Federal Office
for Information Security

Technical Guideline TR-02102-3 Cryptographic Mechanisms: Recommendations and Key Lengths

Part 3 – Use of Internet Protocol Security (IPsec) and Internet Key Exchange (IKEv2)

Document history

Table 1: Document history

Version	***Date***	***Description***
2019-01	2019-02-11	Adjustment of the periods of use, recommendation of CCM mode
2020-01	2020-01-31	Adjustment of the periods of use, discontinuation of HMAC-SHA-1
2021-01	2021-03-12	Adjustment of the periods of use
2022-01	2022-01-24	Adjustment of the periods of use
2023-01	2023-01-17	Increase of the security level to 120 bits, adjustment of the periods of use
2024-01	2024-02-29	Adjustment of the periods of use

Federal Office for Information Security
P.O. Box 20 03 63
53133 Bonn
Tel.: +49 22899 9582-0
E-Mail: TR02102@bsi.bund.de
Internet: https://www.bsi.bund.de

Table of Contents

1 Introduction

This Technical Guideline ("TR" stands for "Technische Richtlinie" in German which means "Technical Guideline") provides recommendations for the use of cryptographic mechanisms in the IPsec (short for Internet Protocol Security) and IKE (short for Internet Key Exchange) protocols. It contains only recommendations for version 2 of the IKE protocol (IKEv2). In this TR, no statements about IKEv1 are made. Using the new IKEv2 protocol is generally recommended for new developments. IKEv2 has advantages over IKEv1, which, however, primarily concern the complexity of the protocol and the required bandwidth when establishing a security association (see also below).

IPsec allows the secure transmission of information in IP-based data networks, ensuring particularly the *confidentiality*, *integrity* and *authenticity* of the information transmitted by means of the IP protocol. There are two types of IPsec protocols:

- *Authentication Header (AH)* ensures the integrity as well as the authenticity of the data transmitted by means of the IP protocol. The confidentiality of the data transmitted is not protected.
- In addition to the objectives realised by AH, *Encapsulated Security Payload (ESP)* also ensures the protection of the confidentiality.

The security objectives listed here are achieved by cryptographic security mechanisms. Moreover, IPsec offers further security mechanisms such as the protection against replaying already processed IPsec packets (replay attack). These mechanisms are not taken into account in this TR.

A fundamental concept of IPsec is the *security association (SA)*. This is an IPsec-secured connection between two communication partners incl. the related cryptographic parameters, algorithms, keys and modes of operation for this connection. With the IKEv2 protocol, a SA can be negotiated. The requirements for this must be defined beforehand by a security administrator. IPsec then allows the actual secure transmission of user data on the level of IP packets on the basis of the previously negotiated SA. The term SA exists analogously for IKEv2. IPsec-SAs (Child-SAs) are derived from previously negotiated IKE-SAs.

Note: Even if all recommendations for the use of IKEv2 and IPsec are taken into account, data can leak from a cryptographic system to a considerable extent, e.g. by using side channels (measurement of the timing behaviour, power consumption, data rates etc.) or by the incorrect configuration of the security protocols on the process platforms. Therefore, the developer should identify possible side channels by involving experts in this field and implement corresponding countermeasures. Depending on the application, this also applies to fault attacks.

Note: For the definitions of the cryptographic terms used in this document, please refer to the glossary in [TR-02102-1].

1.1 Specifications and internet standards

The IKEv2 (or IKE) and IPsec protocols were specified in different RFCs. For IKEv2 (or IKE), the RFCs 2409, 4306, 4718, 5282, 5996, 5998, 7296, 7427, and 8247 (replaces RFC 4307) are available. To IPsec, for example, the RFCs 4106, 4301, 4302, 4303, 4308, 4309, 4543, and 8221 (replaces RFCs 7321 and 4835) apply.

This Technical Guideline provides recommendations for the IKEv2 and IPsec protocols and is primarily based on the currently latest protocol versions and RFCs. For implementations, RFC 7296 (previous version RFC 5996) is particularly important, since it includes a comprehensive revision of previous standards as well as clarifications from RFC 4718.

2 Basic information

2.1 IKEv2

The IKE protocol runs between two IP-based communication systems which would like to communicate using encryption via a (possibly) insecure network by means of IPsec. IKE allows the negotiation and, if necessary, renewal (key change) of the key material to be used for this purpose.

