signaturprüfung mit öffentlichen Schlüsseln und die symmetrischen Algorithmen werden in Software umgesetzt (s. Abschnitt 4.5 auf Seite 22).

Der TOE erzeugt die entsprechenden symmetrischen Schlüssel, die notwendig sind, um die Daten des IKE-Protokolls und des ESP-Protokolls zu verschlüsseln und HMACs für die Integritätssicherung zu erzeugen.

4.1.2. Gültigkeitsprüfung von Zertifikaten

Die Zertifikate der Konzentratoren werden durch den TOE sowohl auf ihre mathematische Korrektheit, auf ihre zeitliche Gültigkeit geprüft. Außerdem wird der Sperrstatus der Zertifikate unter Verwendung von CRLs geprüft (FPT_TDC.1/Zert).

4.2. Netzbasierte Sicherheitsfunktionen (SF.NetworkServices)

Schnittstellen der Sicherheitsfunktion
Der Aufruf der Sicherheitsfunktion SF.NetworkServices erfolgt über: LS.WAN.NTP.

Konfigurationsparameter der Sicherheitsfunktion
Die Konfiguration der Sicherheitsfunktion wird im Administratorhandbuch beschrieben [AGD_ADM, Abschnitt 7.4].

Beschreibung der Sicherheitsfunktion
Der TOE stellt die folgenden Dienste im Netzwerk zur Verfügung (In Klammern das entsprechende Kapitel im Administratorhandbuch [AGD_ADM]):

- NTP-Client (Abschnitt 7.4.3.4)

4.2.1. NTP-Client

Der TOE implementiert einen NTP-Client. Die Sicherheitsfunktion SF.NetworkServices stellt anderen Komponenten des TOE die genaue Uhrzeit bereit (FPT_STM.1).

4.2.2. DHCP-Client

Der TOE bietet an WAN-Schnittstelle (LS.WAN.DHCP) jeweils die Möglichkeit DHCP nach RFC 2131 [RFC 2131] und RFC 2132 [RFC 2132] zu nutzen, um IP-Adressen, Default-Routen und DNS-Server zu beziehen. Die Konfiguration wird im Administratorhandbuch beschrieben [AGD_ADM].

4.3. Selbstschutz (SF.SelfProtection)

Schnittstellen der Sicherheitsfunktion
Der Aufruf der Sicherheitsfunktion SF.SelfProtection erfolgt über: Keine TSFI, LS.LAN.HTTP_MGMT.

Konfigurationsparameter der Sicherheitsfunktion
Die Sicherheitsfunktion verfügt über keine Konfigurationsparameter.

Beschreibung der Sicherheitsfunktion
Diese Sicherheitsfunktion ist verantwortlich für den Selbstschutz des TOE und für den Schutz von Daten, die durch den TOE übertragen werden.

4.3.1. Speicheraufbereitung

Sensible Daten wie kryptografische Schlüssel werden sicher aus dem Speicher des TOE gelöscht sobald diese nicht mehr verwendet werden.

Die Löschung wird durch aktives Überschreiben der entsprechenden Speicherbereiche mit Nullen erreicht (FDP_RIP.1). Die jeweiligen Implementierungen des sicheren Löschens für die verschiedenen Module des TOE sind in der Design-Spezifikation beschrieben [ADV_TDS].

Diese Funktionalität hat keinerlei öffentlich verfügbare Schnittstelle.

4.3.2. Selbsttests

Der TOE implementiert Selbsttests, mit denen die Integrität und die korrekte Funktionsweise des TOE überprüft werden kann.

Zur Laufzeit kann die Durchführung der Tests durch den Administrator über die Managementanwendung gestartet werden (FPT_TST.1).

4.4. Administration (SF.Administration)

Schnittstellen der Sicherheitsfunktion

Der Aufruf der Sicherheitsfunktion SF.Administration erfolgt über: LS.LAN.HTTP_MGMT.

Konfigurationsparameter der Sicherheitsfunktion

Die Funktionalität zur Konfiguration des TOE wird im Administratorhandbuch beschrieben [AGD_ADM, Abschnitte 7.3 bis 7.6].

