Anvisning Kryptering

inera

Ett företag inom SKR

Syfte

Denna anvisning är en del av Ineras Riktlinje för informationssäkerhet och behandlar tekniska detaljer kring konfigurationen av krypterad kommunikation mha VPN (IPSec) eller TLS och kryptering av lagrade data s.k. filkryptering för tjänster som tillhandahålls av Inera.

Syftet är att beskriva de krav som Inera ställer på kryptering av data,

Berörda

Målgrupper för denna anvisning är alla som arbetar med den tekniska överföringen och lagringen av den data som Inera hanterar.

Beskrivning

Val av krypteringsmetod

Grönt är godkänd nivå för transportkryptering och lagring.

Gult är acceptabel nivå för transportkryptering och lagring men det ska finnas en i tiden rimlig avvecklingsplan.

Rött är en icke acceptabel nivå för transportkryptering och lagring och ska Ej användas utan avvecklas snarast.

Lila är en ny och helt acceptabel nivå för transportkryptering och lagring **men** kan innebära kompatibilitetsproblem från kommunicerande tjänster och klienter som kanske ännu inte implementerat stöd för denna algoritm.

Protokoll för transportkryptering med TLS

Rekommenderade protokoll för transportkryptering med TLS

- **TLS 1.3 –** Version 1.3 är from augusti 2018 godkänd av IETF standard. Många Webbläsare har redan idag, i senaste versioner, stöd för TLS 1.3 men alla kommunicerande tjänster och klienter kanske ännu inte implementerat stöd för detta protokoll.
- **TLS 1.2 –** Aktiverat som standard i de senaste versionerna av alla webbläsare, kan manuellt aktiveras på vissa äldre OS/Webbläsare, men är dock aktiverat som standard i de versioner som stöds inom eKlient i Samverkan
- **TLS 1.1** – Rekommenderas INTE, men kan i enstaka fall behöva aktiveras och bibehållas för bakåtkompatibilitet.

- **TLS 1.0** – Rekommenderas INTE, men kan i enstaka fall behöva aktiveras och bibehållas för bakåtkompatibilitet.
- **SSL 3.0** – Rekommenderas INTE. Vid en ev. användning skall man vara extra noga att klienter och server är patchade mot kända attacker och svagheter. Relativt liten skillnad mot TLS 1.0.
- **SSL 2.0** – Rekommenderas INTE, då den har säkerhetsbrister som inte kan åtgärdas genom val av Cipher Suite och fixar på klienter.

Endast TLS 1.2 och TLS 1.2 ska vara aktiverat. TLS 1.0 och TLS 1.1 och lägre versioner ska avvecklas.

Huvudregeln är alltid att stänga av så många äldre protokoll som möjligt.

Cipher Suites

Cipher Suites är ett samlingsbegrepp för de komponenter som ingår när en krypterad session förhandlas inom IPSec och TLS. En Cipher Suite består av följande delkomponenter.

En komplett lista över godkända och accepterade Cipher Suites för TLS och IPsec finns beskrivet längre ner

OBS Cipher Suites är i alla delar inte applicerbart på lagrade data OBS:

- Key Exchange Algorithm (se rubrik Rekommenderade algoritmer för nyckelutbyte (Key Exchange):
  - RSA med DHE eller ECDHE
  - DHE-DSS
  - ECDH-ECDSA
  - RSA med DH, ECDH eller SRP
  - RSA, PSK
- Encryption Algorithm (se rubrik Krypteringsalgoritmer)
  - AES (AES128-GCM, AES256-GCM)
  - ChaCha20 (AEAD_CHACHA20_POLY1305)
  - AES (AES128-CBC, AES256-CBC), Camellia
  - RC4, Triple DES, IDEA, DES,
- Message Authentication Code (MAC) (se rubrik Informativt om Hashed Message Authentication Code, HMAC kapitel)
  - SHA-3
  - Poly1305
  - SHA-2
  - SHA-1
  - MD5

I praktiken fungerar valet av vilken Cipher Suite som ska användas så här:

1. Klienten eller den som initierar kommunikationen skickar en lista över vilka Cipher Suites den har stöd för i prioriteringsordning
2. Servern/eller mottagande part väljer en av dessa, alternativt nekar anslutningen.

