IEEE P802.15
Wireless Personal Area Networks

Project	IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs)	
Title	**TG3 LB19 security comment resolution**	
Date Submitted	[11 September, 2002]	
Source	[John R. Barr] [Motorola] [1303 E. Algonquin Road Schaumburg, IL 60196]	Voice: [847-576-8706] Fax: [847-576-6758] E-mail: [John.Barr@Motorola.com]
Re:	[802.15.3 D11]	
Abstract	[This document is a record of security comment resolutions for LB19.]	
Purpose	[To provide a record of the security comment resolution for LB19.]	
Notice	This document has been prepared to assist the IEEE P802.15. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein.	
Release	The contributor acknowledges and accepts that this contribution becomes the property of IEEE and may be made publicly available by P802.15.	

1. Security Comment Resolution, Monterey

1.1 Tuesday, September 10, 2002

404 - ACCEPT IN PRINCIPLE. Add "The DEV shall set the secure frame counter to 0 whenever it receives a new key." to the end of clause 7.2.8.2.

110 - ACCEPT

5 - ACCEPT IN PRINCIPLE. Change "it shall verify that the beacon number is greater than the MACPIB_CurrentBeaconNumber, that the SECID matches the MACPIB_PNCSECID stored in the MAC PIB and that the integrity code passes. If all of these checks succeed, the DEV shall set the MACPIB_CurrentBeaconNumber to the received beacon number value and set the MACPIB_ValidBeacon to valid. If the beacon number is greater than the MACPIB_CurrentBeaconNumber, but the SECID does not

match the MACPIB_PNCSECID, the device may set the MACPIB_CurrentBeaconNumber to the value in the beacon and send a key request command to the PNC to obtain the new key."

to

"it shall verify that the beacon number in the beacon is greater than the LastKnownGoodBeaconNumber, that the SECID matches the stored broadcast SECID and the integrity code passes. If all of these checks succeed, accept the beacon number in the beacon as the LastKnownGoodBeaconNumber. If the beacon number in the beacon is greater than the LastKnownGoodBeaconNumber, but the SECID does not match the stored broadcast SECID, the device may set the LastKnownGoodBeaconNumber to the value in the beacon and send a key request command to the PNC to obtain the new key."

Delete "stored in the MACPIB_CurrentBeaconNumber in the MAC PIB" from line 16 on page 221.

On page 220, line 41, Change "greater than the CurrentBeaconNumber and less than the CurrentBeaconNumber + aMaxBeaconChange." to "greater than the LastKnownGoodBeaconNumber and less than the LastKnownGoodBeaconNumber + aMaxBeaconChange."

6 - ACCEPT IN PRINCIPLE. Move the definition of public-key object type from 7.5.2.1 to 7.4.15. Replace line 16 on page 128 with lines 1-11 on page 134. Replace lines 1-11 on page 134 with "The public-key object type field specifies the type of public key specified in the public-key object {xref 7.4.15}."

403 - REJECT. To maintain proper synchronization of keying material, the SECID is required. It is the only way a DEV can determine that it is out of sync.

113 - ACCEPT

99 - ACCEPT IN PRINCIPLE. See resolution of CID 6 which did this.

114 - ACCEPT IN PRINCIPLE. On line 30 change "and security manager" to "and security manager of the piconet ". On line 31 change "security manager acts as the central security point for all DEVs to obtain keying material for the piconet." to "piconet security manager authenticates DEVs for membership in the piconet and provides the broadcast payload protection key for the piconet." On line 37, change "all piconet data." to "data using the broadcast key.".

78 - ACCEPT

82 - REJECT. The PublicKeyObjectLength for ECC and RSA certificates can vary. Table 10 on page 43 incorrectly states that PublicKeyObjectLength is defined by the security suite. Change "Defined by the security suite, Clause 10." to "May be constant or variable as defined by the security suite, Clause 10."

