This document is classified as **White** in accordance with the Panel Information Policy. Information can be shared with the public, and any members may publish the information, subject to copyright.

MP129 'Allowing the use of CNSA variant for ECDSA'

Annex B

Legal text – version 1.0

About this document

This document contains the redlined changes to the SEC that would be required to deliver this Modification Proposal.

Schedule 8 'Great Britain Companion Specification'

These changes have been redlined against Schedule 8 version 4.1.

These changes will be applied to version 4.n.

Amend Section 4.3.2.3 as follows:

4.3.2 Security Credentials

4.3.2.1 Introduction – informative

A Device shall be able to process four kinds of Security Credential Document:

- its own Security Credential Documents, provided in the form of Device Certificates. Here the Device needs processing to cover (1) generating new Public-Private Key Pairs and so issuing Device Certification Requests, (2) storing its Device Certificates and (3) providing a copy of those Device Certificates on request;
- Security Credential Documents relating to Known Remote Parties, provided in the form of Organisation Certificates. For these, the Device needs to be capable of (1) storing, (2) replacing and (3) providing details of those it holds on request;
- Security Credential Documents relating to Unknown Remote Parties, provided in the form of Organisation Certificates. For these, the Device will receive them in a Command so that parts of the Response can be Encrypted. The Device does not need to store such Documents; and
- Security Credential Documents relating to Certification Authorities, provided in the form of Certification Authority Certificates. These are processed by the Device only when replacing Remote Parties' Security Credential Documents.

Sections 8 and 13 cover the above functionality.

Section 12 covers requirements related to the structure and content of such Security Credential Documents, where such requirements are relevant to Device processing requirements.

This Section 4.3.2 covers requirements for the storage of such Security Credentials on Devices and their usage in verifying cryptographic protections on Commands the Device receives.

4.3.2.2 Security Credential Documents

A Security Credential Document shall be either:

- a Device Certificate; or
- a Remote Party's Organisation Certificate; or
- a Certification Authority Certificate.

4.3.2.2.1 Device Certificate

A Device Certificate shall relate to only one Device and shall meet the requirements specified at Section 12. A Device Certificate shall either be used for Key Agreement or Digital Signing but not both. Device Certificates shall only be issued by Authorised Public Key Infrastructure (APKI) issuing Certification Authority. Where Security Credentials relating

to a Device are incorporated in a Message, the Security Credentials shall be incorporated in the Message in the form of the Device Certificate.

4.3.2.2.2 Remote Party's Certificate

A Remote Party Certificate shall be one of that Remote Party's Organisation Certificates and so shall relate to only one Remote Party and shall meet the requirements specified at Section 12. As per Section 12, except where `remotePartyRole` = `root` a Remote Party Certificate shall either be used for Key Agreement or Digital Signing but not both. Remote Party Certificates shall only be issued by APKI authorised issuing Certification Authorities. Where Security Credentials relating to a Remote Party are incorporated in a Message, the Security Credentials shall be incorporated in the Message in the form of the Remote Party's Certificate.

4.3.2.2.3 Certification Authority Certificate

A Certification Authority Certificate shall relate to only one Certification Authority and shall meet the requirements specified at Section 12. A Certification Authority Certificate shall only be used by a Device for verifying Digital Signatures on Certificates. Where Security Credentials relating to a Certification Authority are incorporated in a Message, the Security Credentials shall be incorporated in the Message in the form of the Certification Authority's Certificate.

4.3.2.3 Device Security Credentials

Where a Device is of `deviceType` that is `gSME, eSME`[8], `communicationsHubCommunicationsHubFunction`, or `communicationsHubGasProxyFunction`, that Device shall have the capacity to store and use securely four private keys:

- for Key Agreement, a Current Private Key and a Pending Private Key; and
- for Digital Signing, a Current Private Key and a Pending Private Key.

Where a Device is of `deviceType` that is `type1HANConnectedAuxiliaryLoadControlSwitch or type1PrepaymentInterfaceDevice`, that Device shall have the capacity to store and use securely two private keys:

- for Key Agreement, a Current Private Key; and
- for Digital Signing, a Current Private Key.

These stores shall be referred to as Private Key Cells.