The IKE protocol is available in two versions: The first version (IKEv1) was specified in RFC 2409 in 1998. The currently latest version IKEv2 is specified in the three IETF documents RFC 4306, RFC 5996 and RFC 7296. RFC 7296 is a revision of RFC5996 and RFC 4306. The tasks of the IKE protocol can be summarised as follows:

1. Negotiation of the cryptographic algorithms and cryptographic parameters to be used for IKE for the establishment of an encrypted and integrity-protected channel which is to be established between two parties communicating via the IP protocol in an untrusted network communicating parties
2. Establishment of an encrypted and integrity-protected channel using the cryptographic algorithms negotiated in Item 1
3. Mutual authentication of the two parties
4. Negotiation of the cryptographic algorithms, modes of operation, key lengths to be used for IPsec as well as the kind of the IPsec protocol (AH or ESP). This negotiation takes place within the protection of the channel established in Item 2
5. Generation of the IPsec keys for both communication partners by taking into account the algorithms negotiated in Item 4

All communication processes within IKE always consist of a *request* and a *response* message. Taken together, the two messages form an exchange. The two systems or communication partners involved are traditionally referred to in the IKE protocol as *initiator* and *responder*.

With IKEv2, there are the following four types of exchange:

- `IKE_SA_INIT`
- `IKE_AUTH`
- `CREATE_CHILD_SA`
- `INFORMATIONAL`

`IKE_SA_INIT` (steps 1 and 2) and `IKE_AUTH` (steps 3 and 4) are carried out at the beginning of the IKE process. After the successful completion of `IKE_AUTH` IKE security associations (abbreviated IKE-SAs) as well as security associations for the IPsec protocols AH or ESP (Child-SAs, i.e. IPsec-SAs) are available for the two communicating parties. The IKE-SA encompasses the mutual authentication of the initiator and responder as well as the presence of an encrypted and integrity-protected connection between the two of them (steps 1 to 3 completed successfully). A `CREATE_CHILD_SA` exchange is optional and allows, for example, the renewal of the key material of an existing IPsec-SA on the basis of an existing IKE-SA. This means that the steps 4 and 5 are repeated within the sphere of protection of the existing IKE-SA and are carried out after the previously defined lifetime has expired.

Moreover, there are also `INFORMATIONAL` message exchanges for the transmission of error messages and other messages between the initiator and responder. For details, Section 1.4 and Section 1.5 in [RFC 7296] are referred to.

For details on the IKE process, the IETF document [RFC 7296] is referred to.

2.1.1 Key derivation and key generation

The term *key derivation* describes the generation of cryptographic key material both for IKE-SAs and for IPsec-SAs. A major core element of the key derivation in IKE is a Diffie-Hellman key exchange as well as the calculation of the key material with a so-called pseudo random function (PRF).

The calculation of the key material for the IKA-SA takes place after the `IKE_SA_INIT`-exchange and prior to the `IKE_AUTH` exchange. The *first* `IKE_SA_INIT` message contains in the SA payload the following suggestions of the initiator regarding the algorithms to be used:

1. Symmetric encryption algorithm for the encryption of the IKE messages of the `IKE_AUTH` exchange and the optional `CREATE_CHILD_SA` exchange as well as any `INFORMATIONAL` exchange processes
2. Pseudo random function (PRF) for key derivation
3. Algorithm for the protection of the integrity of the IKE messages transmitted afterwards
4. Diffie-Hellman group for the Diffie-Hellman key agreement. A Diffie-Hellman group is either a prime number *p* together with a generator *g* of the cyclic group $(Z_p)^*$ or elliptic curve parameters together with a base point as generator of a subgroup of the point group. Only the standardised identifiers of a DH group are transmitted. Standardised values apply to the identifiers which can be found under "Transform Type 4" at [IANA].

The first `IKE_SA_INIT` message (request) also contains the following:

- A key exchange payload which contains a public Diffie-Hellman key that was generated prior to the transmission using the suggested Diffie-Hellman group and the private Diffie-Hellman key. The recommendations from [TR-02102-1][1] apply to the generation of private Diffie-Hellman keys.
- The so-called nonce value of the initiator. It is generated randomly and unpredictably and may only be used once.

The nonce values `Ni` and `Nr` of the initiator and responder must have a minimum size of 16 bytes and a maximum size of 256 bytes (see [RFC 7296], Section 3.9). After the `IKE_SA_INIT` exchange, both parties (initiator, responder) calculate independently of each other the following values (see Section 2.14 in [RFC 7296]):