Huvudregeln är att begränsa servern till att bara tillåta ett visst urval av Cipher Suites för att undvika klienter som inte har stöd för eller som vill göra en förhandling som innebär dålig eller icke accepterad säkerhetsnivå.

Protokoll för transportkryptering med IKE/IPsec

IKE, Internet Key Exchange

IKE protokollet finns i två versioner, IKEv1 och IKEv2. IKEv2 protokollet har genomgått ett antal säkerhetsförbättringar som förklaras i "IKEv2 Clarifications and Implementation Guidelines".

Rekommenderat IKE protokoll:

- IKEv2 – Den rekommenderade versionen.
- IKEv1 – Main mode. Kan användas där kompatibilitet krävs mot utrustning som inte stöder IKEv2
- IKE v1 – Aggressive mode. Sårbarheter är identifierade som gör att Aggressive mode ska undvikas

Rekommenderad autentiseringsmetod

Det finns ett flertal autentiseringsmetoder beskrivna för IKEv2 men i huvudsak används tre olika sätt:

1. RSA och DSS, digitala signaturer, certifikatsbaserad autentisering.
2. EAP, Extensible Authentication Protocol, som i första hand används för att autentisera IPsec baserade fjärranklienter mot en autentiseringsserver typ Radius eller AD.
3. Shared key (Preshared key), en, mellan kommunicerande parter, delad nyckel.

Rekommenderad autentiseringsmetod är den certifikatsbaserade med t.ex. SITHS.

OBS. Pre-Shared key ska endast användas i lab- eller tidiga testmiljöer och får inte förekomma i en produktionsmiljö då den relativt enkelt kan delas och användas av obehöriga.

Godkända krypteringsalgoritmer

Krypterings- och nyckelutbytesalgoritmer för IPsec baserad VPN ska följa det som finns i kapitel Parametrar för transportkryptering.

Rekommenderad Key Lifetime

Rekommenderad Key Lifetime för en IKE SA: 8 timmar

IPsec
Rekommenderat IPsec protokoll

IPsec ”Phase 2” tunnlar finns i två protokollvarianter ESP och AH, (se Förkortningar kap 1) och i två olika moder, Transport och Tunnel mode.

ESP - Här krypteras och signeras hela original IP paketet in i ett nytt IP paket.

AH – Endast signering av IPsec paketet dvs. ingen kryptering av original IP paketet (payload).

Tunnel mode – Tillsammans med ESP krypteras hela originalpaketet och den inre IP adressen (läs IP header) döljs med VPN gatewayens externa IP adress.

Transport mode – Originalpaketets IP adress (läs IP header) speglas ut till den yttre IPsec headern.

För att erhålla full konfidentialitet, dvs. att dölja hela trafiken mellan två kommunicerande parter, krävs ESP i Tunnel mode.

Rekommenderade krypteringsalgoritmer

Krypterings- och nyckelutbytesalgoritmer för IPsec baserad VPN ska följa det som finns i kapitel Parametrar för transportkryptering.

Rekommenderad Key Lifetime

Rekommenderad Key Lifetime för en IPsec SA:

1 timma (alternativt kan Key Lifetime sättas utifrån överförd datamängd t.ex. 100 MB).

Best practice från NIST ”Guide to IPsec VPNs” säger att en IPsec SA Key Lifetime aldrig ska överstiga 8 timmar (28 880 sekunder) men bör sättas kortare om stora datamängder skickas över IPsec SA. IPsec Key Lifetime bör kunna sätta baserat på både tid och datamängd.

Kryptering av lagrad information (filkryptering)

Godkända krypteringsalgoritmer
Krypteringsalgoritmer för lagrad information och filer följa relevanta delar av det som finns i kapitel Parametrar för transportkryptering.