83 - ACCEPT IN PRINCIPLE. Remove OID and OIDLength from 6.3.7.5 and 6.3.7.6. And update Figure 146 in 9.8.3 to reflect this change.

79 - ACCEPT

366 - ACCEPT

402 - ACCEPT

371 - ACCEPT IN PRINCIPLE. Change Pub_D to Pub_key_D. Change Pr_D to Pr_key_D. Change Pub_SM to Pub_key_SM. Change Pr_SM to Pr_key_SM. Change Keys_D to Sym_keys_D. Change Keys_G to Sym_keys_G. Change BH to SBH and add SCH for secure command header. Beacon and com-

mand shouldn't be lumped together. Add CD for command Data to separate it from BD. Use these identifiers as a guideline and update tables, text and figures in clause 9 as required.

432 - ACCEPT

107 - ACCEPT IN PRINCIPLE. Reference changed to clause where correction is to be made. Add BAD-BEACON-NUMBER to Table 14, page 54, line 49 since it is referenced in clause 9.2.8 on page 220, line 33.

382 - ACCEPT

241 - ACCEPT IN PRINCIPLE. Change "failure" on line 45 to "ChallengeResponse generation failure".

239 - ACCEPT IN PRINCIPLE. Change "Failure" to "Challenge verification failure".

86 - ACCEPT IN PRINCIPLE. The EncryptedSeed should be changed to Key and the IntegrityCode should be removed from the MLME-REQUEST-KEY.response. Change "with an encrypted version of therequested seed" to "with the requested key" in line 4.

90 - ACCEPT

20 - ACCEPT IN PRINCIPLE. Change "EncryptionSeed" on line 32 to "Key". Delete "in an encrypted format" from line 26.

89 - ACCEPT

88 - ACCEPT

87 - ACCEPT

1.2 Thursday September 12, 2002

102 - ACCEPT IN PRINCIPLE:

Author's note: Add the following sub-clause to 6.3 after 6.3.13. OIDs are currently sent on a per-device basis because a security manager may support multiple OIDs.

1.2.1 Retrieving ACL information

There primitives are used to request ACL information about other DEVs in the piconet. The parameters used for the MLME-ACL-INFO primitives are defined in Table 1.

1.2.1.1 MLME-ACL-INFO.request

This primitive initiates a request to the DEV for ACL information regarding either a single DEV or all of the DEVs in the piconet. The semantics of the primitive are as follows:

```
MLME-ACL-INFO.request        (
                             TrgtID,
                             QueriedDEVID,
                             ACLInfoTimeout
                             )
```

The primitive parameters are defined in Table 1.

Table 1—MLME-ACL-INFO primitive parameters

Name	Type	Valid range	Description
QueriedDEVID	Integer	Any valid DEVID as defined in 7.2.3.	Specifies the DEVID of the DEV for which information is being requested. A value of BcstID is defined as a request for information from all associated DEVs.
TrgtID	Integer	Any valid DEVID as defined in 7.2.3.	Specifies the DEVID of the DEV that the ACL information request is intended for.
OrigID	Integer	Any valid DEVID as defined in 7.2.3.	Specifies the DEVID of the DEV that initiated the MLME request.
NumACLInfoSet	Integer	0-65535	Number of enties in the ACLInfoSet
ACLInfoSet	As defined in 1.2.2.2.	As defined in 1.2.2.2.	A set of ACL entry elements for the requested DEVs.
ACLInfoTimeout	Duration	0-65535	The time in milliseconds in which the operation initiated by the MLME request needs to be completed before responding with a ResultCode of TIMEOUT.
ResultCode	Enumeration	SUCCESS, DENIED, TIMEOUT	Indicates the result of the MLME request.

1.2.1.1.1 When generated

The originating DME sends this primitive to its MLME when it wants to obtain ACL information about either an individual DEV or all of the DEVs in the piconet.