Beschreibung der Sicherheitsfunktion

Im TOE bieten die Dienste ihre Managementfunktionen jeweils über eine dedizierte Schnittstelle an, die durch den Administrationsdienst angesprochen wird.

4.5. Kryptografische Dienste (SF.CryptographicServices)

Schnittstellen der Sicherheitsfunktion

Der Aufruf der Sicherheitsfunktion SF.CryptographicServices erfolgt über: LS.LAN.TLS, LS.WAN.IPSec.

Konfigurationsparameter der Sicherheitsfunktion

Die Aufgabe von SF.CryptographicServices ist das Bereitstellen von TLS-Verbindungen zur Managementanwendung.

Beschreibung der Sicherheitsfunktion

Der TOE stellt kryptographische Funktionen bereit, die von den anderen Sicherheitsfunktionen verwendet werden.

4.5.1. Zufallszahlen

Der TOE enthält einen DRNG nach FCS_RNG.1/Hash_DRBG, um Zufallszahlen hoher Qualität zu erzeugen [NIST SP 800-90A].

Der von SF.CryptographicServices bereitgestellte Zufallsgenerator wird verwendet, um die Zufallszahlen und Nonces beim TLS-Verbindungsaufbau (FCS_CKM.1 und FCS_COP.1/TLS.AES) zu erzeugen. Über den TLS-Verbindungsaufbau hinaus wird der Zufallsgenerator immer dann herangezogen, wenn Schlüssel erzeugt werden müssen.

4.5.2. HMAC-Algorithmen

Die Funktion bietet Implementierungen der Algorithmen für die HMAC-Generierung, wobei die Hash-Algorithmen HMAC-SHA-1(-96), HMAC-SHA-256(-128) umgesetzt werden (FCS_COP.1/HMAC).

4.5.3. Signaturverifikation

Der TOE unterstützt die Verifikation von Signaturen.

4.5.3.1. Pseudo Random Function für die Generierung symmetrischer Schlüssel (PRF)

Der TOE erzeugt pseudozufällige Zahlen für die Erzeugung von symmetrischem Schlüssel für Verschlüsselung und Integritätssicherung im Rahmen der Protokolle IKE (IKE_ENCR und IKE_INTEG) und ESP (ESP_ENCR und ESP_INTEG).

Die Algorithmen PRF-HMAC-SHA-256 und PRF-HMAC-SHA-1 werden vom TOE unterstützt.

RFC 2104 spezifiziert den HMAC-Algorithmus für beliebige Hash-Algorithmen [RFC 2104]. FIPS180-4 [FIPS PUB 180-4] spezifiziert die verwendeten Hash-Algorithmen. [RFC 4868] beschreibt die PRFs auf Basis der HMACs, wie sie bei IKE eingesetzt werden (FCS_CKM.1, FCS_CKM.2/IKE).

4.5.3.2. Schlüsselaustausch für IKE (Diffie-Hellman)

Der TOE implementiert den Diffie-Hellman-Algorithmus für den Schlüsselaustausch im Rahmen des IKE-Protokolls IKEv2 [RFC 7296]. Der TOE stellt zudem sicher, dass Diffie-Hellman-Schlüssel nur für eine Sitzung verwendet werden und danach neu erzeugt werden, um (Perfect) Forward Secrecy zu gewährleisten. (FCS_CKM.2/IKE)

4.5.3.3. Schlüsselzerstörung

Der TOE zerstört die symmetrischen Schlüssel für IKE und ESP und die Diffie-Hellman-Schlüssel, die im Rahmen von IKE erzeugt werden, indem er die verwendeten Speicherbereiche nach der Verwendung komplett mit Nullen überschreibt. (FCS_CKM.4)

4.6. TLS-Service (SF.TLS)

Der TOE implementiert das TLS-Protokoll in der Version 1.2. Dabei werden die Cipher Suites in Tabelle B.1 unterstützt. Transport Layer Security Version 1.2 ist in RFC 5246 spezifiziert [RFC 5246]. Weiterhin relevant sind RFC 8017 (RSA) für den Nachweis des Schlüsselbesitzes [RFC 8017] sowie RFC 3526. Neben dem Schlüsselaustausch mit Primzahlen nach Diffie-Hellman beherrscht der TOE auch das ECDHE-Verfahren, bei dem Punkte auf elliptischen Kurven berechnet und ausgetauscht werden. Für dieses Verfahren werden die Kurven secp256r1, secp384r1 [RFC 8422; ANSI X9.62] und brainpoolP256r1, brainpoolP384r1 [RFC 7027] genutzt. Alle TLS-Parameter sind in den Tabellen in Anhang B zusammengefasst.