Wherever one of a Device's Private Keys is required to be used by a GBCS Cryptographic Protection process, only the relevant Current Private Key shall be used. A Device shall not use any Pending Private Key in any GBCS Cryptographic Protection.

Where a Device holds a Private Key that is to be used for Key Agreement, the corresponding Public-Private Key Pair shall have been generated according to the section B.4.1 of *FIPS 186-4*~~NSA's '*Suite B Implementer's Guide to FIPS 186-3 (ECDSA), February 3, 2010*'~~[9] using the 'ECC Key Pair Generation Using Extra Random Bits' method.

[8] So including SAPC as per Section 0.

[9] https://csrc.nist.gov/publications/detail/fips/186/4/final~~https://www.iad.gov/iad/library/ia-guidance/ia-solutions-for-classified/algorithm-guidance/suite-b-implementers-guide-to-fips-186-3-ecdsa.cfm~~

Where a Device holds a Private Key that is to be used for Digital Signing, the corresponding Key Pair shall have been generated according to the section B.4.1 of *FIPS 186-4*~~NSA's '*Suite B Implementer's Guide to FIPS 186-3 (ECDSA), February 3, 2010*'~~[10] using the 'ECC Key Pair Generation Using Extra Random Bits' method.

Where a Device supports the processing of Remote Party Messages, the Device shall:

- have two Trust Anchor Cells to store two Device Certificates relating to itself, with one Trust Anchor Cell for storing Device Certificates where `keyUsage = keyAgreement` and one for Device Certificates where `keyUsage = digitalSignature`;
- where those two Trust Anchor Cells are populated, ensure the Device Certificates have the following attributes:
  - both Device Certificates meet the requirements specified at Section 13;
  - both Device Certificates' `hwSerialNum` fields have a value the same as the Devices' Entity Identifier; and
  - each Device Certificate's `keyUsage` field has the same value as the Trust Anchor Cell in which it is placed.

4.3.2.4 Remote Party Security Credentials

A Device shall only action a Remote Party Command where:

- the Known Remote Party identified by the Command has, according to the Security Credentials held on the Device, a Remote Party Role which, according to the Mapping Table for the Message Code in question, is allowed to request execution of the Command; and
- the Cryptographic Protections in the Command instance received by the Device have been verified, in line with the requirements for a Command with the Message Code in question.

To enable this, Security Credentials relating to the Remote Parties in question:

- shall be held in Trust Anchor Cells on the Device; and
- shall act as the corresponding Trust Anchors.

4.3.2.5 Required Trust Anchor Cells and related Device requirements

The Trust Anchor Cells specified in Table 4.3.2.5 by `TrustAnchorCellIdentifier` are those required on each `deviceType`. Additionally:

- a GSME shall have a Trust Anchor Cell capable of storing Key Agreement Security Credentials for a PPMID; and
- a PPMID shall have a Trust Anchor Cell capable of storing Key Agreement Security Credentials for a GSME.

The types of Device and the corresponding value of `deviceType` shall be defined in ASN.1 notation by:

[10] https://csrc.nist.gov/publications/detail/fips/186/4/final~~https://www.iad.gov/iad/library/ia-guidance/ia-solutions-for-classified/algorithm-guidance/suite-b-implementers-guide-to-fips-186-3-ecdsa.cfm~~

```
DeviceType ::= INTEGER {
   gSME                                                (0),
   eSME                                                (1),
   communicationsHubCommunicationsHubFunction          (2),
   CommunicationsHubGasProxyFunction                   (3),
   type1HANConnectedAuxiliaryLoadControlSwitch         (4),
   type1PrepaymentInterfaceDevice                      (5),
   type2                                               (6)
}
```

An SAPC shall have a `deviceType` of `eSME (1)` throughout this GBCS.

Every Device shall:

- have storage allocated capable of holding Security Credentials as required by Table 4.3.2.5 for its Device type; and
- have all the Trust Anchor Cells, specified in Table 4.3.2.5 as being required for its Device type, populated with Security Credentials that comply with the requirements of this GBCS. Critically, `root`, `recovery` and `accessControlBroker` Trust Anchor Cells shall be populated with valid credentials for each of those three Remote Parties.

