Internet Engineering Task Force (IETF)
Request for Comments: 8221
Obsoletes: 7321
Category: Standards Track
ISSN: 2070-1721

P. Wouters
Red Hat
D. Migault
J. Mattsson
Ericsson
Y. Nir
Check Point
T. Kivinen

October 2017

Cryptographic Algorithm Implementation Requirements and Usage Guidance for Encapsulating Security Payload (ESP) and Authentication Header (AH)

Abstract

This document replaces RFC 7321, "Cryptographic Algorithm Implementation Requirements and Usage Guidance for Encapsulating Security Payload (ESP) and Authentication Header (AH)". The goal of this document is to enable ESP and AH to benefit from cryptography that is up to date while making IPsec interoperable.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc8221.

Table of Contents

1. Introduction

The Encapsulating Security Payload (ESP) [RFC4303] and the Authentication Header (AH) [RFC4302] are the mechanisms for applying cryptographic protection to data being sent over an IPsec Security Association (SA) [RFC4301].

This document provides guidance and recommendations so that ESP and AH can be used with cryptographic algorithms that are up to date. The challenge of such documents is making sure that, over time, IPsec implementations can use secure and up-to-date cryptographic algorithms while keeping IPsec interoperable.

1.1. Updating Algorithm Implementation Requirements and Usage Guidance

The field of cryptography evolves continuously: new, stronger algorithms appear, and existing algorithms are found to be less secure than originally thought. Therefore, algorithm implementation requirements and usage guidance need to be updated from time to time to reflect the new reality. The choices for algorithms must be conservative to minimize the risk of algorithm compromise. Algorithms need to be suitable for a wide variety of CPU architectures and device deployments ranging from high-end bulk encryption devices to small, low-power Internet of Things (IoT) devices.

The algorithm implementation requirements and usage guidance may need to change over time to adapt to the changing world. For this reason, the selection of mandatory-to-implement algorithms was removed from the main Internet Key Exchange Protocol Version 2 (IKEv2) specification [RFC7296] and placed in a separate document.

1.2. Updating Algorithm Requirement Levels

The mandatory-to-implement algorithm of tomorrow should already be available in most implementations of AH/ESP by the time it is made mandatory. This document attempts to identify and introduce those algorithms for future mandatory-to-implement status. There is no guarantee that the algorithms in use today may become mandatory in the future. Published algorithms are continuously subjected to cryptographic attack and may become too weak or could become completely broken before this document is updated.

This document only provides recommendations for the mandatory-to-implement algorithms and "too weak" algorithms that are recommended not to be implemented. As a result, any algorithm listed at the IPsec IANA registry that is not mentioned in this document MAY be implemented. It is expected that this document will be updated over

time and future versions will only mention algorithms that have evolved in status. For clarification, when an algorithm has been mentioned in [RFC7321], this document states explicitly the update of the status.

Although this document updates the algorithms to keep the AH/ESP communication secure over time, it also aims at providing recommendations so that AH/ESP implementations remain interoperable. AH/ESP interoperability is addressed by an incremental introduction or deprecation of algorithms. In addition, this document also considers the new use cases for AH/ESP deployment, such as IoT.

It is expected that deprecation of an algorithm be performed gradually. This provides time for various implementations to update their implemented algorithms while remaining interoperable. Unless there are strong security reasons, an algorithm is expected to be downgraded from MUST to MUST- or SHOULD, instead of MUST NOT (see Section 2). Similarly, an algorithm that has not been mentioned as mandatory-to-implement is expected to be introduced with a SHOULD instead of a MUST.

The current trend toward IoT and its adoption of AH/ESP requires this specific use case to be taken into account as well. IoT devices are resource-constrained devices, and their choice of algorithms is motivated by minimizing the footprint of the code, the computation effort, and the size of the messages to send. This document indicates "(IoT)" when a specified algorithm is specifically listed for IoT devices. Requirement levels that are marked as "IoT" apply to IoT devices and to server-side implementations that might presumably need to interoperate with them, including any general-purpose VPN gateways.