1.2.1.1.2 Effect of receipt

The MLME, upon receiving this primitive, sends the ACL information request command, 1.2.2.1, to the DEV specified by the TrgtID to request security information managed by that DEV.

1.2.1.2 MLME-ACL-INFO.indication

This primitive indicates the reception of a request by a DEV for ACL information it manages regarding either a specific DEV or all of the DEVs in the piconet. The semantics of the primitive are as follows:

```
MLME-ACL-INFO.indication        (
                                QueriedDEVID,
                                OrigID
                                )
```

The primitive parameters are defined in Table 1.

1.2.1.2.1 When generated

The DEV MLME sends this primitive to its associated DME upon receiving an ACL information request command, 1.2.2.1, from the requesting DEV specified by the OrigID.

1.2.1.2.2 Effect of receipt

The DME upon receiving this primitive sends an MLME-ACL-INFO.response to its MLME.

1.2.1.3 MLME-ACL-INFO.response

This primitive initiates a DME response to an MLME-ACL-INFO.indication. The semantics of the primitive are as follows:

```
MLME-ACL-INFO.response      (
                            OrigID,
                            NumACLInfoSet,
                            ACLInfoSet
                            )
```

The primitive parameters are defined in Table 1.

1.2.1.3.1 When generated

The DME sends this primitive to its MLME as a result of receiving an MLME-ACL-INFO.indication.

1.2.1.3.2 Effect of receipt

The MLME upon receiving this primitive sends an ACL information command, 1.2.2.2, to the requesting DEV.

1.2.1.4 MLME-ACL-INFO.confirm

This primitive informs the originating DME that its request for ACL information from the specified DEV is complete. The semantics of the primitive are as follows:

```
MLME-ACL-INFO.confirm       (
                            TrgtID,
                            NumACLInfoSet,
                            ACLInfoSet,
                            ResultCode
                            )
```

The primitive parameters are defined in Table 1.

1.2.1.4.1 When generated

The MLME sends this primitive to its DME upon receiving either an ACL information command, 1.2.2.2, or a TIMEOUT.

1.2.1.4.2 Effect of receipt

The originating DME is informed whether its request for information about either a single DEV or all of the DEVs in the piconet was successful or unsuccessful. If unsuccessful, the DME is able to resend the MLME-ACL-INFO.request. If successful, the DME will have acquired the information it requested.

Author's note: Add the following sub-clauses to 7.5.

1.2.2 ACL information commands

The ACL information commands allow a DEV to request information about the public key of a specified DEV or DEVs.

1.2.2.1 ACL information request command

The ACL information request command shall be formatted as illustrated in Figure 1.

octets: 1	2	2
Queried DEVID	Length (=1)	Command type

Figure 1—ACL information request command format

The queried DEVID indicates the DEV whose ACL information is being requested. If the value of this field is the BcstID, then the DEV is requesting all of the ACL information maintained by the target DEV.

1.2.2.2 ACL information command

The ACL information command shall be formatted as illustrated in Figure 2.

octets: L_m	---	L_2	L_1	1	1	2	2
DEV-m ACL record	…	DEV-2 ACL record	DEV-1 ACL record	Sequence num-ber	Total Number of Frame	Length = $1+L_1+L_2+_{...}+L_m$	Command type

Figure 2—ACL information command

The Total number of frames indicates the number of frames that will be sent to complete this request. The Sequence Number indicates which frame in the sequence is in this command.

The ACL record field shall be formatted as illustrated in Figure 3.

octets: L_2	2	1	1	6	2
Verification info	Verification info length = L_2	Verification info type	DEVID	DEV address	Length

Figure 3—Format of an ACL record in an ACL information command

The DEVID is the ID assigned to the DEV by the PNC. If the DEV is not currently associated in this pico-net, the field shall be set to the UnassocID. This field shall not contain the broadcast or multicast IDs.

The DEV address is the MAC address of the DEV corresponding to the DEVID.