- The *Diffie-Hellman shared secret* `g^ir`
- The parameter `SKEYSEED := prf(Ni | Nr, g^ir)`
  The nonce values `Ni` and `Nr` have been transmitted in the `IKE_SA_INIT` message from the initiator to the responder (`Ni` to the responder) and vice versa (`Nr` to the initiator). They are integrated in concatenated form as keys into the PRF calculation. `g^ir` is the shared secret key according to the Diffie-Hellman key agreement. The value `SKEYSEED` has the output length of the pseudo random function used.
- Based on `SKEYSEED`, the nonces `Ni` and `Nr` as well as the SPI values[2], several keys are calculated:
  `prf+(SKEYSEED, Ni| Nr| SPIi| SPIr) = {SK_d | SK_ai | SK_ar | SK_ei | SK_er | SK_pi | SK_pr}`
  `SPIi` and `SPIr` are the unique identifiers of the IKE-SAs to be negotiated, which are created by the initiator and responder respectively.
  According to [RFC 7296], Section 2.13, `prf+` means the iterated application of the pseudo random function agreed upon in order to achieve an adequate output length for the total amount of the keys to be generated. The number of iterations of the PRF request must be calculated in such a way that the sum

[1] In relation to the use of elliptic curves for the key agreement, [RFC 6954], Section 3, is referred to: *"..., the private Diffie-Hellman keys should be selected with the same bit length as the order of the group generated by the base point G and with approximately maximum entropy."*

[2] Siehe Abschnitt 2.6 in [RFC 7296].

of the bit lengths of SK_d, SK_ai, SK_ar, SK_ei, SK_er, SK_pi and SK_pr is reached. These keys have the following meaning:

Table 2: Overview of the most important keys

Key	***Use***
SK_d	Derivation of keys for Child-SAs
SK_ei	Symmetric key for the encryption of all other IKE messages (IKE_AUTH, CREATE_CHILD_SA, INFORMATIONAL) from the initiator to the responder
SK_ai	Key for the protection of the integrity of all other IKE messages (IKE_AUTH, CREATE_CHILD_SA, INFORMATIONAL) from the initiator to the responder
SK_er	Symmetric key for the encryption of all other IKE messages (IKE_AUTH, CREATE_CHILD_SA, INFORMATIONAL) from the responder to the initiator
SK_ar	Key for the protection of the integrity of all other IKE messages (IKE_AUTH, CREATE_CHILD_SA, INFORMATIONAL) from the responder to the initiator
SK_pi	Key for the generation of AUTH payload for the authentication of the initiator at the responder (for the IKE_AUTH exchange). See also Section 2.15 in [RFC 7296].
SK_pr	Key for the generation of AUTH payload for the authentication of the responder at the initiator (for the IKE_AUTH exchange). See also Section 2.15 in [RFC 7296].

The lengths (in bit) of all keys listed above must be chosen in accordance with the mechanisms recommended in Chapter 3 and their respective bit lengths. In particular, the key lengths of SK_d, SK_pi and SK_pr should be chosen according to the PRF function agreed upon.

2.1.2 Lifetime

Both an IKE-SA and an IPsec-SA should only be valid for a limited period of time and renegotiated after this period of time has expired. As an alternative, the transmitted data volume can also be used as criterion for the renegotiation of an IPsec-SA. According to [RFC 4301], Section 4.4.2.1, an IPsec implementation must support both criteria. Indicating binding periods of validity or an upper limit for the data volume is part of a security policy and must be defined by the system administrator. In contrast to the old IKEv1 protocol, the lifetime of SAs can no longer be renegotiated in the case of IKEv2 (see page 37 in [RFC 7296]).

2.1.3 Rekeying

The term “rekeying” refers to the renegotiation of an expired and thus no longer valid security association. This relates to both IKE-SAs and SAs for IPsec. For both cases, the description in [RFC 7296] is referred to.

2.1.4 RNG/randomness

For the generation of random numbers, e.g. for the generation of cryptographic keys, for the generation of signatures and for the generation of nonces, appropriate random number generators must be used.

A random number generator from one of the classes DRG.3, DRG.4, PTG.3 or NTG.1 according to [AIS 20/31] is recommended, see also Chapter 8 in [TR-02102-1].

2.1.5 Perfect Forward Secrecy (PFS)

Perfect Forward Secrecy means that an intercepted connection cannot be decrypted subsequently even if the long-term keys of the communication partners are known.

With the IKE_AUTH exchange, both the key material for the IKE-SA and for a Child-SA is generated. If further Child-SAs are to be negotiated on the basis of the existing IKE-SA, this can be performed optionally according to Section 2.17 in [RFC 7296] by means of a new Diffie-Hellman key exchange. According to Section 1.3.1 in [RFC 7296], the public Diffie-Hellman keys are transmitted between the initiator and the

responder and the shared Diffie-Hellman secret is calculated afterwards on both sides, which is then integrated into the session key calculation according to [RFC 7296], Section 2.17.

Using PFS is recommended in general.