1.3. Document Audience

The recommendations of this document mostly target AH/ESP implementers as implementations need to meet both high security expectations as well as high interoperability between various vendors and with different versions. Interoperability requires a smooth move to more secure cipher suites. This may differ from a user point of view that may deploy and configure AH/ESP with only the safest cipher suite.

This document does not give any recommendations for the use of algorithms, it only gives recommendations for implementations. The use of algorithms by a specific user is dictated by their own security policy requirements, which are outside the scope of this document.

The algorithms considered here are listed by IANA as part of the IKEv2 parameters. IKEv1 is out of scope of this document. IKEv1 is deprecated; the recommendations of this document must not be considered for IKEv1, nor may IKEv1 parameters be considered by this document.

The IANA registry for "Internet Key Exchange Version 2 (IKEv2) Parameters" contains some entries that are not for use with ESP or AH. This document does not modify the status of those algorithms.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

We define some additional terms here:

SHOULD+ This term means the same as SHOULD. However, it is likely that an algorithm marked as SHOULD+ will be promoted at some future time to be a MUST.

SHOULD- This term means the same as SHOULD. However, an algorithm marked as SHOULD- may be deprecated to a MAY in a future version of this document.

MUST- This term means the same as MUST. However, we expect at some point that this algorithm will no longer be a MUST in a future document. Although its status will be determined at a later time, it is reasonable to expect that if a future revision of a document alters the status of a MUST- algorithm, it will remain at least a SHOULD or a SHOULD- level.

IoT The Internet of Things.

3. Manual Keying

Manual keying SHOULD NOT be used, as it is inherently dangerous. Without any secure keying protocol, such as IKE, IPsec does not offer Perfect Forward Secrecy (PFS) protection; there is no entity to ensure the refreshing of session keys, the tracking of Security Parameter Index (SPI) uniqueness, and the single use of nonces, Initialization Vectors (IVs), and counters. This document was written for deploying ESP/AH using IKE [RFC7296] and assumes that keying happens using IKEv2 or higher.

If manual keying is used regardless, Counter Mode algorithms such as ENCR_AES_CTR, ENCR_AES_CCM, ENCR_AES_GCM, and ENCR_CHACHA20_POLY1305 MUST NOT be used, as it is incompatible with a secure and persistent handling of the counter (as explained in the Security Considerations section of [RFC3686]). This particularly applies to IoT devices that have no state across reboots. At the time of writing, ENCR_AES_CBC is the only mandatory-to-implement encryption algorithm suitable for manual keying.

4. Encryption Must Be Authenticated

Encryption without authentication is not effective and MUST NOT be used. IPsec offers three ways to provide both encryption and authentication:

- ESP with an Authenticated Encryption with Associated Data (AEAD) cipher
- ESP with a non-AEAD cipher + authentication
- ESP with a non-AEAD cipher + AH with authentication

The fastest and most modern method is to use ESP with a combined mode cipher, such as an AEAD cipher, that handles encryption/decryption and authentication in a single step. In this case, the AEAD cipher is set as the encryption algorithm, and the authentication algorithm is set to none. Examples of this are ENCR_AES_GCM_16 and ENCR_CHACHA20_POLY1305.

A more traditional approach is to use ESP with an encryption and an authentication algorithm. This approach is slower, as the data has to be processed twice: once for encryption/decryption and once for authentication. An example of this is ENCR_AES_CBC combined with AUTH_HMAC_SHA2_512_256.

The last method that can be used is ESP+AH. This method is NOT RECOMMENDED. It is the slowest method and also takes up more octets due to the double header of ESP+AH. This results in a smaller effective MTU for the encrypted data. With this method, ESP is only used for confidentiality without an authentication algorithm, and a second IPsec protocol of type AH is used for authentication. An example of this is ESP with ENCR_AES_CBC with AH with AUTH_HMAC_SHA2_512_256.