Rekommenderade krypteringsalgoritmer för att skydda lagrade data är:

AES 256, AES algoritmen med 256 bitars krypteringsnyckel

AES 128, AES algoritmen med 128 bitars krypteringsnyckel

Skydd av krypteringsnycklar
Hänvisning till Instruktion för nyckelhantering till lagrade krypterade data

Lösenordslängd
Ett lösenord ska vara minst 16 tecken långt med komplexitetskrav.

16 tecken x 8 bitar = 128 bitar (minst, se nedan) som i så fall står i paritet med krypteringsalgoritmens nyckellängd.

Med Unicode-baserad teckentabell representeras varje tecken av en till tre bytes, således blir, beroende på valda tecken, lösenordet med 16 tecken minst 128 bitar.

Parametrar för transportkryptering

Algoritmer för nyckelutbyte (Key Exchange)
Godkända kombinationer av algoritmer för nyckelutbyte listas nedan i prioriteringsordning:

Med stöd för RSA:

ECDHE-RSA, DHE-RSA

Övriga alternativ ställer vissa krav på servercertifikatet.
ECDHE-ECDSA, **DHE-DSS**

ECDH-ECDSA, **DH-DSS, SRP**

NIST (National Institute of Standards and Technology) har tagit fram en standard för signaturer DSS (Digital Signature Standard) DSA (Digital Standard Algorithm)

DH grupper för nyckelutbyte
Godkänt är att man använder dem enligt följande i prioriteringsordning:

Cipher_Suite_P521-bit, DH group 21, Elliptic Curve Groups (ECP groups) algorithm

Cipher_Suite_P384-bit, DH group 20, (ECP groups) algorithm

Cipher_Suite_P256-bit, DH group 19, (ECP groups) algorithm

Cipher_Suite_4096-bit, DH group 16, Modular Exponential (MODP) algorithm

Cipher_Suite_3072-bit, DH group 15, (MODP) algorithm

Cipher_Suite_2048-bit, DH group 14, (MODP) algorithm

Cipher_Suite_1536-bit, DH group 5, (MODP) algorithm

Om elliptiska kurvor (EC) inte kan användas, ska man generera nya egna Diffie-Hellman grupper (MODP) med minst 2048-bits gruppstorlek, se Not.

Not. Om man genererar en ny Diffie-Hellman grupp oavsett serverprogramvara som används. ska minst en 2048-bit grupp genereras. Enklaste sättet är att generera en ny grupp är genom Openssl tool.

"openssl dhparam -out dhparams.pem 2048"

Algoritmer för PFS (ephemerala)

De algoritmer för nyckelutbyte som stöder PFS är de som använder så kalla ephemerala (kortlivade) Diffie-Hellman nycklar som:

ECDHE-RSA, DHE-RSA, för RSA (som bl.a. innefattar SITHS):

ECDHE-ECDSA, DHE-DSS, för DSA/DSS

Krypteringsalgoritmer

Valet av krypteringsalgoritm eller chiffer påverkar till stor del säkerheten i vald Cipher Suite. Man ska använda någon av de symmetriska krypteringsalgoritmerna: AES256, AES128 tillsammans med GCM eller med stor tvekan CBC. GCM är idag den rekommenderade blockkrypteringsformen (Block Cipher mode). För TLS krävs att man använder TLS 1.2 alternativt TLS 1.3.

Motsvande val av krypteringsalgoritm som ovan gäller även för IKE/IPsec.

TLS 1.0/1.1, oavsett ”Block Cipher” mode ska helt undvikas.

Referenslitteratur för val av krypteringsalgoritmer FMV, 188 Scheme Crypto Policy samt BlueKrypt, Cryptographic Key Length Recommendation.

