ENTSOG AS4 Profile

Draft Version 0 Revision 09 – 2014-07-30

Not for implementation

ENTSOG AISBL; Av. de Cortenbergh 100, 1000-Brussels; Tel: +32 2 894 5100; Fax: +32 2 894 5101; info@entsog.eu, www.entsog.eu, VAT No. BE0822 653 040

Disclaimer

This document only provides specific technical information given for indicative purposes only and, as such, it is subject to further modifications. The information contained in the document is non-exhaustive and non-contractual in nature and subject to the completion of the applicable process foreseen for the approval of the EU Regulation embedding the Network Code on Interoperability and Data Exchange.

No warranty is given by ENTSOG in respect of any information so provided, including its further modifications. ENTSOG shall not be liable for any costs, damages and/or other losses that are suffered or incurred by any third party in consequence of any use of -or reliance on- the information hereby provided.

Table of contents

1 Introduction

The draft Network Code on Interoperability and Data Exchange Rules for European Gas Transmission Networks delivered to the European Commission (EC) on 11 September 2013 specifies that "*AS4 shall be used as common data exchange protocol for document based data exchanges*" [DNCIDX]. This document defines an ENTSOG AS4 Profile that aims to support cross-enterprise collaboration in the gas sector using secure and reliable exchange of business documents based on the AS4 standard [AS4]. This is done by providing an ENTSOG AS4 ebHandler profile and a usage profile for the AS4 communication protocol that allow actors in the gas sector to deploy AS4 communication platforms in a consistent and interoperable way.

The main goals of this profile are to:

- Support exchange of EDIG@S XML documents and other payloads.
- Support business processes of Transmission System Operators for gas, such as Capacity Allocation Mechanism [CAM] and Nomination [NOM], as well as future business processes.
- Leverage experience gained with other B2B protocols in the gas sector, such as AS2 as described in the EASEE-gas implementation guide [EGMTP].
- Provide security guidance based on state-of-the-art best practices, following recommendations for "near term" (defined as "at least ten years") future system use [ENISAAKSP].
- Provide suppliers of AS4-enabled B2B communication solutions with guidance regarding the required AS4 functionality.

This profile adopts document conventions common in technical specifications for Internet protocols and data formats. The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL", "NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

2 AS4 Profile

This specification defines the ENTSOG AS4 profile as the selection of a specific conformance profile of the AS4 standard [AS4], which is profiled further for increased consistency and ease of configuration, and an AS4 Usage Profile that defines how to use a compliant implementation for gas industry document exchange. Section 2.1 describes the AS4 ebHandler Conformance Profile, of which this profile is an extended subset. Section 2.2 describes the feature set that conformant products are REQUIRED to support. Section 2.3 is a usage guide that describes configuration and deployment options for conformant products.

2.1 AS4 and Conformance Profiles

2.1.1 AS4 Standard

This ENTSOG AS4 profile is based on the AS4 Profile of ebMS 3.0 Version 1.0. OASIS Standard [AS4]. AS4 itself is based on other standards, in particular on OASIS ebXML Messaging Services Version 3.0: Part 1, Core Features OASIS Standard [EBMS3], which in turn is based on various Web Services specifications.

The OASIS Technical Committee responsible for maintaining the AS4, ebMS 3.0 Core and other related specifications is tracking and resolving issues in the specifications, which it intends to publish as a consolidated Specification Errata. Implementations of the ENTSOG AS4 Profile SHOULD track resolutions at https://tools.oasis-open.org/issues/browse/EBXMLMSG.

2.1.2 AS4 ebHandler Conformance Profile

The AS4 standard [AS4] defines multiple conformance profiles, which define specific functional subsets of the version 3.0 ebXML Messaging, Core Specification [EBMS3]. A conformance profile corresponds to a class of compliant applications. This version of the ENTSOG AS4 Profile is based on an extended subset of the **AS4 ebHandler Conformance Profile** and a Usage Profile. It aims to support business processes such as Capacity Allocation Mechanism [CAM] and Nomination [NOM], in which documents are to be transmitted securely and reliably to Receivers with a minimal delay.

2.2 ENTSOG AS4 ebHandler Feature Set

The ENTSOG AS4 feature set is, with some exceptions, a subset of the feature set of the AS4 ebHandler Conformance Profile. This section selects specific options in situations where the AS4 ebHandler provides more than one option. This section is addressed to providers of AS4 products and can be used as a checklist of features to be provided in AS4 products. The structure of this chapter mirrors the structure of the ebMS3 Core Specification [EBMS3].

Compared to the AS4 ebHandler Conformance Profile, this profile adds, or updates, some functionality:

- There is an added requirement to support Two Way MEPs (cf. section 2.2.1).

- Transport Layer Security processing, if handled in the AS4 handler, is profiled (cf. section 2.2.6.1).
- Algorithms specified for securing messages at the Message Layer are updated to current guidelines (cf. section 2.2.6.2).
- It also relaxes some requirements:
- Support for **Pull** mode in AS4 will only be REQUIRED when business processes determine that **Pull** mode exchanges are necessary (cf. section 2.2.2).
- All payloads are exchanged in separate MIME parts (cf. section 2.2.3.2).
- Asynchronous reporting of receipts and errors is not REQUIRED (cf. sections 2.2.4, 2.2.5).
- WS-Security support is limited to the X.509 Token Profile (cf. section 2.2.6.2).

2.2.1 Messaging Model

This profile constrains the channel bindings of message exchanges between two AS4 Message Service Handlers (MSHs), one of which acts as Sending MSH and the other as the Receiving MSH. The following diagram (from [EBMS3]) shows the various actors and operations in message exchange:

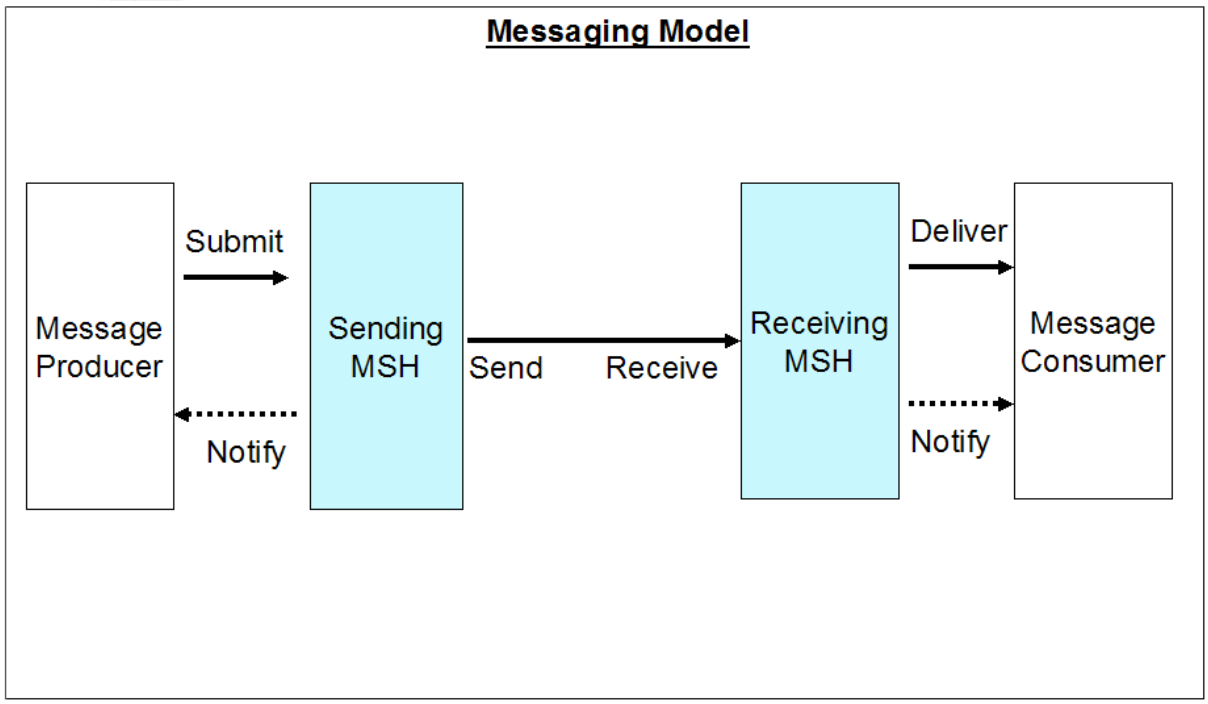

Figure 1 AS4 Messaging Model

Business applications or middleware, acting as *Producer*, *Submit* message content and metadata to the Sending MSH, which packages this content and sends it to the Receiving MSH of the business partner, which in turn *Delivers* the message to another business

application that *Consumes* the message content and metadata. Subject to configuration, Sending and Receiving MSH may *Notify Producer* or *Consumer* of particular events. Note that there is a difference between *Sender* and *Initiator*. For **Push** exchanges, the Sending MSH initiates the transmission of the message. For **Pull** exchanges, the transmission is initiated by the Receiving MSH.

The AS4 ebHandler Conformance Profile is the AS4 conformance profile that provides support for Sending and Receiving roles using **Push** channel bindings. Support is REQUIRED for the following Message Exchange Patterns:

- *One Way / Push*
- *Two Way / Push-and-Push*

While the AS4 ebHandler does not require support for the Two-Way MEP, support for this MEP is REQUIRED in this ENTSOG AS4 profile (see section 2.3.1.3). A message handler that supports Two Way MEPs allows the Producer submitting a message unit to set the optional *RefToMessageId* element in the *MessageInfo* section.

For **PMode.MEP**, support is therefore REQUIRED for the following values:

- http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/oneWay
- http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/twoWay

For **PMode.MEPbinding,** support is REQUIRED for:

- http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/push

Note that these values are identifiers and do not resolve to content on the OASIS site.

2.2.2 Message Pulling and Partitioning

Business processes currently under consideration for this version of this profile are time-critical and considered only supported by the **Push** channel binding, because it allows the *Sender* to control the timing of transmission of the message. Future versions of this profile MAY also support business processes with less time-critical timing requirements. These future uses could benefit from the ebMS3 **Pull** feature. For **PMode.MEPbinding,** applications SHOULD therefore also support:

- http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/pull

This allows implementations of this profile to also support the following Message Exchange Patterns:

- *One Way / Pull*
- *Two Way / Push-and-Pull*
- *Two Way / Pull-and-Push*
- *Two Way / Pull-and-Pull*

Note that any compliant AS4 ebHandler is REQUIRED to support the first of these options. That requirement is relaxed in this profile. The other three options combine Two Way exchanges (see section 2.2.1) with the **Pull** feature.

2.2.3 Message Packaging

The AS4 message structure (see Figure 2) provides a standard message header that addresses B2B requirements and offers a flexible packaging mechanism based on SOAP and MIME enveloping. Dashed line style is used for optional message components.

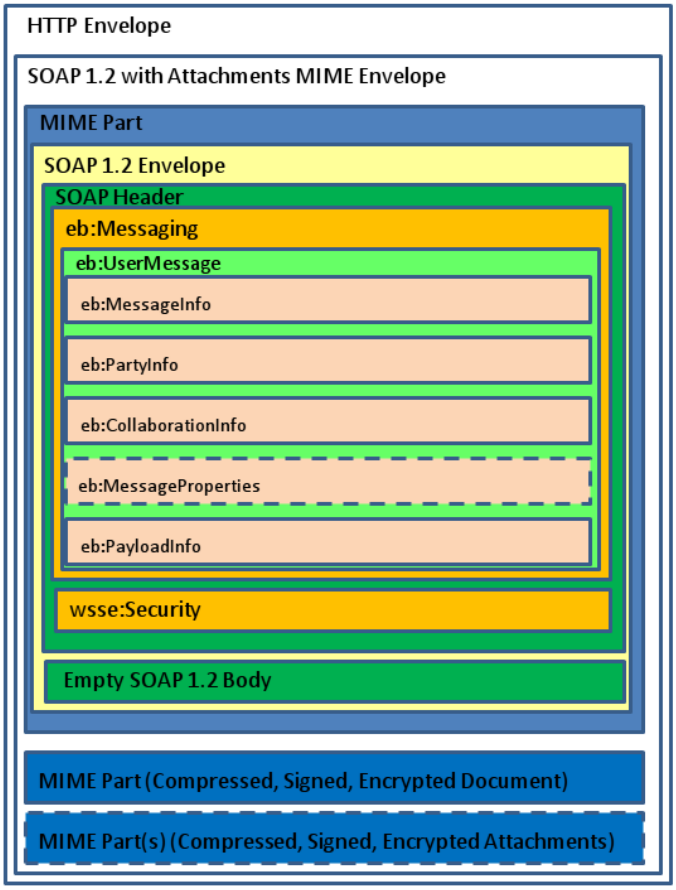

Figure 2 AS4 Message Structure

2.2.3.1 UserMessage

AS4 defines the ebMS3 *Messaging* SOAP header, which envelopes *UserMessage* XML structures, which provide business metadata to exchanged payloads. In AS4, ebMS3 messages other than receipts or errors carry a single *UserMessage*. The ENTSOG AS4 profile follows the AS4 ebHandler Conformance Profile in requiring full configurability for “General” and “BusinessInfo” P-Mode parameters as per sections 2.1.3.1 and 2.1.3.3 of [AS4].

A compliant product MUST allow the Producer, when submitting messages, to set values for *MessageId*, *RefToMessageId* and *ConversationId,* to support correlation (see section 2.3.1.3).