5. ESP Encryption Algorithms

Name	Status	AEAD	Comment
ENCR_DES_IV64	MUST NOT	No	UNSPECIFIED
ENCR_DES	MUST NOT	No	[RFC2405]
ENCR_3DES	SHOULD NOT	No	[RFC2451]
ENCR_BLOWFISH	MUST NOT	No	[RFC2451]
ENCR_3IDEA	MUST NOT	No	UNSPECIFIED
ENCR_DES_IV32	MUST NOT	No	UNSPECIFIED
ENCR_NULL	MUST	No	[RFC2410]
ENCR_AES_CBC	MUST	No	[RFC3602][1]
ENCR_AES_CCM_8	SHOULD	Yes	[RFC4309](IoT)
ENCR_AES_GCM_16	MUST	Yes	[RFC4106][1]
ENCR_CHACHA20_POLY1305	SHOULD	Yes	[RFC7634]

[1] - This requirement level is for 128-bit and 256-bit keys. 192-bit keys remain at the MAY level.

(IoT) - This requirement is for interoperability with IoT. Only 128-bit keys are at the given level.

IPsec sessions may have very long lifetime and carry multiple packets, so there is a need to move to 256-bit keys in the long term. For that purpose, the requirement level for 128-bit keys and 256-bit keys is MUST (when applicable). In that sense, the status for 256-bit keys has been raised from MAY in [RFC7321] to MUST.

IANA has allocated codes for cryptographic algorithms that have not been specified by the IETF. Such algorithms are noted as UNSPECIFIED. Usually, the use of these algorithms is limited to specific cases, and the absence of specification makes interoperability difficult for IPsec communications. These algorithms were not mentioned in [RFC7321], and this document clarifies that such algorithms MUST NOT be implemented for IPsec communications.

Similarly, IANA also allocated code points for algorithms that are not expected to be used to secure IPsec communications. Such algorithms are noted as non-IPsec. As a result, these algorithms MUST NOT be implemented.

Various ciphers that are older, not well tested, and never widely implemented have been changed to MUST NOT.

ENCR_3DES status has been downgraded from MAY in [RFC7321] to SHOULD NOT. ENCR_CHACHA20_POLY1305 is a more modern approach and alternative for ENCR_3DES than ENCR_AES_CBC, and so it is expected to be favored to replace ENCR_3DES.

ENCR_BLOWFISH has been downgraded to MUST NOT as it has been deprecated for years by TWOFISH, which is not standardized for ESP and therefore not listed in this document. Some implementations support TWOFISH using a private range number.

ENCR_NULL status was set to MUST in [RFC7321] and remains a MUST to enable the use of ESP with only authentication, which is preferred over AH due to NAT traversal. ENCR_NULL is expected to remain MUST by protocol requirements.

ENCR_AES_CBC status remains at MUST. ENCR_AES_CBC MUST be implemented in order to enable interoperability between implementations that followed [RFC7321]. However, there is a trend for the industry to move to AEAD encryption, and the overhead of ENCR_AES_CBC remains quite large, so it is expected to be replaced by AEAD algorithms in the long term.

ENCR_AES_CCM_8 status was set to MAY in [RFC7321] and has been raised from MAY to SHOULD in order to interact with IoT devices. As this case is not a general use case for VPNs, its status is expected to remain as SHOULD.

ENCR_AES_GCM_16 status has been updated from SHOULD+ to MUST in order to favor the use of authenticated encryption and AEAD algorithms. ENCR_AES_GCM_16 has been widely implemented for ESP due to its increased performance and key longevity compared to ENCR_AES_CBC.

ENCR_CHACHA20_POLY1305 was not ready to be considered at the time of [RFC7321]. It has been recommended by the Crypto Forum Research Group (CFRG) and others as an alternative to AES-CBC and AES-GCM. At the time of writing, there are not enough ESP implementations of ENCR_CHACHA20_POLY1305 to be able to introduce it at the SHOULD+ level. Its status has been set to SHOULD and is expected to become MUST in the long term.