2.2 IPsec

2.2.1 ESP and AH

The security services of the two IPsec protocols ESP and AH were mentioned in Section 1. For a precise description, [RFC 4302] (for AH) and [RFC 4303] (for ESP) are referred to.

2.2.2 Tunnel and transport mode

Both AH and ESP can be used in two modes of operation: *tunnel mode* and *transport mode*. In tunnel mode, the IPsec security mechanisms are applied to the entire IP packet (i.e. header including the layer 4 protocol) and a new IP header is prefixed. This new header contains the addresses of the cryptographic end points (tunnel ends).

In transport mode, however, the IPsec security mechanisms are only applied to the user data of the IP packet and the original IP header is still used. In contrast to the tunnel mode, the addresses of the systems communicating in a secure manner are not hidden. When intercepting on the secure connection, an attacker would thus obtain information on the communication behaviour or on the secure network.

A precise description of the two modes of operation for AH can be found in [RFC 4302] in Section 3.1.1 and Section 3.1.2. [RFC 4303] in Section 3.1.1 and Section 3.1.2 contains the description of the two modes of operation for ESP. The choice for the tunnel or transport mode depends on the respective application (see also Section 4 in [RFC 4301]). In general, however, the tunnel mode should be preferred over the transport mode when using ESP, because, in case of the tunnel mode, there are no hidden channels from the network to be protected into the untrusted network due to the encryption of the entire internal IP packet. In addition, a complete traffic flow analysis is not possible when using ESP in tunnel mode, since the address information of the internal IP header is hidden by the encryption.

2.2.3 SAD and SPD

The *Security Association Database (SAD)* and the *Security Policy Database (SPD)* are two important IPsec databases that are used when processing IPsec packets (see Section 4.4.1 and Section 4.4.2 in [RFC 4301] for details).

The SPD contains rules defining how incoming and outgoing packets are processed by IPsec. All packets (even non-IPsec packets) are processed based on the rules in the SPD. For example, there are rules defining how the connection between two communication partners is protected. The protection itself can then be performed by AH or ESP.

In the SAD, the SAs are managed. For each connection, there is an entry in the SAD, which contains, for example, the key for the security protocol of the connection that has been agreed upon. There are separate entries in the database for AH and ESP.

Note: The SAD and SPD databases must be stored in a secure manner in order to prevent them from being manipulated by attackers.

3 Recommendations

3.1 General remarks

3.1.1 Periods of use

The recommendations in this Technical Guideline have a maximum period of use. The indication of the year means that the respective mechanism can be used until the end of the year stated. If the year is marked with a "+" sign, it is possible to extend the period of use.

3.1.2 Security level

The security level for all cryptographic mechanisms in this Technical Guideline depends on the security level stated in Section 1.1 in [TR-02102-1] and is 120 bits.

3.2 IKEv2

This Section gives recommendations for the following IKE components:

1. Encryption of IKE messages
2. A function for key derivation or key generation
3. Integrity protection of the IKE messages
4. Groups for the Diffie-Hellman key exchange
5. Mechanisms for mutual authentication

3.2.1 Encryption of IKE messages

The recommendations apply to the encryption of messages exchanged in the `IKE_AUTH`, `CREATE_CHILD_SA` and `INFORMATIONAL` exchanges. The following encryption algorithms are recommended for IKE:

Table 3: Recommended encryption algorithms for IKE messages

No.	*Algorithm*	*IANA no.*	*Specification*	*AES key lengths*	*Use up to*
1	`ENCR_AES_CBC`	12	[RFC 7296]	128 / 256	2030+
2	`ENCR_AES_CTR`	13	[RFC 5930]	128 / 256	2030+
3	`ENCR_AES_GCM_16`	20	[RFC 5282] [RFC 8247]	128 / 256	2030+
4	`ENCR_AES_GCM_12`	19	[RFC 5282] [RFC 8247]	128 / 256	2030+
5	`ENCR_AES_CCM_16`	16	[RFC 5282]	128 / 256	2030+
6	`ENCR_AES_CCM_12`	15	[RFC 5282]	128 / 256	2030+

Note: The first two algorithms in Table 3 must be combined with one of the mechanisms for the protection of the integrity listed in Section 3.2.3. The keys for the algorithms in the table above are calculated according to the requirement given in Section 2.1.1. The keys to be applied are `SK_ei` and `SK_er`.

For further information on the GCM and CCM modes of operation, [TR-02102-1], Section 3.1.2, is referred to. If these modes of operation are used, no algorithm for the protection of the integrity of the messages transmitted may be used according to [RFC 5282], Section 8.