As in the AS4 ebHandler profile, support for **MessageProperties** is REQUIRED in this profile.

2.2.3.2 Payloads

Section 5.1.1 of the ebMS3 Core Specification [EBMS3] requires implementations to process both non-multipart (simple SOAP) messages and multipart (SOAP-with-attachments) messages, and this is a requirement for the AS4 ebHandler Conformance Profile. Due to the mandatory use of AS4 compression in this profile (see section 2.2.3.3), XML payloads are converted to binary data, which is carried in separate MIME parts and not in the SOAP Body. Therefore, AS4 messages based on this profile always have an empty SOAP Body.

The ebMS3 mechanism of supporting "external" payloads via hyperlink references (as mentioned in section 5.2.2.12 of [EBMS3]) MUST NOT be used.

2.2.3.3 Message Compression

The AS4 specification defines payload compression as one of its additional features. Payload compression is a useful feature for many content types, including XML content.

- The parameter **PMode[1].PayloadService.CompressionType** MUST be set to the value *application/gzip* (Note that GZIP is the only compression type currently supported in AS4).

Mandatory use of compression is consistent with current practices for gas B2B data exchange, such as the EASEE-gas AS2 profile [EGMTP]. Compressed payloads are in separate MIME parts.

2.2.4 Error Handling

This profile specifies that errors MUST be reported and transmitted synchronously to the Sender and SHOULD be reported to the Consumer.

- The parameter **PMode[1].ErrorHandling.Report.AsResponse** MUST be set to the value *true*.
- The parameter **PMode[1].ErrorHandling.Report.ProcessErrorNotifyConsumer** SHOULD be set to the value *true*.

2.2.5 Reliable Messaging and Reception Awareness

This profile specifies that non-repudiation receipts MUST be sent synchronously for each message type.

- The parameter **PMode[1].Security.SendReceipt.NonRepudiation** MUST be set to the value *true*.
- The parameter **PMode[1].Security.SendReceipt.ReplyPattern** MUST be set to the value *Response*.

This profile requires the use of the AS4 Reception Awareness feature. This feature provides a built-in *Retry* mechanism that can help overcome temporary network or other issues and detection of message duplicates.

- The parameter **PMode[1].ReceptionAwareness** MUST be set to *true*.

- The parameter **PMode[1].ReceptionAwareness.Retry** MUST be set to *true*.
- The parameter **PMode[1].ReceptionAwareness.DuplicateDetection** MUST be set to *true*.

The parameters **PMode[1].ReceptionAwareness.Retry.Parameters** and related **PMode[1].ReceptionAwareness.DuplicateDetection.Parameters** are sets of parameters configuring retries and duplicate detection. These parameters are not fully specified in [AS4] and implementation-dependent. Products MUST support configuration of parameters for retries and duplicate detection.

Reception awareness errors generated by the Sender MUST be reported to the Submitting application:

- The parameter **PMode[1].ErrorHandling.Report.MissingReceiptNotifyProducer** MUST be set to *true*.
- The parameter **PMode[1].ErrorHandling.Report.SenderErrorsTo** MUST NOT be set. There is no support for reporting sender errors to a third party.

2.2.6 Security

AS4 message exchanges can be secured at multiple communication layers: the network layer, the transport layer, the message layer and the payload layer. The first and last of these are not normally handled by B2B communication software and therefore out of scope for this section. Transport layer security is addressed, even though its functionality MAY be offloaded to another infrastructure component.

This section provides parameter settings based on multiple published sets of best practices. It is noted that after publication of this document, vulnerabilities may be discovered in the security algorithms, formats and exchange protocols specified in this section. Such discoveries SHOULD lead to revisions to this specification.

2.2.6.1 Transport Layer Security

When using AS4, Transport Layer Security (TLS) is an option to provide message confidentiality and authentication. Server authentication, using a server certificate, allows the client to make sure the HTTPS connection is set up with the right server.

- When a message is pushed, the Sender authenticates Recipient's server to which the message is pushed.
- When a message is pulled, the Receiver authenticates Sender's server from which the message is pulled.

Guidance on the use of Transport Layer Security is published in the ENISA Algorithms, Key Sizes and Parameters Report 2013 [ENISAAKSP] and in a Mindest-standard of the Bundesamt für Sicherheit in der Informationstechnik [BSITLS]. If TLS is handled by the AS4 message handler (and not offloaded to some infrastructure component), then:

- It MUST be possible to configure the accepted TLS version(s) in the AS4 message handler. The ENISA and BSI reports state that TLS 1.0 and TLS 1.1 SHOULD NOT be used in new applications. Older version such as SSL 2.0 [RFC6176] and SSL 3.0 MUST NOT be used. Products compliant with this profile MUST therefore support TLS 1.2 [RFC5246].
- It MUST be possible to configure accepted TLS cipher suites in the AS4 message handler. IANA publishes a list of TLS cipher suites [TLSSP], only a subset of which the ENISA Report considers future-proof (see [ENISAAKSP], section 5.1.2). Products MUST support cipher suites included in this subset. Vendors MUST add support for newer, safer cipher suites, as and when such suites are published by IANA/IETF.
- Support for SSL 3.0 and for cipher suites that are not currently considered secure SHOULD be disabled by default.
- Perfect Forward Secrecy, which is REQUIRED in [BSITLS], is supported by the TLS_ECDHE_* and TLS_DHE_* cipher suites, which SHOULD be supported.

If TLS is not handled by the AS4 message handler, but by another component, these requirements are to be addressed by that component (see section 2.3.2.2).

Transport Layer client authentication authenticates the Sender (when used with the Push MEP binding) or Receiver (when used with Pull). Since this profile uses WS-Security for message authentication (see section 2.2.6.2), the use of client authentication at the Transport Layer can be considered redundant. Whether or not client authentication is to be used depends on the deployment environment (see section 2.3.2.2). To support deployments that do require client authentication, products MUST allow Transport Layer client authentication to be configured for an AS4 HTTPS endpoint.

2.2.6.2 Message Layer Security

To provide message layer protection for AS4 messages, this profile REQUIRES the use of the following Web Services Security version 1.1.1 OASIS Standards, profiled in ebMS3.0 [EBMS3] and AS4 [AS4]:

- Web Services Security SOAP Message Security [WSSSMS].
- Web Services Security X.509 Certificate Token Profile [WSSX509].
- Web Services Security SOAP Message with Attachments (SwA) Profile [WSSSWA].

The X.509 Certificate Token Profile supports signing and encryption of AS4 messages. This profile REQUIRES the use of X.509 tokens for message signing and encryption, for all AS4 exchanges. This is consistent with current practice in the gas sector, as specified in the EASEE-gas AS2 profile [EGMTP]. The AS4 option of using Username Tokens, which is supported in the AS4 ebHandler Conformance Profile, MUST NOT be used.