The verification info type indicates the type of verification information that is included in the ACL entry. The valid verification info types are:

0 -> NULL
1 -> ECMQV Koblitz-283 key
2 -> RSA-OAEP raw 1024 key
3 -> NTRUEncrypt 251-1 key
4 -> ECMQV Koblitz-283 implicit certificate
5 -> X.509 certificate
6 -> X.509 CA certificate
7 -> ECMQV Koblitz-283 CA key
{Ed note: Make the type entry two four bit fields, one for the security suite type and one for the hash code.}
8 -> SHA-1 hash
9 -> SHA-256 hash
10 -> Certificate chain URL
11-255 -> Reserved

These types are defined in 1.2.4.

The verification info length indicates the length of the verification information that is included in the ACL entry. If this length is 0, no verification information field shall be included.

The verification info specifies the ACL verification info that may be used to verify the validity of the public key associated with that particular DEV.

Author's note: Replace sub-clause 9.2.4 with the following sub-clause.

1.2.3 PNC handover

When a PNC chooses to handover the PNC role to another DEV in the piconet, the authentication relationships with the old PNC no longer apply to the new PNC. When the old PNC hands over the piconet information using a PNC information command, 7.5.4.2, the list of authenticated DEVs is passed to the new PNC.

PNC handover does not affect the group membership, so it does not require a rekey of the group keys. However, in a piconet with payload protection, the command functions of the PNC that relate to specific DEVs are not implemented until the new PNC has performed the authentication protocol with each DEV in the piconet. When the PNC role has been handed over, the new PNC shall set up time slots for each of the authenticated DEVs to perform the authentication protocol with the new PNC if commands are not allowed in the CAP.

The new PNC or any other DEV may request public-key verification information from the old PNC using an ACL information command, 10.0.2.2.

Author's note: Add the following sub-clause at the end of 10.2.

1.2.4 Public-key verification information

DEVs may exchange public-key verification information that is intended to be used to verify public keys received in the authentication request command and challenge request command. The types of public-key verification information are specified in 1.2.2.2. The types have the following meanings:

Types 1 through 5 correspond to the actual public key objects themselves. {Ed Note: Revise to match frame format specificaiton.}

Type 6, X.509 CA certificate, corresponds to an X.509 certificate as defined in RFC 3280 that belongs to the CA that signed the corresponding DEV's certificate.

Type 7, ECMQV Koblitz-283 CA key, corresponds to the ECC public key of the CA that signed an implicit certificate for the ecmqv-implicit-1 sub-suite.

Type 8 , SHA-1 hash, is used to transmit the hash of the public-key and ID used in the rsa-oaep-raw-1 and ntruencrypt-raw-1 sub-suites.

Type 9 , SHA-256 hash, is used to transmit the hash of the public-key and ID used in the ecmqv-manual-1 sub-suite.

Type 10, certificate chain url, is used to transmit a uniform resource locator of the certificate chain from which a certificate is built.

319 - ACCEPT IN PRINCIPLE. Delete on page 265, lines 30-31 and Figure 163.

372 - ACCEPT IN PRINCIPLE. Resolve as indicate in CID 374.

373 - Withdrawn

374 - ACCEPT IN PRINCIPLE. Figure 147 is the MSC for Authentication. Page 242, line 4 add "The MSC for authentication is shown in {xref Figure 147}. Page 246, line 2 add "The MSC for authentication is shown in {xref Figure 147}. Delete clause 9.8.3.5 containing Figure 150.

375 - ACCEPT IN PRINCIPLE. Page 256, line 4 add "The MSCs that correspond with this state machine are shown in {xref Figure 147, 153, 154, 156, and 162 Ed to fix}."

376 - ACCEPT IN PRINCIPLE. Page 259, line 17 add "The MSCs that correspond with this state machine are shown in {xref Figure 147, 153, 154, 156, and 162 Ed to fix}."

377 - ACCEPT IN PRINCIPLE. Delete subclause 9.8.7.5 containing Figure 159.