Godkända och accepterade Cipher Suites för TLS protokollen

Presenteras i prioriteringsordning som de ska föredras av servrar för tjänster inom Inera

Cipher Suites i TLS 1.3 (med AEAD)

Även om TLS 1.3 använder samma ”Cipher Suite” rymd som tidigare versioner av TLS så definieras TLS 1.3 annorlunda, dvs. endast det symmetriska kryptot specificeras, och kan **inte** användas för TLS 1.2. På liknande sätt kan ”Cipher Suites” definierade för TLS 1.2 **inte** användas för TLS 1.3.

1. TLS_AES_256_GCM_SHA384
2. TLS_AES_128_GCM_SHA256
3. TLS_AES_128_CCM_SHA256
4. TLS_AES_128_CCM_8_SHA256

5. TLS_CHACHA20_POLY1305_SHA256

Cipher Suites i TLS 1.2 (i prioriteringsordning)

- AEAD_CHACHA20_POLY1305 (GCM)

1. TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256
2. TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256
3. TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256

- AES, SHA-2 och GCM med PFS (RSA/DSA/DSS)

9. TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (SHA-2)
10. TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (SHA-2)
11. TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (SHA-2)
12. TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (SHA-2)
13. TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 (SHA-2)
14. TLS_DHE_DSS_WITH_AES_256_GCM_SHA384 (SHA-2)
15. TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 (SHA-2)
16. TLS_DHE_DSS_WITH_AES_128_GCM_SHA256 (SHA-2)

- AES, SHA-2 och GCM utan PFS (RSA/DSA/DSS)

17. TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384 (SHA-2)
18. TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384 (SHA-2)
19. TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256 (SHA-2)
20. TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256 (SHA-2)
21. TLS_DH_RSA_WITH_AES_256_GCM_SHA384 (SHA-2)
22. TLS_DH_DSS_WITH_AES_256_GCM_SHA384 (SHA-2)
23. TLS_DH_RSA_WITH_AES_128_GCM_SHA256 (SHA-2)
24. TLS_DH_DSS_WITH_AES_128_GCM_SHA256 (SHA-2)

- AES, SHA-2 och CBC med PFS (RSA/DSA/DSS)

25. TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (SHA-2)
26. TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 (SHA-2)
27. TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (SHA-2)
28. TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 (SHA-2)
29. TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 (SHA-2)
30. TLS_DHE_DSS_WITH_AES_256_CBC_SHA256 (SHA-2)
31. TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 (SHA-2)
32. TLS_DHE_DSS_WITH_AES_128_CBC_SHA256 (SHA-2)

- AES, SHA-2 och CBC utan PFS (RSA/DSA/DSS)

33. TLS_ECDH_RSA_WITH_AES_256_CBC_SHA384 (SHA-2)
34. TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384 (SHA-2)
35. TLS_ECDH_RSA_WITH_AES_128_CBC_SHA256 (SHA-2)

36. TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256 (SHA-2)
37. TLS_DH_RSA_WITH_AES_256_CBC_SHA256 (SHA-2)
38. TLS_DH_DSS_WITH_AES_256_CBC_SHA256 (SHA-2)
39. TLS_DH_RSA_WITH_AES_128_CBC_SHA256 (SHA-2)
40. TLS_DH_DSS_WITH_AES_128_CBC_SHA256 (SHA-2)

TLS 1.0 och 1.1

TLS 1.0 och TLS 1.1 är icke acceptabla transportprotokoll.

Alla varianter av TLS 1.0 och 1,1 ska helt undvikas.

Godkända och accepterade Cipher Suites för IKE och IPsec protokollen

Presenteras i den prioriteringsordning som föredras för servrar och tjänster inom Inera.

Cipher Suites i IKE och IPsec

I första hand:

AES256GCM16-PRFSHA384-ECP384, (AES-GCM-256 AEAD, SHA-384 as PRF and ECDH) key exchange with 384-bit key length)

AES128GCM16-PRFSHA256-ECP256, (AES-GCM-128 AEAD, SHA-256 as PRF and ECDH) key exchange with 256-bit key length)

Krypterings-, hashing- och nyckelutbytesalgoritmer

Övriga rekommenderade krypterings-, hashing- och nyckelutbytesalgoritmer grupperade i prioritetsordning.