6. ESP and AH Authentication Algorithms

Authentication algorithm recommendations in this section are targeting two types of communications:

- Authenticated-only communications without encryption, such as ESP with NULL encryption or AH communications.
- Communications that are encrypted with a non-AEAD algorithm that MUST be combined with an authentication algorithm.

Name	Status	Comment
AUTH_NONE	MUST / MUST NOT	[RFC7296][RFC5282] AEAD-only
AUTH_HMAC_MD5_96	MUST NOT	[RFC2403][RFC7296]
AUTH_HMAC_SHA1_96	MUST-	[RFC2404][RFC7296]
AUTH_DES_MAC	MUST NOT	UNSPECIFIED
AUTH_KPDK_MD5	MUST NOT	UNSPECIFIED
AUTH_AES_XCBC_96	SHOULD / MAY	[RFC3566][RFC7296] (IoT)
AUTH_AES_128_GMAC	MAY	[RFC4543]
AUTH_AES_256_GMAC	MAY	[RFC4543]
AUTH_HMAC_SHA2_256_128	MUST	[RFC4868]
AUTH_HMAC_SHA2_512_256	SHOULD	[RFC4868]

(IoT) - This requirement is for interoperability with IoT.

AUTH_NONE has been downgraded from MAY in [RFC7321] to MUST NOT. The only case where AUTH_NONE is acceptable is when authenticated encryption algorithms are selected from Section 5. In all other cases, AUTH_NONE MUST NOT be selected. As ESP and AH both provide authentication, one may be tempted to combine these protocols to provide authentication. As mentioned by [RFC7321], it is NOT RECOMMENDED to use ESP with NULL authentication (with non-authenticated encryption) in conjunction with AH; some configurations of this combination of services have been shown to be insecure [PD10]. In addition, AUTH_NONE authentication cannot be combined with ESP NULL encryption.

AUTH_HMAC_MD5_96 and AUTH_KPDK_MD5 were not mentioned in [RFC7321]. As MD5 is known to be vulnerable to collisions, these algorithms MUST NOT be used.

AUTH_HMAC_SHA1_96 has been downgraded from MUST in [RFC7321] to MUST- as there is an industry-wide trend to deprecate its usage.

AUTH_DES_MAC was not mentioned in [RFC7321]. As DES is known to be vulnerable, it MUST NOT be used.

AUTH_AES_XCBC_96 is set as SHOULD only in the scope of IoT, as IoT deployments tend to prefer AES-based Hashed Message Authentication Code (HMAC) functions in order to avoid implementing SHA2. For the wide VPN deployment, as it has not been widely adopted, it has been downgraded from SHOULD to MAY.

AUTH_AES_128_GMAC status has been downgraded from SHOULD+ to MAY. Along with AUTH_AES_192_GMAC and AUTH_AES_256_GMAC, these algorithms should only be used for AH and not for ESP. If using ENCR_NULL, AUTH_HMAC_SHA2_256_128 is recommended for integrity. If using AES-GMAC in ESP without authentication, ENCR_NULL_AUTH_AES_GMAC is recommended. Therefore, these algorithms are kept at MAY.

AUTH_HMAC_SHA2_256_128 was not mentioned in [RFC7321], as no SHA2-based authentication was mentioned. AUTH_HMAC_SHA2_256_128 MUST be implemented in order to replace AUTH_HMAC_SHA1_96. Note that due to a long standing common implementation bug of this algorithm that truncates the hash at 96 bits instead of 128 bits, it is recommended that implementations prefer AUTH_HMAC_SHA2_512_256 over AUTH_HMAC_SHA2_256_128 if they implement AUTH_HMAC_SHA2_512_256.

AUTH_HMAC_SHA2_512_256 SHOULD be implemented as a future replacement of AUTH_HMAC_SHA2_256_128 or when stronger security is required. This value has been preferred to AUTH_HMAC_SHA2_384, as the additional overhead of AUTH_HMAC_SHA2_512 is negligible.