AS4 message signing is based on the W3C XML Signature recommendation. AS4 can be configured to use specific digest and signature algorithms based on identifiers defined in this recommendation. At the time of publication of the AS4 standard [AS4], the current version

of W3C XML Signature was the June 2008, XML Signature, Second Edition specification [XMLDSIG]. The current version is the April 2013, Version 1.1 specification [XMLDSIG1], which defines important new algorithm identifiers, including identifiers for SHA2, and deprecates SHA1, in line with guidance from ENISA [ENISAAKSP].

This ENTSOG AS4 profile uses the following AS4 parameters and values:

- The **PMode[1].Security.X509.Sign** parameter MUST be set in accordance with section 5.1.4 and 5.1.5 of [AS4].
- The **PMode[1].Security.X509.Signature.HashFunction** parameter MUST be set to *http://www.w3.org/2001/04/xmlenc#sha256*.
- The **PMode[1].Security.X509.Signature.Algorithm** parameter MUST be set to *http://www.w3.org/2001/04/xmldsig-more#rsa-sha256*.

This anticipates an update to the AS4 specification to reference this newer specification that has been identified as part of the OASIS AS4 maintenance work.

For encryption, WS-Security leverages the W3C XML Encryption recommendation. The following AS4 configuration options configure this feature:

- The **PMode[1].Security. X509.Encryption.Encrypt** parameter MUST be set in accordance with section 5.1.6 and 5.1.7 of [AS4].
- The parameter **PMode[1].Security.X509.Encryption.Algorithm** MUST be set to *http://www.w3.org/2009/xmlenc11#aes128-gcm*. This is the algorithm used as value for the *Algorithm* attribute of *xenc:EncryptionMethod* on *xenc:EncryptedData*.

AS4 also references an older version of XML Encryption than the current one ([XMLENC] instead of [XMLENC1]). However, the AES 128 algorithm [AES] was already referenced in that earlier version. AES is fully consistent with current recommendations for “near term” future system use [ENISAAKSP]. However, the newer W3C specification recommends AES GCM strongly over any CBC block encryption algorithms.

Key Transport algorithms are public key encryption algorithms especially specified for encrypting and decrypting keys, such as symmetric keys used for encryption of message content. No parameter is defined to support configuration of key transport in [EBMS3]. Implementations are RECOMMENDED to support the following algorithms:

- For encryption method algorithm, *http://www.w3.org/2009/xmlenc11#rsa-oaep*. This is the algorithm used as value for the *Algorithm* attribute of *xenc:EncryptionMethod* on *xenc:EncryptedKey*.
- As mask generation function, *http://www.w3.org/2009/xmlenc11#mgf1sha256*. This is the algorithm used as value for the *Algorithm* attribute of *xenc:MGF* in *xenc:EncryptionMethod*.
- As digest generation function, *http://www.w3.org/2001/04/xmlenc#sha256*. This is the algorithm used as value for the *Algorithm* attribute on *ds:DigestMethod* in *xenc:EncryptionMethod*.

2.2.7 Networking

AS4 communication products compliant with this profile MUST support both IPv4 and IPv6 and MUST be able to connect using either IP4 or IPv6. To support transition from IPv4 to IPv6, products SHOULD support the "happy eyeballs" requirements defined in [RFC6555].

2.3 Usage Profile

This section contains implementation guidelines that specify how products that comply with the requirements of the ENTSOG AS4 ebHandler (section 2.2) SHOULD be configured and deployed. This is similar to the concept of Usage Agreements in section 5 of [AS4] as it does not constrain how AS4 products are implemented, but rather how they are configured and used. The audience for this section are operators/administrators of AS4 products and B2B integration project teams. The structure of this chapter also partly mirrors the structure of [EBMS3], and furthermore covers some aspects outside core pure B2B messaging functionality.

2.3.1 Message Packaging

This usage profile constrains values for several elements in the AS4 message header.

2.3.1.1 Party Identification

When exchanging messages in compliance with this profile, parties registered in the ENTSOG Energy Identification Coding Scheme (EIC) for natural gas transmission MUST be identified using the appropriate EIC Code [EIC]. Entities that do not have an EIC code and need to use this profile MUST contact ENTSOG and request an EIC code. This value MUST be used as the content for the **PMode.Initiator.Party** and **PMode.Responder.Party** processing mode parameters, which AS4 message handlers use to populate the **UserMessage/PartyInfo/{From|to}/PartyId** elements.

Note that AS4 party identifiers identify the communication partner. The communication partner may be:

1. The entity involved in the business transaction
2. A third party providing B2B communication services for other entities.

In case the second case, there are two options for setting the P-Mode parameters:

1. The communication partner may *impersonate* the business entity. In this case the AS4 **Party** identifier is the identifier of the business entity.
2. The business entity may explicitly *delegate* message processing to the communication partner. In this case the AS4 **Party** identifier is the identifier of the communication partner.

Parties MAY use third party communication providers for AS4 communication. Such providers MAY use either the impersonation or delegation model, subject to approval by the business transaction partner.

The AS4 processing layer will validate the identifiers of Sender and Receiver specified in the ebMS3 headers against P-Mode configurations. This involves the validation of message signatures against configured X.509 certificates. The exchanged payloads (EDIG@S or other) typically also reference sending and receiving entities. The responsibility of determining the validity of implied delegation relations between business document layer entities and entities at the AS4 layer is not in scope for the AS4 message handler, but SHOULD be addressed in business applications or integration middleware.

In AS4, it is possible to qualify the Party identifier value using a Party *type* attribute. EIC code values are sufficiently distinct from other codes to not require disambiguation, and this profile does not support other identifier types. Therefore, the *type* attribute MUST NOT be used.

2.3.1.2 Business Process Alignment

The **Service** and **Action** header elements in the **UserMessage/ CollaborationInfo** group relate a message to the business process the message relates to and the roles that sender and receiver perform. This profile is intended to be used with business processes that are currently being modelled by ENTSOG as well as future, possibly not yet identified, business processes. For current and future business processes, ENTSOG will maintain and publish an enumeration of **Service** and **Action** value pairs on its public Web site.

The values for the **UserMessage/PartyInfo/ {From|to}/Role** elements MUST be set to values specified in the Business Process Requirements specification. ENTSOG will also provide and publish values for these elements in BRSs that are to be implemented.

2.3.1.3 Message Correlation

AS4 provides multiple mechanisms to correlate messages within a particular flow.

1. **UserMessage/MessageInfo/RefToMessageId** provides a way to express that a message is a response to a single specific previous message. Presence of a **RefToMessageId** is REQUIRED in response messages in Two Way message exchanges. Whether two exchanges in a business process are modelled as a Two Way exchange or as two One Way exchanges is a decision made in the Business Requirements Specification for the business process. By default, exchanges are considered One Way.
2. **UserMessage/CollaborationInfo/ConversationId** provides a more general way to associate a message with an ongoing conversation, without requiring a message to be a response to a single specific previous message, but allowing update messages to existing conversations from both Sender and Receiver of the original message.