3.2.2 Pseudo random functions for key generation

As explained in Section 2.1.1, a pseudo random function (PRF) is used to generate key material. The following PRFs are recommended:

Table 4: Recommended PRFs for the generation of key material

No.	*PRF*	*IANA no.*	*Specification*	*Use up to*
1	PRF_AES128_XCBC	4	[RFC 4434]	2030+
2	PRF_AES128_CMAC	8	[RFC 4615]	2030+
3	PRF_HMAC_SHA2_256	5	[RFC 4868]	2030+
4	PRF_HMAC_SHA2_384	6	[RFC 4868]	2030+
5	PRF_HMAC_SHA2_512	7	[RFC 4868]	2030+

Note: The length of the generated key (output length of the PRF) must have at least the length of the recommended key length of the encryption algorithm used from Table 3. It must be taken into account that the pseudo random function according to Section 2.13 in [RFC 7296] might have to be called several times.

When using function no. 1 or no. 2 from Table 3, the corresponding notes from Section 2.14 in [RFC 7296] must be taken into account.

3.2.3 Protection of the integrity of IKE messages

The following algorithms are recommended for the protection of the integrity of the messages exchanged in the IKE_AUTH, CREATE_CHILD_SA and INFORMATIONAL exchange:

Table 5: Recommended algorithms for the protection of the integrity of IKE messages

No.	*Algorithm*	*IANA no.*	*Specification*	*Use up to*
1	AUTH_AES_XCBC_96	5	[RFC 7296]	2030+
2	AUTH_HMAC_SHA2_256_128	12	[RFC 4868]	2030+
3	AUTH_HMAC_SHA2_384_192	13	[RFC 4868]	2030+
4	AUTH_HMAC_SHA2_512_256	14	[RFC 4868]	2030+

Note: The key length for the algorithms listed in Table 5 must at least correspond to the required key lengths in the respectively stated RFCs.

For new developments, one of the algorithms based on SHA-2 (no. 2-4) in Table 5 is recommended.

3.2.4 Groups for the Diffie-Hellman key exchange

The following groups are recommended for the key exchange with Diffie-Hellman:

Table 6: Recommended groups for the Diffie-Hellman key exchange

No.	*Name*	*IANA no.*	*Specification*	*Use up to*
1	3072-bit MODP Group	15	[RFC 3526]	2030+
2	4096-bit MODP Group	16	[RFC 3526]	2030+
3	256-bit random ECP group	19	[RFC 5903]	2030+
4	384-bit random ECP group	20	[RFC 5903]	2030+
5	521-bit random ECP group	21	[RFC 5903]	2030+
6	brainpoolP256r1	28	[RFC 6954]	2030+
7	brainpoolP384r1	29	[RFC 6954]	2030+
8	brainpoolP512r1	30	[RFC 6954]	2030+

Remark 1: In order to realise the Perfect Forward Secrecy (PFS) property, another Diffie-Hellman key exchange can be carried out in the `CREATE_CHILD_SA` exchange. The recommended elliptic curves and groups are the same as in the table above.

Remark 2: Using Brainpool curves is recommended in general.

Remark 3: Using additional Diffie-Hellman tests (see [RFC 6989]) is recommended. These tests are recommended especially when using elliptic curves; see Section 2.3 in [RFC 6989].

Remark 4: The elliptic curves with the IANA no. 19, 20 and 21 are NIST curves. In Table 6, the IANA identifiers are used. For alternative names of the curves (e.g. from NIST), see Chapter 5 in [RFC 5903].

3.2.5 Authentication methods

The following authentication methods are recommended:

Table 7: Recommended authentication methods

No.	*Authentication method*	*Bit length*	*Hash function*	*IANA no.*	*Specification*	*Use up to*
1	`ECDSA-256 with curve secp256r1`	256	SHA-256	9	[RFC 4754] [RFC 5903]	2030+
2	`ECDSA-384 with curve secp384r1`	384	SHA-384	10	[RFC 4754] [RFC 5903]	2030+
3	`ECDSA-512 with curve secp521r1`	512	SHA-512	11	[RFC 4754] [RFC 5903]	2030+
4	`ECDSA-256 with curve brainpoolP256r1`	256	SHA-256	14	[RFC 7427]	2030+
5	`ECDSA-384 with curve brainpoolP384r1`	384	SHA-384	14	[RFC 7427]	2030+
6	`ECDSA-512 with curve brainpoolP512r1`	512	SHA-512	14	[RFC 7427]	2030+
7	`RSASSA-PSS`	4096	SHA-384	14	[RFC 7427] [RFC 4055]	2030+
8	`ECGDSA-256 with curve brainpoolP256r1`[3]	256	SHA-256	14	[RFC 7427]	2030+
9	`ECGDSA-384 with curve brainpoolP384r1`	384	SHA-384	14	[RFC 7427]	2030+
10	`ECGDSA-512 with curve brainpoolP512r1`	512	SHA-512	14	[RFC 7427]	2030+