7. ESP and AH Compression Algorithms

Name	Status	Comment
IPCOMP_OUI	MUST NOT	UNSPECIFIED
IPCOMP_DEFLATE	MAY	[RFC3173]
IPCOMP_LZS	MAY	[RFC2395]
IPCOMP_LZJH	MAY	[RFC3051]

Compression was not mentioned in [RFC7321]. As it is not widely deployed, it remains optional and at the MAY level.

8. Summary of Changes from RFC 7321

The following table summarizes the changes from RFC 7321.

Algorithm	RFC 7321	RFC 8221
ENCR_AES_GCM_16	SHOULD+	MUST
ENCR_AES_CCM_8	MAY	SHOULD
ENCR_AES_CTR	MAY	MAY(*)
ENCR_3DES	MAY	SHOULD NOT
AUTH_HMAC_SHA1_96	MUST	MUST-
AUTH_AES_128_GMAC	SHOULD+	MAY
AUTH_NONE	MAY	MUST / MUST NOT

(*) This algorithm is not mentioned in the above sections, so it defaults to MAY.

9. IANA Considerations

This document does not require any IANA actions.

10. Security Considerations

The security of a system that uses cryptography depends on both the strength of the cryptographic algorithms chosen and the strength of the keys used with those algorithms. The security also depends on the engineering and administration of the protocol used by the system to ensure that there are no non-cryptographic ways to bypass the security of the overall system.

This document concerns itself with the selection of cryptographic algorithms for the use of ESP and AH, specifically with the selection of mandatory-to-implement algorithms. The algorithms identified in this document as "MUST implement" or "SHOULD implement" are not known to be broken at the current time, and cryptographic research to date leads us to believe that they will likely remain secure into the foreseeable future. However, this is not necessarily forever. Therefore, we expect that revisions of that document will be issued from time to time to reflect the current best practice in this area.

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <https://www.rfc-editor.org/info/rfc4301>.

[RFC4302] Kent, S., "IP Authentication Header", RFC 4302, DOI 10.17487/RFC4302, December 2005, <https://www.rfc-editor.org/info/rfc4302>.

[RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, DOI 10.17487/RFC4303, December 2005, <https://www.rfc-editor.org/info/rfc4303>.

[RFC7321] McGrew, D. and P. Hoffman, "Cryptographic Algorithm Implementation Requirements and Usage Guidance for Encapsulating Security Payload (ESP) and Authentication Header (AH)", RFC 7321, DOI 10.17487/RFC7321, August 2014, <https://www.rfc-editor.org/info/rfc7321>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

11.2. Informative References

[PD10] Paterson, K. and J. Degabriele, "On the (in)security of IPsec in MAC-then-encrypt configurations", Proceedings of the 17th ACM Conference on Computer and Communications Security (ACM CCS), DOI 10.1145/1866307.1866363, 2010.

[RFC2395] Friend, R. and R. Monsour, "IP Payload Compression Using LZS", RFC 2395, DOI 10.17487/RFC2395, December 1998, <https://www.rfc-editor.org/info/rfc2395>.

[RFC2403] Madson, C. and R. Glenn, "The Use of HMAC-MD5-96 within ESP and AH", RFC 2403, DOI 10.17487/RFC2403, November 1998, <https://www.rfc-editor.org/info/rfc2403>.

[RFC2404] Madson, C. and R. Glenn, "The Use of HMAC-SHA-1-96 within ESP and AH", RFC 2404, DOI 10.17487/RFC2404, November 1998, <https://www.rfc-editor.org/info/rfc2404>.

[RFC2405] Madson, C. and N. Doraswamy, "The ESP DES-CBC Cipher Algorithm With Explicit IV", RFC 2405, DOI 10.17487/RFC2405, November 1998, <https://www.rfc-editor.org/info/rfc2405>.

[RFC2410] Glenn, R. and S. Kent, "The NULL Encryption Algorithm and Its Use With IPsec", RFC 2410, DOI 10.17487/RFC2410, November 1998, <https://www.rfc-editor.org/info/rfc2410>.

[RFC2451] Pereira, R. and R. Adams, "The ESP CBC-Mode Cipher Algorithms", RFC 2451, DOI 10.17487/RFC2451, November 1998, <https://www.rfc-editor.org/info/rfc2451>.