The ebMS3 and AS4 specifications do not constrain the use of these elements. The use of **RefToMessageId** and **ConversationId** shall be defined in the Business Requirements Specification of the concerned business process context. If no explicit rules are defined, the following rule shall apply:

1. **UserMessage/MessageInfo/RefToMessageId** MUST NOT be used. The default exchange is the One Way exchange.
2. **UserMessage/CollaborationInfo/ ConversationId** MUST be included in any AS4 message (as it is a mandatory element) with as content the empty string.

2.3.2 Security

This section describes configuration and deployment considerations in the area of security.

2.3.2.1 Network Layer Security

This profile is intended to support exchange of AS4 messages using either the public Internet or private data networks for communication. When using the public Internet, each organization is individually responsible to implement security measures to protect access to its IT infrastructure. Data exchange may use IPv4 or IPv6.

Organizations SHOULD use firewalls to restrict incoming or outgoing message flows to specific IP addresses, or address ranges. This prevents unauthorized hosts from connecting to the AS4 communication server Organizations therefore:

- MUST use static IP addresses (or IP address ranges) for inbound and outbound AS4 HTTPS connections.
- MUST communicate all IP addresses (or IP address ranges) used for outgoing and incoming connections to their trading partners, also covering addresses of any passive nodes in active-passive clusters. Note that the address of the HTTPS endpoint which an AS4 server is to push messages to or pull messages from MAY differ from the address (or addresses) used for outbound connections.
- MUST notify their trading partners about any IP address changes sufficiently in advance to allow firewall and other configuration changes to be applied.

2.3.2.2 Transport Layer Security

The Transport Layer Security settings defined in section 2.2.6.1 MAY be implemented in the AS4 communication server but TLS MAY also be offloaded to a separate infrastructure component (such as a firewall, proxy server or router). In that case, the recommendations on TLS version and cipher suites of 2.2.6.1 MUST be addressed by that component.

The X.509 certificate used by such a separate component MAY follow the requirements of section 2.3.2.4, but this is NOT REQUIRED.

The TLS cipher suites recommended in section 2.2.6.1 are supported in recent versions of TLS toolkits and which therefore are available for use. Support for these suites is RECOMMENDED. Whether or not less secure cipher suites (which are only recommended for legacy applications) are allowed is a local policy decision.

This profile does NOT REQUIRE the use of client authentication. Client authentication MAY be a requirement in the networking policy of individual organizations that the AS4 deployment needs to meet, but is NOT RECOMMENDED.

2.3.2.3 Message Layer Security

The following parameters control configuration of security at the message layer:

- The **PMode[1].Security.X509.Signature.Certificate** parameter MUST be set to a value matching the requirements specified in section 2.3.2.4.
- The **PMode[1].Security.X509.Encryption.Certificate** parameter MUST be set to a value matching the requirements specified in section 2.3.2.4.

2.3.2.4 Certificates and Public Key Infrastructure

In this Usage Profile, X.509 certificates are used to secure both Transport Layer and Message Layer communication. An overview of relevant European and other standards and best practices is provided in [ENTCREQ]. Requirements on certificates can be sub-divided into three groups:

- General requirements;
- Requirements for Transport Layer Security;
- Requirements for Message Layer Security.

The following general requirements apply to all certificates:

- A three year validity period for certificates is RECOMMENDED.
- Guidance on size for RSA public keys for future system use indicates a key size of 2048 bits [BSIALG] or even 3072 bits [ENISAAKSP] is appropriate. Keys with size less than 2048 bits MUST NOT be used.
- The signature algorithm used to sign public keys MUST be based on at least the SHA-256 hashing algorithm.

The following additional requirements apply for certificates for Transport Layer Security:

- TLS server certificates for use in production environments MUST be issued by a Certification Autority (CA). This CA SHOULD meet the requirements specified in [EN 319 411-1].
- No additional requirements are placed on TLS client certificates.

The following additional requirements apply for certificates for Message Layer Security:

- The Message Layer Security certificates for use in production environments MUST be issued by a Certification Authority (CA).
- Organizations MAY use certificates issued by EASEE-gas.
- Use of certificates issued by another Certification Authority is subject to review by ENTSOG. The issuing CA SHOULD meet the “Normalized” Certificate Policy requirements specified in [EN 319 411-3]. A sample certificate profile is provided in section **Error! Reference source not found.**. It follows the EASEE-gas convention of ncluding the party EIC code (see section 2.3.1.1) as value for the Common Name.

- The type of certificates MUST be certificates for organizations, for which proof of identity is required (often referred to as “Class 2” certificates).

B2B document exchange typically occurs in a community of known entities, where communication between parties and counterparties is secured using pre-agreed certificates. Such an environment is different from open environments, where certificates establish identities for (possibly previously unknown) entitities and Certification Authorities play an essential role to establish trust. Entities MUST proactively notify all communication partners of any updates to certificates used, and in turn MUST process any certificate updates from their communication partners. This concerns both regular renewals of certificates at their expiration dates and replacements for revoked certificates.

Organizations MAY also use Certificate Revocation Lists (CRL) or the Online Certificate Status Protocol (OCSP). Individual companies should assess the potential impact on the availability of the AS4 service when using such mechanisms, as their use may cause a certificate to be revoked automatically and messages to be rejected.

2.3.2.5 Certificate Profile

This section defines a profile for X.509 certificates to secure AS4 communication. This profile is consistent with the EASEE-gas certificate profile. For specific requirements, see [ENISAAKSP].

2.3.2.5.1 Key Size

Entity	Algoritm	Keylength
Root-CA	RSA	Minimum of 4096 bits
Sub-CA	RSA	Minimum of 4096 bits
End-Entities	RSA	Minimum of 2048 bits

2.3.2.5.2 Key Algorithm

Entity	Signing Algoritm	O.I.D.
Root-CA	sha256WithRSAEncryption	1.2.840.113549.1.1.11
Sub-CA	sha256WithRSAEncryption	1.2.840.113549.1.1.11
End-Entities	sha256WithRSAEncryption	1.2.840.113549.1.1.11

2.3.2.5.3 Naming

The following example uses the ENTSOG name as CA. This is only provided as an illustration. ENTSOG does not currently intend to become a Certification Authority.