Note 1: The algorithms RSA (IANA no. 1) and DSS (IANA no. 3) are only specified in connection with the hash function SHA-1 in [RFC 7296]. SHA-1, however, should generally not be used any more for the generation of signatures due to attacks on its collision resistance properties. See also Remark 4.3 in [TR-02102-1]. Instead, RSASSA should only be used in connection with PSS (see Section 8.1 and Section 9.1 in [RFC 8017]) and a hash function from the SHA-2 family.

Note 2: When creating an ECDSA signature, it must be taken into account that the nonce k is chosen randomly and distributed evenly from the interval $[1, q-1]$, whereas q is the order of the base point of the elliptic curve. The nonce as well as the long-term key must be kept secret and deleted immediately after

[3] For the encoding of the ECGDSA signatures, see Section 5.2.1 in [TR-03111]. For the OIDs of the ECGDSA versions, see Section 5.2.1.2 in [TR-03111]. For the public key format, OID 1.3.36.3.3.2.5 as well as [Teletrust] and Section 4.4 in [ECGDSA] are referred to.

they have been used once. The messages to be signed in IKEv2 are described in Section 2.15 in [RFC 7296]. The signature created is transmitted in the authentication payload.

Note 3: With authentication method no. 14 [RFC 7427], the signature algorithm and the hash function are stored as an ASN.1 object directly prior to the actual signature within the authentication payload. The ASN.1 object contains the OIDs of the methods used.

The Technical Guideline [TR-02103] contains recommendations on X.509 certificates and certification path validation.

3.3 IPsec

In this section, recommendations are given for the IPsec protocols *Encapsulating Security Payload (ESP)* and *Authentication Header (AH)*. Recommendations for the following security objectives are defined:

1. Protection of the confidentiality of the ESP packages by means of encryption
2. Protection of the integrity of the ESP packets
3. Protection of the integrity of the AH packets

3.3.1 Encryption of ESP packets

The recommendations relate to the encryption of the area to be encrypted of the ESP packets. The recommendations do not depend on whether the tunnel or transport mode of ESP is used. For details about the areas to be encrypted, [RFC 4303], Section 3.1.1 and Section 3.1.2, is referred to.

Table 8: Recommended encryption algorithms for ESP packets

No.	*Algorithm*	*IANA no.*	*Specification*	*AES key lengths*	*Use up to*
1	ENCR_AES_CBC	12	[RFC 3602]	128 / 256	2030+
2	ENCR_AES_CTR	13	[RFC 3686]	128 / 256	2030+
3	ENCR_AES_GCM_16	20	[RFC 4106] [RFC 8247]	128 / 256	2030+
4	ENCR_AES_GCM_12	19	[RFC 4106] [RFC 8247]	128 / 256	2030+
5	ENCR_AES_CCM_16	16	[RFC 4309]	128 / 256	2030+
6	ENCR_AES_CCM_12	15	[RFC 4309]	128 / 256	2030+

Note: The first two algorithms in Table 8 must be combined with one of the mechanisms for the protection of the integrity listed in Section 3.3.2. When using the GCM or CCM mode of operation, a separate protection of the integrity of the ESP packets must be omitted.

3.3.2 Protection of the integrity of ESP packets

The following recommendations relate to the protection of the integrity of ESP packets. The recommendations do not depend on whether the tunnel or the transport mode of ESP is used. For details about the areas to be protected within the ESP packet, [RFC 4303], Section 3.1.1 and Section 3.1.2, is referred to.

Table 9: Recommended algorithms for the protection of the integrity of ESP packets

No.	*Algorithm*	*IANA no.*	*Specification*	*Use up to*
1	AUTH_AES_XCBC_96	5	[RFC 3566]	2030+
2	AUTH_AES_CMAC_96	8	[RFC 4494]	2030+
3	AUTH_HMAC_SHA2_256_128	12	[RFC 4868]	2030+

No.	Algorithm	IANA no.	Specification	Use up to
4	AUTH_HMAC_SHA2_384_192	13	[RFC 4868]	2030+
5	AUTH_HMAC_SHA2_512_256	14	[RFC 4868]	2030+

For new developments, one of the algorithms based on SHA-2 (no. 3-5) in Table 9 is recommended.