378 - ACCEPT IN PRINCIPLE. Delete subclause 9.8.7.6 containing Figure 160.

379 - ACCEPT IN PRINCIPLE. Delete subclause 9.8.7.7 containing Figure 161.

380 - Withdrawn 12 September, 2002

381 - Withdrawn 12 September, 2002

85 - ACCEPT

81 - ACCEPT

367 - ACCEPT IN PRINCIPLE. Add the following to the sentence that ends with "to join the secure pico-net." New text: "and that the DEV accepts the PNC."

325 - ACCEPT IN PRINCIPLE. Resolve as indicated in CID 81.

326 - ACCEPT IN PRINCIPLE. Add OriginatorDEVAddress parameter to MLME_Authenticate.indication primitive and Table 10. The description is "The DEV Address of the originator of the authentication process." Add TargetDEVAddress to MLME_Challenge.request. The description is "The DEV Address of the security manager."

327 - ACCEPT IN PRINCIPLE. Delete the OID and OIDLength parms from the MLME-CHALLENGE.request parm list.Insert the ReasonCode parm just before the SECID parm in the MLME-CHALLENGE.request parm list. Leave TrgtID and OrigID as defined.

328 - ACCEPT IN PRINCIPLE. Add the ReasonCode parm just before the SECID in the MLME-CHALLENGE.indication parm list.Remove the OIDLength and OID parms from the MLME-CHALLENGE.indication parm list.

329 - Withdrawn 12 September, 2002

84 - Withdrawn 12 September, 2002

330 - Withdrawn 12 September 2002.

397 - REJECT. To avoid potential security problems it is best to not have any exceptions to the defined security policies. In addition, disassociation or deauthentication causes a broadcast key change that will require authentication.

1 - ACCEPT IN PRINCIPLE. Resolve as in CID 5.

118 - REJECT. Agree that the command will contain redundant information, but the BRC does not wish to change the frame format to remove the redundancy.

119 - ACCEPT IN PRINCIPLE. Add to the end of each of the descriptions in Table 89, ", as defined in 10.3.1.3." Proposed diagrams for authentication in the Ntru and RSA suites will be provided by Ari Singer.

120 - ACCEPT IN PRINCIPLE. In table 94 change "Authorization of public key" to Verification of public key".

295 - ACCEPT IN PRINCIPLE. Allow the distribute key request command to be sent with no ack policy. This allows the PNC to open a broadcast slot and send all of the distribute key commands for all DEVs quickly. Add page 136, line 45, "This command may have the ACK policy set to no ack if the source ID is the PNCID." Page 219, line 29 change sentence beginning with "Once all …" to "Once the distribute key command has been issued for all of the authenticated DEVs that are in ACTIVE mode, the PNC may change the SECID in the beacon."

61 - Withdrawn 12 September 2002.

77 - ACCEPT

121 - ACCEPT

123 - ACCEPT

80 - ACCEPT

98 - ACCEPT IN PRINCIPLE: Change as indicated in 02399r1.

Change figure 10 to the following:

b7-b4	b3	b2	b1	b0
Reserved	SEC use	CAP association	CAP commands	CAP data

Figure 4—Piconet mode field

Change the text and Table 40 to the following:

"The SEC use indicates whether secure authentication is required in the piconet. The field is encoded as illustrated in Table 2:"

Table 2—SEC use field encodings

Type value b3	SEC use
0	Authentication required
1	Authentication not required

108 - ACCEPT IN PRINCIPLE. Resolve as indicated in CID 5.

2 - ACCEPT IN PRINCIPLE. Resolve as indicated in CID 5.

3 - ACCEPT IN PRINCIPLE. Resolve as indicated in CID 5.