Entiteit	Example Value	Comments
Root-CA	C=BE	ISO country code (ISO 3166)
	O=ENTSOG	Name of the Organization
	CN=ENTSOG CA	Name of the CA
Sub-CA	C=	ISO country code (ISO 3166)

	O=	Name of the Organization
	OU=	Name of the organisational unit
	CN=	Name of the sub-CA

2.3.2.5.4 Certificate Body

Certificate Component	Example Value	Presence	Comments
Certificate		M	
TBSCertificate		M	
Version	v3	M	X.509 version 3 is usually required.
serialNumber	Unique number	M	A unique CA generated number
Signature		M	The calculated signature (for instance the sha2 value encrypted with RSA key with length 4096)
validity.notBefore	Date	M	The start date of the certificate
validity.notAfter	Date	M	The end date of the certificate, at most 3 years after the start date (for end-entities).
issuer.countryName	BE	M	The country code of the country where the CA resides (ISO 3166)
issuer.organizationName	ENTSOG	M	Example, if ENTSOG is the CA
issuer.commonName	ENTSOG CA	M	Example, if ENTSOG is the CA
subject.countryName	BE	M	ISO country code (ISO 3166)
subject.organisationName	Fluxys	M	Name of member organization
subject.organisationUnit			Not applicable
subject.serialNumber	Unique number	M	A unique CA generated number
subject.commonName	EIC code	M	Preferrably the EIC code. Depends on what the CA allows.
subjectPublicKeyInfo.Algorithm	RsaEncryption	M	The encryption algorith, at least RSA.
subjectPublicKeyInfo.SubjectPublicKey			The public key of the subject.
Extensions		M	
signatureAlgorithm	sha2WithRSAEncryption	M	At least SHA-2 is required. SHA-1 is not allowed.
signatureValue	Signature of ENTSOG CA	M	The digital signature value.

2.3.2.5.5 Extensions Signing and Encryption End Entities

Extension Name	Ref RFC 5280	Sign end entity	Encrypt end entity	TLS Client / Server end entity	Comments

Extension Name	Ref RFC 5280	Sign end entity	Encrypt end entity	TLS Client / Server end entity	Comments
AuthorityKeyIdentifier	4.2.1.1	M	M	M	
keyIdentifier		x	x	X	
authorityCertIssuer		M	M	M	
authorityCertSerialNumber		M	M	M	
SubjectKeyIdentifier	4.2.1.2	M	M	M	
subjectKeyIdentifier		M	M	M	
KeyUsage	4.2.1.3	MC	MC	MC	
digitalSignature		M	x	M	
nonRepudiation		x	x	x	Usually only used for qualified certificates for natuaral persons.
keyEncipherment		x	x	M	
dataEncipherment		x	M	x	
keyAgreement		x	x	M	
keyCertSign		x	x	x	Only for CA root and sub-CA certificates.
cRLSign		x	x	x	Only for CA CRL publishing.
encipherOnly		x	x	x	
decipherOnly		x	x	x	
CertificatePolicies	4.2.1.4	x	x	x	
PolicyMappings	4.2.1.5	x	x	x	
SubjectAltName	4.2.1.6	x	x	x	
otherName					TRUE if applicable.
otherName.type-id					OID = 1.3.6.1.4.1.311.20.2.3 Preferrably the subjectserialnumber followed by ENTSOG serialnumber
IssuerAltName	4.2.1.7	x	x	x	
SubjectDirectoryAttributes	4.2.1.8	x	x	x	
BasicConstraints	4.2.1.9	M	M	M	
CA		False	False	False	Only TRUE in case of a CA root or sub-CA certificate.
PathLenConstraint		x	x	x	

Extension Name	Ref RFC 5280	Sign end entity	Encrypt end entity	TLS Client / Server end entity	Comments
NameConstraints	4.2.1.10	x	x	x	
AuthorityInfoAccess		M	M	M	The URL of the OCSP responder.
PolicyConstraints	4.2.1.11	x	x	x	
ExtKeyUsage	4.2.1.12	x	x	M	See next table.
CRLDistributionPoints	4.2.1.13	x	x	x	The URL of the CRL.
InhibitAnyPolicy	4.2.1.14	x	x	x	
FreshestCRL	4.2.1.15	x	x	x	
privateInternetExtensions	4.2.2	x	x	x	

2.3.2.5.6 Extended Key Usage

Extended Key Usage OID	Ref RFC 5280	TLS Client / Server end entity
id-kp-clientAuth	4.2.1.12	M
id-kp-serverAuth	4.2.1.12	M

2.3.2.5.7 Certificate Lifetime

Entity	Maximum Period	Start Refresh
Root-CA	15 years	2 years before
Sub-CA	10 years	1 year before
End Entities	3 years	6 months before

2.3.3 Message Payload and Flow Profile

A single AS4 UserMessage MUST reference, via the *PayloadInfo* header, a single structured business document and MAY reference one or more other (structured or unstructured) payload parts. The business document is considered the “leading” payload part for business processing. Any payload parts other than the business document are not to be processed in isolation but only as adjuncts to the business document. Business document, attachments and metadata MUST be submitted and delivered as a logical unit. The format of the business document SHOULD be XML, but other datatypes MAY be supported in specific business processes or contexts.

For each business process, the Business Requirement Specification MUST specify the XML schema definition (XSD) that the business document MUST conform to. The mapping from **Service** and **Action** value pairs to XSDs MUST be unique, allowing Receivers to validate XML documents using a specific XML schema.

Some gas data exchanges are traditional batch-scheduled exchanges that can involve very large payloads. The trend in the industry towards service-oriented and event-driven exchanges is leading to more, and more frequent, exchanges, with smaller payloads per exchange. It is expected that the vast majority of payloads will be less than 1 MB in size (prior to compression), with rare exceptions up to 10 MB. The number of messages exchanged over a period, their distribution over time and the peak load/average load ratio, are dependent on business process and other factors. Parties MUST take peak message volumes and maximum message size into account when initially deploying AS4. Parties SHOULD also monitor trends in message traffic for existing processes and anticipate any new business processes being deployed (and the expected increases in message and data volumes), and adjust their deployments accordingly in a timely manner.

In practice, there are limitations on the maximum size of payloads that business partners can accept. These limitations may be caused by capabilities of the AS4 message product, or by constraints of the business application, internal middleware, storage or other software or hardware. When designing business processes and document schemas, and when generating content based on those schemas, these requirements SHOULD be taken into account. In particular, business processes in which large amounts of data are exchanged and the business applications supporting these processes SHOULD be designed such that data can be exchanged as a series of related messages, the payload size of each of which does not exceed 10 MB, rather than as a single message carrying a single large payload that could potentially be much larger.

2.3.4 Test Service

Section 5.2.2 of [EBMS3] defines a server test feature that allows an organization to “Ping” a communication partner. The feature is based on messages with the values of:

- **UserMessage/CollaborationInfo/Service** set to *http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/service*
- **UserMessage/CollaborationInfo/Action** set to *http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/test*.

This feature MUST be supported so that business partners can perform a basic test of the communication configuration (including security at network, transport and message layer, and reliability) in any environment, including the production environment. This functionality MAY be supported as a built-in feature of the AS4 product. If not, a PMode MUST be configured with these values. The AS4 product MUST be configured so that messages with these values are not delivered to any business application.