Im Rahmen des sicheren Löschen von geheimen Schlüsselmaterial muss die Speicherverwaltung berücksichtig werden. Die Details zur sicheren Schlüssellöschung sind in [ADV_TDS] dokumentiert. (FCS_CKM.4)

A. Zuordnung von SFR zu TSFI

SFR	TSFI	Verwendung
FCS_CKM.1	LS.LAN.TLS LS.WAN.IPSec	Schlüsselaushandlung für TLS Schlüsselaushandlung für VPN
FCS_CKM.2/IKE	LS.WAN.IPSec	Schlüsselverteilung für VPN
FCS_CKM.2/TLS	LS.LAN.TLS	Schlüsselverteilung für TLS
FCS_CKM.4	LS.LAN.TLS LS.WAN.IPSec	TLS Verbindungen im LAN abbauen IPSEC Verbindungen im WAN abbauen
FCS_COP.1/Hash	LS.WAN.IPSec LS.LAN.TLS	IPSec Hash Operationen TLS Hash Operationen
FCS_COP.1/HMAC	LS.WAN.IPSec LS.LAN.TLS	IPSec HMAC Operationen TLS HMAC Operationen
FCS_COP.1/TLS.AES	LS.LAN.TLS	TLS Verbindungen
FCS_COP.1/TLS.Auth	LS.LAN.TLS	TLS Verbindungen
FCS_RNG.1/Hash_DRBG	LS.LAN.TLS LS.WAN.IPSec	TLS Verbindungen Schlüsselaushandlung für VPN
FDP_RIP.1	Keine TSFI	Sicheres Löschen nicht von außen aufrufbar
FPT_TDC.1/TLS.Zert	LS.LAN.TLS	TLS Zertifikate prüfen
FPT_TDC.1/Zert	LS.WAN.IPSec	VPN Zertifikate prüfen
FPT_STM.1	LS.WAN.NTP	Zugang zum Zeitdienst
FPT_TST.1	LS.LAN.HTTP_MGMT	Aufruf des Selbsttests
FTP_ITC.1/TLS	LS.LAN.TLS	Sichere Verbindung zur Managemant-schnittstelle
FTP_ITC.1/VPN	LS.WAN.IPSec	Sicherer IPSec Tunnel
FTP_TRP.1/Admin	LS.LAN.HTTP_MGMT	Verbindung zur Managemantschnittstelle

Tabelle A.1.: Zuordnung von SFR zu TSFI

B. TLS Verbindungen

Für die TLS-Verbindungen werden die im Schutzprofil genannten Cipher Suiten verwendet. Der TOE beherrscht genau diese Cipher Suiten und keine darüber hinaus. Tabelle B.1 listet diese Cipher Suiten auf. Tabelle B.2 zeigt die elliptischen Kurven, die beim ECDHE Schlüsselaustausch zur Anwendung kommen.

Algorithmen / Cipher Suite	IANA ID	TLS 1.2 [RFC 5246]
TLS_DHE_RSA_WITH_AES_128_CBC_SHA	`0x00,0x33`	✓
TLS_DHE_RSA_WITH_AES_256_CBC_SHA	`0x00,0x39`	✓
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA	`0xc0,0x13`	✓
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA	`0xc0,0x14`	✓
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256	`0xc0,0x27`	✓
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384	`0xc0,0x28`	✓
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	`0xc0,0x2f`	✓
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	`0xc0,0x30`	✓

Tabelle B.1.: Cipher Suites der TLS Verbindungen des TOE

Elliptische Kurve	IANA ID	Standard
secp256r1 (P-256)	`23`	[RFC 8422; ANSI X9.62]
secp384r1 (P-384)	`24`	[RFC 8422; ANSI X9.62]
brainpoolP256r1	`26`	[RFC 7027]
brainpoolP384r1	`27`	[RFC 7027]

Tabelle B.2.: Elliptische Kurven für die TLS Verbindungen des TOE

Der TOE kommuniziert mit anderen vertrauenswürdigen IT-Produkten über gesicherte Verbindungen. Die Integrität und die Vertrauenswürdigkeit der Verbindungen wird durch die Verwendung von TLS in der Version 1.2 und die in Tabelle B.1 genannten Algorithmen und Cipher Suiten sichergestellt. Tabelle B.4 listet die Verbindungen auf, die der TOE eingeht. Die Spalten dieser Tabelle werden in Tabelle B.3 beschrieben.