1.3 Thursday 19 September, 2002

Attendees: John Barr, Ari Singer, Dan Bailey, Rene Struik, Gregg Rasor, Bill Shvodian

Start - 2PM CDT

Some comments on work done in Monterey were raised by Ari. Comment 306 was changed to 366 since it was recorded incorrectly. The use of LastKnownGoodBeaconNumber in resolution of comment 5 raised the question of where it was defined. From what James said in Monterey, it just needs to be used consistently. Question on comment 120 whether the resolution does enough. Discussion moved to who needs to be able to interpret public key objects. Comment 102, it is up to the security suite to determine how the verification information passed in a DEV ACL record in interpreted. Ari will provide description of the verification information for the Ntru and RSA security suites.

Comment 92 & 98 - Made security a single mode. Rene - We need more than just a single mode. Document 02364r0 is an explanation. Only justification for removing modes is simplification, but no measure of simplification given. Rene, John and Gregg argue for a mode where the entire piconet (even peer ro peer) can be guaranteed to run using certificates to prevent any "man in the middle" attacks. Some users (e.g. CableLabs

compliant products) will require use of certificates to identify devices and ensure that the network is not compromised.

End - 3:50 CDT

1.4 Week of September 30, 2002

384 (Barr, TR) [SEC/PIB] MAC PIB ACL group defined as an array whose contents are defined in Table 33. All of the entries are dynamic, but no clear mechanism to update these entries has been included in the draft. There are no limits on the minimum and maximum number of entries allowed in the ACL. The only use for this array in the MAC is for generation of the CCM nonce and obtaining the keys associated with a particular SECID for encoding or decoding payloads. Either remove the MAC PIB ACL or add appropriate method for updating the information in the array. If the ACL is kept, add limit for the minimum number of ACLs that must be supported for a DEV, SM, and PNC. Provide a mechanism for updating and accessing the contents of an ACL entry. Suggest defining MLME commands for doing this using an index to the array. Add MAC PIB entries to indicate last index used in the array. Finally, clarify relationship between DEVHost and MAC regarding use and management of informaiton in the ACL. **Suggest accept in principle.** Add new MLME as indicated in 02399r4 section 1.4.1. Remove sub-clause 6.5.6 on page 87, lines 8-31. Add MLME-SECID-UPDATE.req on DEV and PNC sides to end of figure 147 on page 241.

1.4.1 Initializing and Updating SECID Information

This primitive is used to initialize or update the management security information associated with a new SECID as the result of an authorization process. The parameters used for the MLME-SECID-UUPDATE primitive are defined in Table 3.

Table 3—MLME-SECID-UPDATE primitive parameters

Name	Type	Valid range	Description
ManagementSECID	Integer	Any valid SECID as defined in {xref 7.2.8.1}	Specifies the security session ID for the management key.
TrgtID	Integer	Any valid DEVID as defined in {xref 7.2.3}.	The DEVID of the target DEV for this relationship.
SecurityManager	Boolean	True/False	This DEV is the security manager for this relationship.
KeyInfoLength	Integer	0-255	Length of ManagementKeyInfo
ManagementKeyInfo	Dynamic	Any valid payload protection key as defined by the security suite, {xref 10}.	The key agreed upon during authentication that are used for protecting commands between this DEV and the TrgtID DEV.

1.4.1.1 SECID-UPDATE.request

This primitive requests that the SECID and management keying information associated with the DEV be included or updated. The semantics of the primitive are as follows:

```
MLME-SECID-UPDATE.request    (
                             ManagementSECID,
                             TrgtID,
                             SecurityManager,
                             KeyInfoLength
                             ManagementKeyInfo
                             )
```

The primitive parameters are defined in Table 1.

1.4.1.1.1 When generated

The DME sends this request to the MLME after completing authentication with the PNC or a peer DEV.

1.4.1.1.2 Effect of receipt

The MLME adds this SECID to the list of authenticated SECIDs that can be used to protect command data with the target DEV in this authentication relationship.