2.3.5 Environments

B2B data exchange solutions are part of the overall IT service lifecycle, in which different environments are operated (typically in parallel) for development, test, pre-production (in some companies referred to as “acceptance environments” or “QA environments”) and production. Development and test are typically internal environments in which trading partners are simulated using stubs. When exchanging messages between organizations (in

either pre-production or production environments), they must target the appropriate environment. In order to prevent a configuration error from causing non-production messages to be delivered to production environments or vice versa, organizations SHOULD configure processing modes at message handlers so that messages from one type of environment cannot be accepted inadvertently by a different type of environment.

3 Example

The following non-normative example is included to illustrate the structure of an AS4 message conforming to this profile, for a *ChangeCapacity* action invoked on a service *CapacityAllocationMechanism:1* in an environment named Test between a hypothetical transmission system operator 21X-EU-A-X0A0Y-Z and a hypothetical auction office 21X-EU-B-P0Q0R-S. The detailed contents of the *wsse:Security* header is omitted.

```
POST /as4handler HTTP/1.1
Host: receiver.example.com:8893
User-Agent: Turia
Content-Type: multipart/related; start="<bb058f4@sender.example.com>"; boundary= "c5bae1842d1e";
type="application/soap+xml"
Content-Length: 472639

--c5bae1842d1e
Content-Id: <bb058f4@sender.example.com>
Content-Type: application/soap+xml; charset="UTF-8"

<S12:Envelope xmlns:S12="http://www.w3.org/2003/05/soap-envelope"
 xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
 xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
 xmlns:eb3="http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/">
  <S12:Header>
    <eb3:Messaging wsu:Id="_18f85fc2-a956-431e-a80e-09a10364871b">
      <eb3:UserMessage>
         <eb3:MessageInfo>
            <eb3:Timestamp>2014-04-03T14:49:28.886Z</eb3:Timestamp>
            <eb3:MessageId>2014-921@5209999001264.example.com</eb3:MessageId>
         </eb3:MessageInfo>
         <eb3:PartyInfo>
            <eb3:From>
               <eb3:PartyId>21X-EU-A-X0A0Y-Z</eb3:PartyId>
               <eb3:Role>TransmissionSystemOperator</eb3:Role>
            </eb3:From>
            <eb3:To>
               <eb3:PartyId>21X-EU-B-P0Q0R-S</eb3:PartyId>
               <eb3:Role>AuctionOffice</eb3:Role>
            </eb3:To>
         </eb3:PartyInfo>
         <eb3:CollaborationInfo>
            <eb3:Service>CapacityAllocationMechanism:1</eb3:Service>
            <eb3:Action>ChangeCapacity</eb3:Action>
            <eb3:ConversationId>2014-921</eb3:ConversationId>
         </eb3:CollaborationInfo>
         <eb3:PayloadInfo>
          <eb3:PartInfo href="cid:e1d-8821-49191b4ece93@sender.example.com">
            <eb3:PartProperties>
              <eb3:Property name="MimeType">application/xml</eb3:Property>
              <eb3:Property name="CharacterSet">utf-8</eb3:Property>
              <eb3:Property name="CompressionType">application/gzip</eb3:Property>
            </eb3:PartProperties>
          </eb3:PartInfo>
        </eb3:PayloadInfo>
      </eb3:UserMessage>
    </eb3:Messaging>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd"
      xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd">
      <!-- details omitted -->
    </wsse:Security>
  </S12:Header>
  <S12:Body wsu:Id="_b656ef2c-516"/>
</S12:Envelope>

--c5bae1842d1e
Content-Id: <e1d-8821-49191b4ece93@sender.example.com>
Content-Type: application/octet-stream
Content-Transfer-Encoding: binary
```

```
BINARY CIPHER DATA

--c5bae1842d1e--
```

4 Revision History

Revision	Date	Editor	Changes Made
v0r1	2013-10-29	PvdE	First Draft for discussion
V0r2	2013-11-18	PvdE	> Textual updates from discussions at F2F 2013-11-04. > Improved separation of the AS4 feature set (chapter 2.2) and the usage profile (2.3). For the feature set the audience are vendors and for the usage profile users/implementers. > Provided guidance for TLS based on ENISA and other guidelines (section 2.2.6.1). > Provided guidance on WS-Security based on ENISA guidelines, advice from XML Security experts (section 2.2.6.2). > Added test service (section **Error! Reference ource not found.**). > Added support for CL3055 (section 2.3.1.1). > Guidance on correlation is now mentioned as an option only, leaving choice between document-oriented and service-oriented exchanges (section 2.3.1.3). > More guidance on certificates (section 2.3.2.4). > Added a section on environments (section 2.3.5). > Added an example message (section 3). > Values to be confirmed: five minutes for retries (section 2.2.5), 10 MB total payload size (section 2.3.3)
V0r3	2013-11-29	PvdE	> Textual updates from F2F on 2013-11-21. > Added messaging model diagram (section 2.2.1). > Add note that Pull is not required to summary (section 2.2) > Added a diagram of AS4 message structure (section 2.2.3).

			> All payloads are carried in separate MIME parts; no support for external payloads; renamed from “attachments” to “payloads” (section 2.2.3.2). > The reference to TLS cipher suites is more general (section 2.2.6.1). > Simplified party identifiers, only EIC codes are allowed (section 2.3.1.1). > ENTSOG will publish Service/Action info (section 2.3.1.2). > Guidance on correlation is left to business processes (section 2.3.1.3). > Client authentication not recommended (section 2.3.2.2). > No preferred CA; state the 3072 is for future applications (section 2.3.2.4). > The test service is now in the Usage Profile as it can be provided via configuration (section 2.3.4). > The section on separating environments is simplified (section 2.3.5). > The usage profile on reliable messaging is removed. > Fixed reference to BSI TLS document (section 5).
V0r4	2013-12-04		> Updates based on discussions at F2F, 2013-12-03 > Disclaimer added. > In 2.2.1, explained Sender-Receiver concepts are orthogonal to Initiator-Responder. > Updated guidance on payload size. > Added RFC 6176 reference. > Improved wording on environments. > Anonymous EIC codes in example.
V0r5	2013-12-06	PvdE	> Draft finalized in team teleconference.
V0r6	2014-02-14	PvdE, EJvN	> Updates based on team teleconference > Generalized title of 2.3.2.4 and updated content

			to reflect the new appendix on certificate requirements. > Added reference to [BSIALG]. > Added discussion on key transport algorithms. > Updated AES encryption from to *http://www.w3.org/2001/04/xmlenc#aes128-cbc* to http://www.w3.org/2001/04/xmlenc#aes128-gcm following [XMLENC1]. > Added section **Error! Reference source not ound.**, **Error! Reference source not found.**.
V0r7	2014-04-22	PvdE	> ENISA comments: > In 2.3.2.1, change use of firewalls from MAY to SHOULD. > New section 2.2.7 which recommends IPv6.
V0r8	2014-07-28	PvdE	> The AES-GCM encryption URI is identified using *http://www.w3.org/2009/xmlenc11#aes128-gcm*. > Moved the certificate profile into the Usage Profile section. > Minor editorial changes.
V0r9	2014-07-30	PvdE	> Fixed header dates. Accepted all changes to fix Microsoft Word change track formatting errors.