Spalte	Beschreibung
ID	Symbolischer Name der Verbindung
Schnittstelle	Logische Schnittstelle, deren Kommunikation abgesichert wird.
Rolle	Beschreibt, ob der TOE in dieser Verbindung Client oder Server ist.
Peer	Beschreibung des Partners in der TLS-Verbindung
Protokoll	Anwendungsprotokoll, das für die Verbindung genutzt wird.
Subsystem::Modul	Name des Subsystems und des Moduls, von dem die Verbindung ausgeht, bzw. das die Verbindung empfängt und behandelt.
Port	Port, den der TOE öffnet, um die Verbindung aufzubauen. Für Verbindungen, bei denen der TOE Server ist, steht hier eine Portnummer. Wenn der TOE Client ist, steht „dyn.“ für die ephemerische Portvergabe bei TCP-Verbindungen. „konfig.“ steht dafür, dass der Zielport konfigurierbar ist.
Schnittstelle	Logische Schnittstelle des TOE , über die die Verbindung läuft.
Identität des TOE	Zertifikat, mit dem sich der TOE gegenüber dem Peer authentisiert.
Identität des Peer	Zertifikat/Verfahren, mit dem sich der Peer gegenüber dem TOE authentisiert.
Authentifizierung des Peer durch	Verfahren, Datenquelle oder Subsystem/Modul, mit dem der TOE die Identität des Peers verifiziert.

Tabelle B.3.: Legende zu den TLS Verbindungen

ID	Schnittstelle (Protokoll)	Rolle	Peer	Subsystem::Modul	Port	Identität des TOE	Identität des Peer	Authentifizierung des Peer durch
TLS.1	LS.LAN.HTTP_MGMT	Server	Browser	Administrationssystem::HTTP-Server	443	Zertifikat aus Mauve CA	Benutzername/Passwort	Benutzerverwaltung im TOE

Tabelle B.4.: TLS Verbindungen des MauveVPN Client

Literatur

[ADV_ARC] Mauve Corp. *MauveCorp MauveVPN Client. Sicherheitsarchitektur*. Common Criteria Komponente ADV_ARC. Vorgelegt im Verfahren BSI-DSC-CC-xyz zu BSI-CC-PP-00zz.

[ADV_TDS] Mauve Corp. *MauveCorp MauveVPN Client. TOE Design Spezifikation*. Common Criteria Komponente ADV_TDS. Vorgelegt im Verfahren BSI-DSC-CC-xyz zu BSI-CC-PP-00zz.

[AGD_ADM] Mauve Corp. *Administratorhandbuch MauveCorp MauveVPN Client*. Common Criteria Komponente AGD_ADM. Vorgelegt im Verfahren BSI-DSC-CC-xyz zu BSI-CC-PP-00zz.

[ANSI X9.62] Accredited Standards Committee X9. *ANSI X9.62, Public Key Cryptography For The Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA)*. Standard. ANSI, 16. Nov. 2005.

[FIPS PUB 180-4] National Institute of Standards und Technology. *Secure Hash Standard (SHS)*. Federal Information Processing Standards Publication. Information Technology Laboratory, Aug. 2015. URL: http://dx.doi.org/10.6028/NIST.FIPS.180-4.

[IEEE802.3] „IEEE Standard for Ethernet“. In: *IEEE Std 802.3-2015 (Revision of IEEE Std 802.3-2012)* (März 2016), S. 1–4017. DOI: 10.1109/IEEESTD.2016.7428776.