1.4.2 Following provided by Ari Singer, Ntru.

54 (Gilb, TR) [SEC] We should specify that commands that fail IC check should be ignored. Sometimes they aren't (e.g. beacons). Add text as indicated. **Suggest accept in principle.** Add the following text to the end of 9.1.7: “All secure data frames that fail integrity checks are discarded.” Add the following text to the end of 9.1.8: “Under normal operations, the integrity check on the beacon provides evidence that the piconet is operating properly and that no security changes have occurred. If the integrity check on the beacon fails, the DEV is alerted to the fact that the DEV does not have its security state synchronized with the PNC.” Add the following text to the end of 9.1.10: “All secure commands that fail integrity checks are discarded.”

63 (Gilb, TR) [SEC] We still don't have a good description of what to do with commands sent or received with security on. Also need to generalize for the case of peer-to-peer security. Add description including peer-to-peer security. **Suggest accept in principle. (Note that this introduces functionality to maintain separate modes for different DEVs)** Add the following text to the beginning of 9.2.11:

“DEVs shall maintain a security state denoting whether security is required for each security relationship. If security is required for a particular security relationship, all frames transmitted to and received from another DEV in that relationship shall be protected by the keys indicated in {xref - Table 57}. A DEV may send or receive certain command frames without protection as indicated in {xref - Table 48}. If a DEV receives a frame that is not protected as required, the DEV shall discard the frame. If security is not required for a security relationship, all frames transmitted to and received from another DEV in that relationship shall be sent without security. If a DEV receives a protected frame when security is not required, the DEV shall discard the frame.

If the DEV is participating in a secure piconet, the security state for the relationship with the PNC, and consequently the broadcast key, shall be set to security required. For peer-to-peer communications, the DEV may choose to require security or not for that relationship, regardless of the security state shared with the PNC. If security is required in a peer-to-peer relationship, but the DEVs have not authenticated with each other, the group data key shall be used to protect frames between the DEVs.”

93 (Gilb, TR) [SEC/PIB] The security suite and public-key verification information have been removed from the MAC PIB. This implies that there is no accessible information that the MLME can use to perform verification on the public key. In particular, the CA certificate or hash of the public key are not available. In clause 10, each security sub-suite specifies that the ACL indicates which public-keys shall be accepted and which shall be rejected. Add public-key verification information back into the ACL and specify in each security suite how that ACL information is to be used. If there are operations to be performed by the DME to verify the public key, those should be mentioned in the security suites as well. This applies to 6.5.6 as well. **Suggest accept in principle.** The public key verification operations are performed by the DME and are hence outside the scope of the standard. Rather than adding this information to the PIB, the description of the security suites in clause 10 should be modified to reflect that this is out of scope. Make the following changes to clause 10:

In clause 10.3.1.4.3, change the paragraph in lines 18-19 to: "The certificate shall be generated using the digital signature algorithm ECDSA as specified in 10.3.1.4.4. The validation of the certificate is outside the scope of this standard."

In 10.3.2.2.2, remove step 2) and the final sentence and add the following text:

"Processing shall be aborted if the public key is not successfully extracted.

The DEV should perform additional checks such as comparing the DEV address in the ManCert to the DEV address in the authentication request or comparing the received key and ID to values stored in its ACL to verify the authenticity of the public key."

In 10.3.2.3.2, remove step 2) and the final sentence and add the following text:

"Processing shall be aborted if the public key is not successfully extracted.

The DEV should perform additional checks such as comparing the DEV address authenticated in the ImplCert with the DEV address stored in its ACL to verify that the device is authorized."

In 10.3.2.4.2, remove step 2) and the following paragraphs and add the following text:

"Processing shall be aborted if the public key is not successfully extracted.

The DEV should perform additional checks such as signature verification as specified in 10.3.1.4.3, CRL checking, validity period verification, key use checking and comparing the DEV address in the X.509 certificate with the DEV address stored in its ACL to verify that the device is authorized."