3.3.3 Protection of the integrity of AH packets

The following recommendations relate to the calculation of the integrity check value (ICV) within the IPsec protocol *Authentication Header (AH)*. The recommendations do not depend on whether the tunnel or the transport mode of AH is used. For details about the areas to be protected within the AH packet, [RFC 4302], Section 3.1.1 and Section 3.1.2, is referred to.

Table 10: Recommended algorithms for the protection of the integrity of AH packets

No.	Algorithm	IANA no.	Specification	Use up to
1	AUTH_AES_XCBC_96	5	[RFC 3566]	2030+
2	AUTH_AES_CMAC_96	8	[RFC 4494]	2030+
3	AUTH_HMAC_SHA2_256_128	12	[RFC 4868]	2030+
4	AUTH_HMAC_SHA2_384_192	13	[RFC 4868]	2030+
5	AUTH_HMAC_SHA2_512_256	14	[RFC 4868]	2030+

For new developments, one of the algorithms based on SHA-2 (no. 3-5) in Table 10 is recommended.

3.4 SA lifetime and rekeying

The SA lifetime should be defined depending on the security requirements of the application. This applies to both IKE-SAs and IPsec-SAs. In ordinary operating scenarios, the IKE-SA lifetime should not exceed 24 h and the IPsec-SA lifetime should not exceed 4 h. For special scenarios, longer SA lifetimes can be used after consultation with an expert.

Bibliography

[**AIS 20/31**] BSI: AIS 20/31 – A proposal for: Functionality classes for random number generators, 2011

[**ECGDSA**] Erwin Hess, Marcus Schafheutle, Pascale Serf (Siemens AG): The Digital Signature Scheme ECGDSA, 2006, URL: https://www.teletrust.de/fileadmin/files/oid/ecgdsa_final.pdf

[**IANA**] Internet Assigned Numbers Authority (IANA): Internet Key Exchange Version 2 (IKEv2) Parameters, URL: https://www.iana.org/assignments/ikev2-parameters/ikev2-parameters.xhtml

[**RFC 2104**] H. Krawczyk, M. Bellare, R. Canetti: RFC 2104, HMAC: Keyed-Hashing for Message Authentication, 1997

[**RFC 2404**] C. Madson, R. Glenn: RFC 2404, The Use of HMAC-SHA-1-96 within ESP and AH, 1998

[**RFC 2409**] D. Harkins, D. Carrel: RFC 2409, The Internet Key Exchange (IKE), 1998

[**RFC 3526**] T. Kivinen, M. Kojo: RFC 3526, More Modular Exponential (MODP) Diffie-Hellman groups for Internet Key Exchange (IKE), 2003

[**RFC 3566**] S. Frankel, H. Herbert: RFC 3566, The AES-XCBC-MAC-96 Algorithm and Its Use With IPsec, 2003

[**RFC 3602**] S. Frankel, R. Glenn, S. Kelly: RFC 3602, The AES-CBC Cipher Algorithm and Its Use with IPsec, 2003

[**RFC 3686**] R. Housley: RFC 3686, Using Advanced Encryption Standard (AES) Counter Mode With IPsec Encapsulating Security Payload (ESP), 2004

[**RFC 4055**] J. Schaad, B. Kaliski, R. Housley: RFC 4055, Additional Algorithms and Identifiers for RSA Cryptography for use in the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, 2005

[**RFC 4106**] J. Viega, D. McGrew: RFC 4106, The Use of Galois/Counter Mode (GCM) in IPsec Encapsulating Security Payload (ESP), 2005

[**RFC 4301**] S. Kent, K. Seo: RFC 4301, Security Architecture for the Internet Protocol, 2005

[**RFC 4302**] S. Kent: RFC 4302, IP Authentication Header, 2005

[**RFC 4303**] S. Kent: RFC 4303, IP Encapsulating Security Payload (ESP), 2005

[**RFC 4306**] C. Kaufman (Ed.): RFC 4306, Internet Key Exchange (IKEv2) Protocol, 2005

[**RFC 4307**] J. Schiller: RFC 4307, Cryptographic Algorithms for Use in the Internet Key Exchange Version 2 (IKEv2), 2005

[**RFC 4308**] P. Hoffman: RFC 4308, Cryptographic Suites for IPsec, 2005

[**RFC 4309**] R. Housley: RFC 4309, Using Advanced Encryption Standard (AES) CCM Mode with IPsec Encapsulating Security Payload (ESP), 2005

[**RFC 4494**] JH. Song, R. Poovendran, J. Lee: RFC 4494, The AES-CMAC-96 Algorithm and Its Use with IPsec, 2006