				Type of Device (✓= is required; empty = is not required)					
				ESME /SAPC	GSME	CH (CHF)	CH (GPF[11])	HCALCS	PPMID
	deviceType value(s)			1	0	2	3	4	5
	TrustAnchorCellIdentifier								
No	**remotePartyRole**	**keyUsage**	**cellUsage**						
1	root	keyCertSign	management	✓	✓	✓	✓	✓	✓
2	recovery	digitalSignature	management	✓	✓	✓	✓	✓	✓
3	supplier	digitalSignature	management	✓	✓		✓	✓	
4	supplier	keyAgreement	management	✓	✓		✓		

[11] Supplier and Network Operator credentials on the Communications Hub (Gas Proxy Function) relate to the supply of gas only. These Trust Anchor Cells on a Communications Hub are still required and valid where there is no GSME connected to the SMHAN, but the stores should be populated with Access Control Broker certificates (so ensuring the Gas Proxy Function functionality, apart from Update Security Credentials, is inoperable)

5	supplier	keyAgreement	prePaymentTopUp	✓ [12]	✓				
6	networkOperator	digitalSignature	management	✓			✓		
7	networkOperator	keyAgreement	management	✓			✓		
8	accessControlBroker	digitalSignature	management			✓			✓
9	accessControlBroker	keyAgreement	management	✓	✓	✓	✓	✓	✓
10	transitionalCoS	digitalSignature	management	✓	✓		✓	✓	
11	wanProvider	digitalSignature	management			✓			
12	loadController	digitalSignature	management	✓					
13	loadController	keyAgreement	management	✓					

Table 0: Requirements for Trust Anchor Cells by Device Type

For clarity, the GPF and CHF shall each have their own set of Trust Anchor Cells.

A specific Trust Anchor Cell shall be identified in this GBCS using the notation `{remotePartyRole, keyUsage, cellUsage}`. For example `{supplier, digitalSignature, management}` shall refer to the Trust Anchor Cell that holds the Device's Supplier Digital Signing Security Credentials, so including the Supplier's:

- Entity Identifier;
- Remote Party Role; and
- Digital Signing Public Key.

Where a Device supports the processing of Remote Party Messages, that Device:

- shall support the processing of the Update Security Credentials Commands required by Section 13; and
- shall not allow execution of any Remote Party Command other than an Update Security Credentials Command or Provide Security Credentials Command, nor issue any Remote Party Alerts, in relation to a Remote Party Role where the Remote Party Role stored in a Trust Anchor Cell is different than that of the Trust Anchor Cell itself.

When verifying a Cryptographic Protection applied to a Command instance it receives, a Device shall use the Remote Party Security Credentials that it holds at the time of Command processing.

Devices shall only be capable of replacing Remote Party Security Credentials on receipt of an Update Security Credentials Command specified in this GBCS.

[12] This Trust Anchor Cell is required on an SAPC, recognising that, in line with SMETS, the SAPC may or may not support prepayment related functionality.

4.3.2.6 What is the Public Key in each Trust Anchor Cell to be used for – informative

TrustAnchorCellIdentifier			Usage of the Public Key in the Trust Anchor Cell
remotePartyRole	keyUsage	cellUsage	
root	keyCertSign	management	Used only in Certification Path Validation to check that Certification Authority Certificates and Certificates related to change of root credentials were validly issued
recovery	digitalSignature	management	Used only to verify recovery's signature on Update Security Credentials Commands addressed to the Device
supplier	digitalSignature	management	Used to verify the supplier's signature on Critical Commands the supplier has addressed to the Device
supplier	keyAgreement	management	Used in applying MACs to Alerts and Responses addressed to the supplier, where they are not Critical. Used in encrypting data in Alerts and Responses addressed to the supplier
supplier	keyAgreement	prePaymentTopUp	Used to check the supplier MAC on Prepayment Top Up Commands. The supplier can decide whether this is the same key as the Key Agreement key used for other purposes
networkOperator	digitalSignature	management	Used to check the signature of the networkOperator on Critical Commands the networkOperator has sent to the Device. This only equates to Update Security Credentials Commands
networkOperator	keyAgreement	management	Used in applying MACs to Alerts and Responses addressed to the networkOperator, where they are not Critical. Used in encrypting data in Responses addressed to the networkOperator
accessControlBroker	digitalSignature	management	Used to verify the accessControlBroker's signature on Commands addressed to the Device
accessControlBroker	keyAgreement	management	Used in checking the accessControlBroker MAC on Commands received and to