[NIST SP 800-90A] Elaine Barker und John Kelsey. *Recommendation for Random Number Generation Using Deterministic Random Bit Generators. National Industrial Security Program Operating Manual*. NIST Special Publication. Version Revision 1. National Institute of Standards und Technology, Juni 2015. URL: http://dx.doi.org/10.6028/NIST.SP.800-90Ar1.

[RFC 1812] F. Baker. *Requirements for IP Version 4 Routers*. RFC 1812 (Proposed Standard). RFC. Updated by RFCs 2644, 6633. Fremont, CA, USA: RFC Editor, Juni 1995. DOI: 10.17487/RFC1812. URL: https://www.rfc-editor.org/rfc/rfc1812.txt.

[RFC 2104] H. Krawczyk, M. Bellare und R. Canetti. *HMAC: Keyed-Hashing for Message Authentication*. RFC 2104 (Informational). RFC. Updated by RFC 6151. Fremont, CA, USA: RFC Editor, Feb. 1997. DOI: 10.17487/RFC2104. URL: https://www.rfc-editor.org/rfc/rfc2104.txt.

[RFC 2131] R. Droms. *Dynamic Host Configuration Protocol*. RFC 2131 (Draft Standard). RFC. Updated by RFCs 3396, 4361, 5494, 6842. Fremont, CA, USA: RFC Editor, März 1997. DOI: 10.17487/RFC2131. URL: https://www.rfc-editor.org/rfc/rfc2131.txt.

[RFC 2132] S. Alexander und R. Droms. *DHCP Options and BOOTP Vendor Extensions*. RFC 2132 (Draft Standard). RFC. Updated by RFCs 3442, 3942, 4361, 4833, 5494. Fremont, CA, USA: RFC Editor, März 1997. DOI: 10.17487/RFC2132. URL: https://www.rfc-editor.org/rfc/rfc2132.txt.

[RFC 2616] R. Fielding u. a. *Hypertext Transfer Protocol – HTTP/1.1*. RFC 2616 (Draft Standard). RFC. Obsoleted by RFCs 7230, 7231, 7232, 7233, 7234, 7235, updated by RFCs 2817, 5785, 6266, 6585. Fremont, CA, USA: RFC Editor, Juni 1999. DOI: 10.17487/RFC2616. URL: https://www.rfc-editor.org/rfc/rfc2616.txt.

[RFC 2644] D. Senie. *Changing the Default for Directed Broadcasts in Routers*. RFC 2644 (Best Current Practice). RFC. Fremont, CA, USA: RFC Editor, Aug. 1999. DOI: 10.17487/RFC2644. URL: https://www.rfc-editor.org/rfc/rfc2644.txt.

[RFC 3526] T. Kivinen und M. Kojo. *More Modular Exponential (MODP) Diffie-Hellman groups for Internet Key Exchange (IKE)*. RFC 3526 (Proposed Standard). RFC. Fremont, CA, USA: RFC Editor, Mai 2003. DOI: 10.17487/RFC3526. URL: https://www.rfc-editor.org/rfc/rfc3526.txt.

[RFC 4301] S. Kent und K. Seo. *Security Architecture for the Internet Protocol*. RFC 4301 (Proposed Standard). RFC. Updated by RFCs 6040, 7619. Fremont, CA, USA: RFC Editor, Dez. 2005. DOI: 10.17487/RFC4301. URL: https://www.rfc-editor.org/rfc/rfc4301.txt.

[RFC 4303] S. Kent. *IP Encapsulating Security Payload (ESP)*. RFC 4303 (Proposed Standard). RFC. Fremont, CA, USA: RFC Editor, Dez. 2005. DOI: 10.17487/RFC4303. URL: https://www.rfc-editor.org/rfc/rfc4303.txt.

[RFC 4306] C. Kaufman. *Internet Key Exchange (IKEv2) Protocol*. RFC 4306 (Proposed Standard). RFC. Obsoleted by RFC 5996, updated by RFC 5282. Fremont, CA, USA: RFC Editor, Dez. 2005. DOI: 10.17487/RFC4306. URL: https://www.rfc-editor.org/rfc/rfc4306.txt.

[RFC 4868] S. Kelly und S. Frankel. *Using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with IPsec*. RFC 4868 (Proposed Standard). RFC. Fremont, CA, USA: RFC Editor, Mai 2007. DOI: 10.17487/RFC4868. URL: https://www.rfc-editor.org/rfc/rfc4868.txt.