[RFC3051] Heath, J. and J. Border, "IP Payload Compression Using ITU-T V.44 Packet Method", RFC 3051, DOI 10.17487/RFC3051, January 2001, <https://www.rfc-editor.org/info/rfc3051>.

[RFC3173] Shacham, A., Monsour, B., Pereira, R., and M. Thomas, "IP Payload Compression Protocol (IPComp)", RFC 3173, DOI 10.17487/RFC3173, September 2001, <https://www.rfc-editor.org/info/rfc3173>.

[RFC3566] Frankel, S. and H. Herbert, "The AES-XCBC-MAC-96 Algorithm and Its Use With IPsec", RFC 3566, DOI 10.17487/RFC3566, September 2003, <https://www.rfc-editor.org/info/rfc3566>.

[RFC3602] Frankel, S., Glenn, R., and S. Kelly, "The AES-CBC Cipher Algorithm and Its Use with IPsec", RFC 3602, DOI 10.17487/RFC3602, September 2003, <https://www.rfc-editor.org/info/rfc3602>.

[RFC3686] Housley, R., "Using Advanced Encryption Standard (AES) Counter Mode With IPsec Encapsulating Security Payload (ESP)", RFC 3686, DOI 10.17487/RFC3686, January 2004, <https://www.rfc-editor.org/info/rfc3686>.

[RFC4106] Viega, J. and D. McGrew, "The Use of Galois/Counter Mode (GCM) in IPsec Encapsulating Security Payload (ESP)", RFC 4106, DOI 10.17487/RFC4106, June 2005, <https://www.rfc-editor.org/info/rfc4106>.

[RFC4309] Housley, R., "Using Advanced Encryption Standard (AES) CCM Mode with IPsec Encapsulating Security Payload (ESP)", RFC 4309, DOI 10.17487/RFC4309, December 2005, <https://www.rfc-editor.org/info/rfc4309>.

[RFC4543] McGrew, D. and J. Viega, "The Use of Galois Message Authentication Code (GMAC) in IPsec ESP and AH", RFC 4543, DOI 10.17487/RFC4543, May 2006, <https://www.rfc-editor.org/info/rfc4543>.

[RFC4868] Kelly, S. and S. Frankel, "Using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with IPsec", RFC 4868, DOI 10.17487/RFC4868, May 2007, <https://www.rfc-editor.org/info/rfc4868>.

[RFC5282] Black, D. and D. McGrew, "Using Authenticated Encryption Algorithms with the Encrypted Payload of the Internet Key Exchange version 2 (IKEv2) Protocol", RFC 5282, DOI 10.17487/RFC5282, August 2008, <https://www.rfc-editor.org/info/rfc5282>.

[RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T. Kivinen, "Internet Key Exchange Protocol Version 2 (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October 2014, <https://www.rfc-editor.org/info/rfc7296>.

[RFC7634] Nir, Y., "ChaCha20, Poly1305, and Their Use in the Internet Key Exchange Protocol (IKE) and IPsec", RFC 7634, DOI 10.17487/RFC7634, August 2015, <https://www.rfc-editor.org/info/rfc7634>.

Acknowledgements

Some of the wording in this document was adapted from [RFC7321], the document that this one obsoletes, which was written by D. McGrew and P. Hoffman.

Authors' Addresses

Paul Wouters
Red Hat

Email: pwouters@redhat.com

Daniel Migault
Ericsson
8275 Trans Canada Route
Saint-Laurent, QC H4S 0B6
Canada

Phone: +1 514-452-2160
Email: daniel.migault@ericsson.com

John Mattsson
Ericsson AB
SE-164 80 Stockholm
Sweden

Email: john.mattsson@ericsson.com

Yoav Nir
Check Point Software Technologies Ltd.
5 Hasolelim St.
Tel Aviv 6789735
Israel

Email: ynir.ietf@gmail.com

Tero Kivinen

Email: kivinen@iki.fi