TrustAnchorCellIdentifier			Usage of the Public Key in the Trust Anchor Cell
remotePartyRole	keyUsage	cellUsage	
			calculate the MAC for Responses addressed to the accessControlBroker
transitionalCoS	digitalSignature	management	Used only to check transitionalCoS's signature on Update Security Credentials Commands received by the Device
wanProvider	digitalSignature	management	Used by the Communications Hub (CHF) to verify the wanProvider's signature on Critical Commands addressed to the Communications Hub (CHF)
loadController	digitalSignature	management	Used to check the signature of the loadController on Critical Commands the loadController has sent to the Device
loadController	keyAgreement	management	Used in encrypting data in Alerts and Responses addressed to the loadController

Table 0: Use of Public Keys in each Trust Anchor Cell

4.3.2.7 Mapping a Command to the Remote Party Security Credentials to be used in verifying the Command's cryptographic protections

Except for the Security Credentials related Commands (see Section 13), a Device shall apply the requirements of this Section 4.3.2.7 to identify which of the Remote Party Public Keys that it holds are to be used to verify the cryptographic protections on a Command.

4.3.2.7.1 Message Authentication Codes

Where a Command is a Prepayment Top Up Command, the supplier MAC in that Command shall be verified using the Public Key in Trust Anchor Cell {**remotePartyRole** *supplier*, **keyUsage** *keyAgreement*, **cellUsage** *prePaymentTopUp*}, along with the Device's Key Agreement Private Key.

All other MACs in Commands shall be verified using the Public Key in Trust Anchor Cell {**remotePartyRole** *accessControlBroker*, **keyUsage** *keyAgreement*, **cellUsage** *management*}, along with the Device's Key Agreement Private Key.

4.3.2.7.2 Signature

Where a Command has a Digital Signature, the Device shall identify the Remote Party Role(s) which can legitimately sign the Command according to the message code identified in the Mapping Table.

If there is only one Remote Party Role so identified, then the signature shall be verified using the Public Key in Trust Anchor Cell {**remotePartyRole** *(the identified remote party role)*, **keyUsage** digitalSignature, **cellUsage** management}.

If there is more than one Remote Party Role so identified, the Device shall use the Business Originator ID in the Command to identify the Trust Anchor Cell(s) where:

- **keyUsage** = digitalSignature;

- **`cellUsage`** = `management`; and
- `existingSubjectUniqueID` = the Business Originator ID in the Command

If there is only one Trust Anchor Cell so identified, then the signature shall be verified using the Public Key in that Trust Anchor Cell.

If there is more than one Trust Anchor Cell so identified the Device shall attempt to verify the Digital Signature using each Trust Anchor Cell identified. These attempts shall be according to the following precedence, and attempts to verify shall cease when a signature verification succeeds:

1. supplier;
2. wanProvider;
3. networkOperator;
4. accessControlBroker.

For clarity, other Remote Party Roles on Devices are limited to Commands related to Security Credentials and so cannot have Trust Anchor Cells identified according to this Section 4.3.2.7.2.

4.3.2.8 Certification Path Validation

4.3.2.8.1 Access Control Broker requirements

Before it calculates the Access Control Broker to Device MAC (ACB-SMD MAC) in line with Section 6.2.3, the Access Control Broker shall undertake Certification Revocation List (CRL) Validation for any Organisation Certificate in a Command:

- either by using the algorithm specified in IETF RFC 5280[13] section 6.3; or
- by using functionality equivalent to the external behaviour resulting from that algorithm.

Only if the CRL Validation is successful shall the Access Control Broker calculate the ACB-SMD MAC. For clarity, the Access Control Broker shall never send a Message to a Device which contains any Certificate that has failed CRL Validation.

4.3.2.8.2 Device requirements

The requirements in this Section 4.3.2.8.2 shall apply only to Use Cases CS02b (Update Security Credentials) and CS02g (Update Load Controller Security Credentials).

Where a Device has successfully completed all required Command Authenticity and Integrity checks on a Command, of a type covered by Use Cases CS02b and CS02g, the Device shall undertake either:

- Certification Path Validation, including time checks; or
- Certification Path Validation, excluding time checks.