[**RFC 4543**] D. McGrew, J. Viega: RFC 4543, The Use of Galois Message Authentication Code (GMAC) in IPsec ESP and AH, 2006

[**RFC 4615**] J. Song, R. Poovendran, J. Lee, T. Iwata: RFC 4615, The Advanced Encryption Standard-Cipher-based Message Authentication Code-Pseudo-Random Function-128 (AES-CMAC-PRF-128) Algorithm for the Internet Key Exchange Protocol (IKE), 2006

[**RFC 4718**] P. Eronen, P. Hoffman: RFC 4718, IKEv2 Clarifications and Implementation Guidelines, 2006

[**RFC 4754**] D. Fu, J. Solinas: RFC 4754, IKE and IKEv2 Authentication Using the Elliptic Curve Digital Signature Algorithm (ECDSA), 2007

[**RFC 4835**] V. Manral: RFC 4835, Cryptographic Algorithm Implementation Requirements for Encapsulating Security Payload (ESP) and Authentication Header (AH), 2007

[**RFC 4868**] S. Kelly, S. Frankel: RFC 4868, Using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with IPsec, 2007

[**RFC 5114**] M. Lepinski, S. Kent: RFC 5114, Additional Diffie-Hellman Groups for Use with IETF Standards, 2008

[**RFC 5282**] D. Black, D. McGrew: RFC 5282, Using Authenticated Encryption Algorithms with the Encrypted Payload of the Internet Key Exchange version 2 (IKEv2) Protocol, 2008

[**RFC 5903**] D. Fu, J. Solinas: RFC 5903, Elliptic Curve Groups modulo a Prime (ECP Groups) for IKE and IKEv2, 2010

[**RFC 5930**] S. Shen, Y. Mao, NSS. Murthy: RFC 5930, Using Advanced Encryption Standard Counter Mode (AES-CTR) with the Internet Key Exchange version 02 (IKEv2) Protocol, 2010

[**RFC 5996**] C. Kaufman, P. Hoffman, Y. Nir, P. Eronen: RFC 5996, Internet Key Exchange Protocol Version 2 (IKEv2), 2010

[**RFC 5998**] P. Eronen, H. Tschofenig, Y. Sheffer: RFC 5998, An Extension for EAP-Only Authentication in IKEv2, 2010

[**RFC 6954**] J. Merkle, M. Lochter: RFC 6954, Using the Elliptic Curve Cryptography (ECC) Brainpool Curves for the Internet Key Exchange Protocol Version 2 (IKEv2), 2013

[**RFC 6989**] Y. Sheffer, S. Fluhrer: RFC 6989, Additional Diffie-Hellman Tests for the Internet Key Exchange Protocol Version 2 (IKEv2), 2013

[**RFC 7296**] C. Kaufman, P. Hoffman, Y. Nir, P. Eronen, T. Kivinen: RFC 7296, Internet Key Exchange Protocol Version 2 (IKEv2), 2014

[**RFC 7321**] D. McGrew, P. Hoffman: RFC 7321, Cryptographic Algorithm Implementation Requirements and Usage Guidance for Encapsulating Security Payload (ESP) and Authentication Header (AH), 2014

[**RFC 7427**] T. Kivinen, J. Snyder: RFC 7427, Signature Authentication in the Internet Key Exchange Version 2 (IKEv2), 2015

[**RFC 8017**] K. Moriarty, B. Kaliski, J. Jonsson, A. Rusch: RFC 8017, PKCS #1: RSA Cryptography Specification Version 2.2, 2016

[**RFC 8221**] P. Wouters, D. Migault, J. Mattsson, Y. Nir, T. Kivinen: RFC 8221, Cryptographic Algorithm Implementation Requirements and Usage Guidance for Encapsulating Security Payload (ESP) and Authentication Header (AH), 2017

[**RFC 8247**] Y. Nir, T. Kivinen, P. Wouters, D. Migault: RFC 8247, Algorithm Implementation Requirements and Usage Guidance for the Internet Key Exchange Protocol Version 2 (IKEv2), 2017

[**Teletrust**] Teletrust: OID database. URL: https://www.teletrust.de/fileadmin/docs/projekte/oid/OID-Liste_1_3_36_3_3.pdf

[**TR-02102-1**] BSI: Technische Richtlinie TR-02102-1, Kryptographische Verfahren: Empfehlungen und Schlüssellängen, 2024

[**TR-02103**] BSI: Technische Richtlinie TR-02103, X.509 Zertifikate und Zertifizierungspfadvalidierung, Version 1.0, 2020

[**TR-03111**] BSI: Technical Guideline TR-03111, Elliptic Curve Cryptography, Version 2.10, 2018