5 References

[AES] Advanced Encryption Standard. FIPS 197. NIST, November 2001. http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

[AS4] AS4 Profile of ebMS 3.0 Version 1.0. OASIS Standard, 23 January 2013. http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/profiles/AS4-profile/v1.0/

[BSIALG] Entwurf Algorithmenkatalog 2014. Bundesamt für Sicherheit in der Informationstechnik (BSI). Bonn, 11 Oktober 2013. https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/ElekSignatur/Algorithmenkatalog_Entwurf_2013.pdf?__blob=publicationFile.

[BSITLS] Mindeststandard des BSI nach § 8 Abs. 1 Satz 1 BSIG für den Einsatz des SSL/TLS-Protokolls in der Bundesverwaltung. Bundesamt für Sicherheit in der Informationstechnik (BSI). Bonn, 08 Oktober 2013. https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Mindeststandards/Mindeststandard_BSI_TLS_1_2_Version_1_0.pdf

[CAM] Business Requirements Specification for the Capacity Allocation Mechanism (CAM) Network Code. Draft Version 0 Revision 05 – 2012-10-05.

[DNCIDX] ENTSOG Draft Network Code on Interoperability and Data Exchange Rules. http://www.entsog.eu/public/uploads/files/publications/INT%20Network%20Code/2013/ACERSubmission/INT0352-130910%20Network%20Code%20on%20Interoperability%20and%20Data%20Exchange%20Rules_Final.pdf

[EBMS3] OASIS ebXML Messaging Services Version 3.0: Part 1, Core Features. OASIS Standard. 1 October 2007. http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/os/

[EGCDN] Common Data Network. EASEE-gas Common Business Practice 2007-002/01. http://easee-gas.eu/docs/cbp/approved/CBP2007-002-01_DataNetwork.pdf

[EGMTP] Message Transmission Protocol. EASEE-gas Common Business Practice 2007-001/01. http://easee-gas.eu/docs/cbp/approved/CBP2007-001-01_MessageTransmissionProtocol.pdf

[EIC] ENTSOG. Energy Identification Coding Scheme (EIC) for natural gas transmission. Party Codes. http://www.entsog.eu/eic-codes/eic-party-codes-x

[EN 319 411-1] Draft European Standard. Electronic Signatures and Infrastructures (ESI); Policy and security requirements for Trust Service Providers issuing certificates; Part 1: Policy requirements for Certification Authorities issuing web site certificates, v0.0.4, 2013-11. http://docbox.etsi.org/esi/Open/Latest_Drafts/prEN-319411-1v004-Policy-req-for-CA-issuing-website-cert-STABLE-DRAFT.pdf

[EN 319 411-3] European Standard. Electronic Signatures and Infrastructures (ESI); Policy and security requirements for Trust Service Providers issuing certificates; Part 3: Policy requirements for Certification Authorities issuing public key certificates, v1.1.1, 2013-01. (Formerly [ETSI TS 102 042]) http://www.etsi.org/deliver/etsi_EN/319400_319499/31941103/01.01.01_60/EN_31941103v010101p.pdf

[ENISAAKSP] Algorithms, Key Sizes and Parameters Report 2013 recommendations version 1.0 – October 2013. ENISA. http://www.enisa.europa.eu/activities/identity-and-trust/library/deliverables/algorithms-key-sizes-and-parameters-report

[ENTCREQ] ENTSOG Certificate Requirements. Version 0.5, February 2014.

[NOM] Business Requirements Specification for the Nomination (NOM) Network Code. Draft Version 0 Revision 9 – 2013-06-04.

[OSSLTLS] OpenSSL TLS 1.2 Cipher Suites. http://www.openssl.org/docs/apps/ciphers.html#TLS_v1_2_cipher_suites.

[RFC2119] A. Ramos. Key words for use in RFCs to Indicate Requirement Levels. IETF RFC 2119. January 1998. http://www.ietf.org/rfc/rfc2119.txt

[RFC5246] T. Dierks et al. The Transport Layer Security (TLS) Protocol Version 1.2. IETF RFC 5246. August 2008. http://tools.ietf.org/html/rfc5246

[RFC6176] S. Turner et al.Prohibiting Secure Sockets Layer (SSL) Version 2.0. RFC 6176. March 2011. http://tools.ietf.org/html/rfc6176

[RFC6555] D. Wing et al. Happy Eyeballs: Success with Dual-Stack Hosts. http://tools.ietf.org/html/rfc6555

[TLSSP] Transport Layer Security (TLS) Parameters. Last Updated 2013-10-03. http://www.iana.org/assignments/tls-parameters/tls-parameters.xml#tls-parameters-4

[WSSSMS] OASIS Web Services Security: SOAP Message Security Version 1.1.1. OASIS Standard, May 2012. http://docs.oasis-open.org/wss-m/wss/v1.1.1/wss-SOAPMessageSecurity-v1.1.1.doc

[WSSSWA] OASIS Web Services Security: Web Services Security SOAP Message with Attachments (SwA) Profile Version 1.1.1. OASIS Standard, May 2012. http://docs.oasis-open.org/wss-m/wss/v1.1.1/wss-SwAProfile-v1.1.1.doc

[WSSX509] OASIS Web Services Security: Web Services Security X.509 Certificate Token Profile Version 1.1.1. OASIS Standard, May 2012. http://docs.oasis-open.org/wss-m/wss/v1.1.1/wss-x509TokenProfile-v1.1.1.doc

[XMLDSIG] XML Signature Syntax and Processing (Second Edition). W3C Recommendation 10 June 2008.

[XMLDSIG1] XML Signature Syntax and Processing Version 1.1. W3C Recommendation 11 April 2013. http://www.w3.org/TR/xmldsig-core1/

[XDSIGBP] XML Signature Best Practices. W3C Working Group Note 11 April 2013. http://www.w3.org/TR/2013/NOTE-xmldsig-bestpractices-20130411/

[XMLENC] XML Encryption Syntax and Processing. W3C Recommendation 10 December 2002. http://www.w3.org/TR/xmlenc-core/

[XMLENC1] XML Encryption Syntax and Processing Version 1.1. W3C Recommendation 11 April 2013. http://www.w3.org/TR/xmlenc-core1/