If the Device does not have Reliable Time (as defined in Use Cases GCS28 and ECS70 Set Clock) it shall always undertake Certification Path Validation, excluding time checks. Otherwise the validation to be undertaken shall be determined by the contents of the Remote Party Command instance. For clarity, Device types which are not required to have a clock, shall always undertake Certification Path Validation, excluding time checks.

[13] http://datatracker.ietf.org/doc/rfc5280/

The Device shall undertake Certification Path Validation, including time checks:

- either by using the algorithm specified in IETF RFC 5280 section 6.1; or
- by using functionality equivalent to the external behaviour resulting from that algorithm.

The Device shall undertake Certification Path Validation, excluding time checks:

- either by using the algorithm specified in IETF RFC 5280 section 6.1 but not applying the check at 6.1.3 (a) (2) ('the certificate validity period includes the current time'); or
- by using functionality equivalent to the external behaviour resulting from that algorithm where not applying the check that 'the certificate validity period includes the current time'.

The 'trust anchor' information (with the meaning in IETF RFC 5280) shall be in the `root` Security Credentials held on the Device.

If the Device's Certificate Path Validation does not confirm the required certification path validity, then the Device shall undertake no further processing of the Command, except for the issuance of a Message containing an `executionOutcome` (with its Section 13.3.4.6 meaning) notifying that the Command was unsuccessful in applying at least some of the changes in the Command.

4.3.2.9 DLMS Client and Server

The Access Control Broker shall perform the role of DLMS COSEM client in relation to the DLMS COSEM Application Associations, and the Device shall perform the role of DLMS COSEM server.

Amend Section 4.3.3.2 as follows:

4.3.3 Cryptographic primitives and their usage

In relation to any Remote Party Message, Smart Metering Entities shall:

- use SHA-256, as specified in FIPS 180-414, as the Hash function;
- use the AES-128 cipher, as specified in FIPS 19715, as the block cipher primitive;
- use the Galois Counter Mode (GCM) mode of operation as specified in NIST Special Publication 800-38D16 ;
- use the GMAC technique, based on the use of AES-128, for the calculation of Message Authentication Codes (MACs), as specified in NIST Special Publication 800-38D (see above);
- use, as the Digital Signature technique, ECDSA (as specified in FIPS PUB 186-4) in combination with the curve P-256 (as specified in FIPS PUB 186-4 at section D1.2.3)

[14] http://csrc.nist.gov/publications/fips/fips180-4/fips-180-4.pdf
[15] http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf
[16] http://csrc.nist.gov/publications/nistpubs/800-38D/SP-800-38D.pdf

and SHA-256 as the Hash function. Within Messages, Signatures shall be in the Plain Format;

- use, to calculate the Shared Secret Z, the Static Unified Model, C(0e, 2s, ECC CDH) Key Agreement technique (as specified in NIST Special Publication 800-56Ar217 save for the requirement to zeroize the Shared Secret) with:
  - the Single-step Key Derivation Function (KDF) based on SHA-256, as specified in NIST Special Publication 800-56Ar2; and
  - the P-256 curve for the elliptic curve operations.

Resulting DerivedKeyingMaterial (with its meaning in *NIST Special Publication 800-56Ar2*) shall only ever be used in relation to one Message instance. Any Shared Secret that is not 'zeroized' shall be stored and used with the same security protections as Private Keys.

4.3.3.1 Scope of Cryptographic Protections

The fields that shall always contribute to MAC and Digital Signature are detailed in Section 7.2. Fields that vary across Messages are specified in Section 6, and in the relevant Use Cases. For clarity, a Message instance may transit through multiple Smart Metering Entities before delivery to its target Device, and more than one Smart Metering Entity may be required to apply a Cryptographic Protection to that Message instance. Thus, the scope of protection can only be across fields in the Message instance as constructed at the point the protection is applied.

Where a Message has multiple Cryptographic Protections, the order in which the Smart Metering Entities apply these Cryptographic Protections is specified in this GBCS.

A Device verifying the Cryptographic Protections in such Messages shall undertake such verifications in the reverse sequence to that in which the Cryptographic Protections were applied. This order is also specified in this GBCS.