[RFC 5246] T. Dierks und E. Rescorla. *The Transport Layer Security (TLS) Protocol Version 1.2*. RFC 5246 (Proposed Standard). RFC. Updated by RFCs 5746, 5878, 6176, 7465, 7507, 7568, 7627, 7685, 7905, 7919. Fremont, CA, USA: RFC Editor, Aug. 2008. DOI: 10.17487/RFC5246. URL: https://www.rfc-editor.org/rfc/rfc5246.txt.

[RFC 5905] D. Mills u. a. *Network Time Protocol Version 4: Protocol and Algorithms Specification*. RFC 5905 (Proposed Standard). RFC. Updated by RFC 7822. Fremont, CA, USA: RFC Editor, Juni 2010. DOI: 10.17487/RFC5905. URL: https://www.rfc-editor.org/rfc/rfc5905.txt.

[RFC 7027] J. Merkle und M. Lochter. *Elliptic Curve Cryptography (ECC) Brainpool Curves for Transport Layer Security (TLS)*. RFC 7027 (Informational). RFC. Fremont, CA, USA: RFC Editor, Okt. 2013. DOI: 10.17487/RFC7027. URL: https://www.rfc-editor.org/rfc/rfc7027.txt.

[RFC 7159] T. Bray. *The JavaScript Object Notation (JSON) Data Interchange Format*. RFC 7159 (Proposed Standard). RFC. Fremont, CA, USA: RFC Editor, März 2014. DOI: 10.17487/RFC7159. URL: https://www.rfc-editor.org/rfc/rfc7159.txt.

[RFC 7296] C. Kaufman u. a. *Internet Key Exchange Protocol Version 2 (IKEv2)*. RFC 7296 (Internet Standard). RFC. Updated by RFCs 7427, 7670. Fremont, CA, USA: RFC Editor, Okt. 2014. DOI: 10.17487/RFC7296. URL: https://www.rfc-editor.org/rfc/rfc7296.txt.

[RFC 768] J. Postel. *User Datagram Protocol*. RFC 768 (Internet Standard). RFC. Fremont, CA, USA: RFC Editor, Aug. 1980. DOI: 10.17487/RFC0768. URL: https://www.rfc-editor.org/rfc/rfc768.txt.

[RFC 791] J. Postel. *Internet Protocol*. RFC 791 (Internet Standard). RFC. Updated by RFCs 1349, 2474, 6864. Fremont, CA, USA: RFC Editor, Sep. 1981. DOI: 10.17487/RFC0791. URL: https://www.rfc-editor.org/rfc/rfc791.txt.

[RFC 792] J. Postel. *Internet Control Message Protocol*. RFC 792 (Internet Standard). RFC. Updated by RFCs 950, 4884, 6633, 6918. Fremont, CA, USA: RFC Editor, Sep. 1981. DOI: 10.17487/RFC0792. URL: https://www.rfc-editor.org/rfc/rfc792.txt.

[RFC 793] J. Postel. *Transmission Control Protocol*. RFC 793 (Internet Standard). RFC. Updated by RFCs 1122, 3168, 6093, 6528. Fremont, CA, USA: RFC Editor, Sep. 1981. DOI: 10.17487/RFC0793. URL: https://www.rfc-editor.org/rfc/rfc793.txt.

[RFC 8017] K. Moriarty (Ed.) u. a. *PKCS #1: RSA Cryptography Specifications Version 2.2*. RFC 8017 (Informational). RFC. Fremont, CA, USA: RFC Editor, Nov. 2016. DOI: 10.17487/RFC8017. URL: https://www.rfc-editor.org/rfc/rfc8017.txt.

[RFC 8422] Y. Nir, S. Josefsson und M. Pegourie-Gonnard. *Elliptic Curve Cryptography (ECC) Cipher Suites for Transport Layer Security (TLS) Versions 1.2 and Earlier*. RFC 8422 (Proposed Standard). RFC. Fremont, CA, USA: RFC Editor, Aug. 2018. DOI: 10.17487/RFC8422. URL: https://www.rfc-editor.org/rfc/rfc8422.txt.