Encryption Algorithms

Keyword	Description
1. chacha20poly1305	256-bit ChaCha20/Poly1305 with 128-bit ICV
2. aes256gcm8 or aes256gcm64	256-bit AES-GCM with 64-bit ICV
3. aes256gcm12 or aes256gcm96	256-bit AES-GCM with 96-bit ICV
4. aes256gcm16 or aes256gcm128	256-bit AES-GCM with 128-bit ICV
5. aes256ccm8 or aes256ccm64	256-bit AES-CCM with 64-bit ICV
6. aes256ccm12 or aes256ccm96	256-bit AES-CCM with 96-bit ICV
7. aes256ccm16 or aes256ccm128	256-bit AES-CCM with 128-bit ICV
8. aes192gcm8 or aes192gcm64	192-bit AES-GCM with 64-bit ICV
9. aes192gcm12 or aes192gcm96	192-bit AES-GCM with 96-bit ICV
10. aes192gcm16 or aes192gcm128	192-bit AES-GCM with 128-bit ICV

11.	aes192ccm8 or aes192ccm64	192-bit AES-CCM with 64-bit ICV
12.	aes192ccm16 or aes192ccm128	192-bit AES-CCM with 128 bit ICV
13.	aes192ccm12 or aes192ccm96	192-bit AES-CCM with 96-bit ICV
14.	aes128gcm16 or aes128gcm128	128-bit AES-GCM with 128-bit ICV
15.	aes128gcm12 or aes128gcm96	128-bit AES-GCM with 96-bit ICV
16.	aes128gcm8 or aes128gcm64	128-bit AES-GCM with 64-bit ICV
17.	aes128ccm16 or aes128ccm128	128-bit AES-CCM with 128-bit ICV
18.	aes128ccm12 or aes128ccm96	128-bit AES-CCM with 96-bit ICV
19.	aes128ccm8 or aes128ccm64	128-bit AES-CCM with 64-bit ICV
20.	aes128	128-bit AES-CBC
21.	aes192	192-bit AES-CBC
22.	aes256	256-bit AES-CBC
23.	camellia256ccm16 or camellia256ccm128	256-bit Camellia-CCM with 128-bit ICV
24.	camellia256ccm12 or camellia256ccm96	256-bit Camellia-CCM with 96-bit ICV
25.	camellia256ccm8 or camellia256ccm64	256-bit Camellia-CCM with 64-bit ICV
26.	camellia192ccm16 or camellia192ccm128	192-bit Camellia-CCM with 128-bit ICV
27.	camellia192ccm12 or camellia192ccm96	192-bit Camellia-CCM with 96-bit ICV
28.	camellia192ccm8 or camellia192ccm64	192-bit Camellia-CCM with 64-bit ICV
29.	camellia128ccm16 or camellia128ccm128	128-bit Camellia-CCM with 128-bit ICV
30.	camellia128ccm12 or camellia128ccm96	128-bit Camellia-CCM with 96-bit ICV
31.	camellia128ccm8 or camellia128ccm64	128-bit Camellia-CCM with 64-bit ICV
32.	camellia256	256-bit Camellia-CBC
33.	camellia192	192-bit Camellia-CBC
34.	camellia128	128 bit Camellia-CBC

Integrity Algorithms

	Keyword	Description
1.	sha512 or sha2_512	SHA2_512_256 HMAC
2.	sha384 or sha2_384	SHA2_384_192 HMAC
3.	sha256 or sha2_256	SHA2_256_128 HMAC
4.	sha256_96 or sha2_256_96	SHA2_256_96 HMAC
5.	sha1_160	SHA1_160 HMAC

Diffie Hellman Groups

Keyword	DH Group
NIST Elliptic Curve Groups	
1. ecp521	21
2. ecp384	20
3. ecp256	19
4. ecp224	26
5. ecp192	25
Regular Groups	
6. modp8192	18
7. modp6144	17
8. modp4096	16
9. modp3072	15
10. modp2048	14
11. modp1536	5