4.3.3.2 ECDSA per message secret number

When generating a Digital Signature, the Smart Metering Entity shall calculate the DSA Per-Message Secret Number '~~k~~k' with respect to ECDSA (with the meaning in section 6.3 of *FIPS 186-4*) to be the SHA-256 hash of the concatenation of:

- the parts of the Message to be signed, as defined in Section 7.2.7; and
- the Private Key that the Smart Metering Entity will use in the Digital Signature generation.

If the value of ~~k~~k so calculated is zero or greater than n -1, or results in an '~~r~~r' or '~~s~~s' value of 0, where ~~r~~r and ~~s~~s have the meanings in ~~the NSA's '*Suite B Implementer's Guide to FIPS 186-3 (ECDSA)*'~~ *FIPS 186-4*, then a new value for ~~k~~k shall be calculated to be the SHA-256 hash of the concatenation of:

- the parts of the Message to be signed, as defined in Section 7.2.7;
- the Private Key that the Smart Metering Entity will use in the Digital Signature generation; and
- 0x00.

[17] http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-56Ar2.pdf

The addition of 0x00 to the concatenation shall be repeated until a value of ~~k~~k is generated that does not result in ~~k~~k being zero or greater than n -1, or an '~~r~~r' or '~~s~~s' value of 0.

As an alternative to the above, a Remote Party may choose to derive *k* as defined in section 6.3 of *FIPS 186-4*, using either of the methods defined in section B.5 of *FIPS 186-4*.

4.3.3.3 Calculating unique Shared Secret Keys for a Remote Party Message Instance

Where a Smart Metering Entity executes the KDF in relation to a Message instance, the *OtherInfo* field, with the meaning in *NIST Special Publication 800-56Ar2*, shall be populated using the value of information provided in, or to be placed in, the originator-system-title, recipient-system-title and transaction-id fields of the Grouping Header, as per the requirements of Section 7.2.7.

The *OtherInfo* shall be in the Concatenation Format as defined in section 5.8.1.2.1 of NIST Special Publication 800-56Ar2 and shall be the concatenation:

AlgorithmID || value of originator-system-title || length of transaction-id || value of transaction-id || value of recipient-system-title

where:

- AlgorithmID is that for AES-GCM-128 and so has a value 0x60857406080300, as specified by section 9.2.3.4.6.5 of the Green Book; and
- length of transaction-id has the value 0x09.

4.3.3.4 Calculating the Initialization Vector for GCM and GMAC

In relation to Remote Party Messages, Smart Metering Entities shall use a 96 bit Initialization Vector (IV) for the GCM and GMAC algorithms as defined in *NIST Special Publication 800-38D*. The IV shall be the concatenation:

FixedField || *InvocationField*

where:

- FixedField shall always have the same value as the Business Originator ID in the Grouping Header part of the Message being processed (see Section 7.2.7); and
- InvocationField = 0x00000000.

The DLMS COSEM Authentication Key (AK), as defined in the Green Book, shall be a zero length string.

4.3.3.4.1 Other input parameters to MAC and Encryption / Decryption operations – informative

Other input parameters for MAC, Encryption and Decryption are not specified in this Section 4.3.3 because they vary dependent on a number of factors. These other input parameters are listed in tables of the same format as Table 4.3.3.4.1 and their values are specified in each part of the GBCS where such an operation is specified.

The template for such tables is the Table 4.3.3.4.1. Please note that this table does not contain any values as it is a template only.

Input Parameter	Value	Note
To calculate the Shared Secret ('*Z*') input to the KDF:		
Private Key Agreement Key		
Public Key Agreement Key		

Input Parameter	Value	Note
The other input to the KDF (‘*OtherInfo*’) shall be calculated according to the requirements of Section 4.3.3.3.		
As input to the GMAC function, the IV shall be constructed according to the requirements of Section 4.3.3.4, the Plaintext shall be empty and:		
Additional Authenticated Data shall be the concatenation:		

Table 4.3.3.4.1: Template for other input parameters

4.3.3.4.2 Size of MAC

The bit length of the MAC shall be 96 except for the MAC contained in the WrappedApexContingencyKey extension within root Certificates, where the bit length of the MAC shall be 128.