In 10.4.2.2, change the table entry for Verification of Public-Key to the following text:

"The ID and public-key received during the authentication protocol should be verified by the DME using checks such as generating the SHA-1 hash of the device address concatenated with the public-key and comparing that to the hash of the ID and public key stored in the ACL."

In 10.5.2.2, change the table entry for Verification of Public-Key to the following text:

"The ID and public-key received during the authentication protocol should be verified by the DME using checks such as generating the SHA-1 hash of the device address concatenated with the public-key and comparing that to the hash of the ID and public key stored in the ACL."

In 10.5.3.2, change the table entry for Verification of Public-Key to the following text:

"The X.509 certificate received in the authentication protocol should be verified by performing checks such as signature verification as specified in 10.5.1.7, CRL checking, validity period verification, key use checking and comparing the DEV address in the X.509 certificate with the DEV address stored in its ACL to verify that the device is authorized."

100 (Gilb, TR) [SEC] The public-key object types listed in 7.5.2.1 are not necessarily sufficient for information to verify a public-key object that is received. A new information element called ACL entry value should be added. The ACL entry value should have a type, length, DEV address and verification value. This verification value may be a SHA-1 hash, a SHA-256 hash, an X.509 CA certificate, an implicit certificate CA certificate or some other as yet undefined field. This should be flexible since in future iterations, the verification information may change form. **Suggest accept in principle.** A new command is being added to pass ACL information as specified in the resolution to 102. There are reserved types, so this will be extensible if needed. This also includes an updated table for ACL entries that include the listed values. Resolve as indicated in 102.

101 (Gilb, TR) [SEC] The disassociation request command may be sent before the device is authenticated. In addition, the table does not specify when the commands shall be sent with security turned on. The X should be removed from the "authenticated (if required)" column for the disassociation request command and a column should be added indicating which frames shall be sent with security when authenticated. Recommend allowing the probe command and piconet services command and all of the association, authentication and challenge commands not to require security and all the rest of the commands to require security when authenticated. Alternately, this information could be added to clause 9 if that is a more appropriate location. **Suggest accept in principle.** Remove the "X" from the disassociation request command. Add a column to Table 48 with the heading "Security required if authenticated" and insert an "X" in every entry except for association request, association response, authentication request, authentication response, challenge request, challenge response, probe and piconet services. Re-write the first paragraph of 7.5 to coordinate this resolution with the resolution to 63. Change first paragraph to:

"The MAC command types are listed in Table 48 and are described in the following subclauses. If the column labeled "Associated" in Table 48 is marked with an "X" then that command shall only be sent by a DEV that is associated in the piconet. If the column labeled "Authenticated (if required)" in Table 48 is marked with an "X" and authentication is required for the piconet, then that command shall only be sent by a DEV that is authenticated with the PNC in the piconet. For peer-to-peer communications, if the DEV requires security with the selected peer DEV, and the "Authenticated (if required)" column is marked with an "X", that command shall be sent to the peer DEV only if the DEVs are authenticated to each other. If the column labeled "Security required if authenticated" in Table 48 is marked with an "X" and authentication is required for the piconet, then that command shall be sent securely using the key specified in {xref - Table 57} for that command. For peer-to-peer communications, if the DEV requires security with the selected peer DEV, and thecolumn labeled "Security required if authenticated" in Table 48 is marked with an "X", then that command shall be sent securely to the peer DEV using the key specified in {xref - Table 57}."

105 (Gilb, TR) [SEC] The probe command will not function as specified in this sub-clause, as the recipient device will assume that the public-key information being sent belongs to the old PNC, not the new PNC. Change this paragraph to reference a new command that may be sent to pass ACL information to other DEVs. **Suggest accept in principle.** Reference the new command included in the resolution to 102. Change the last paragraph of 9.2.4 to the following:

"The old PNC may send ACL information about the new PNC to the other DEVs in the piconet and send ACL information about all of the authenticated DEVs in the piconet to the new PNC when it hands over the role of the PNC. This can be accomplished by sending a directed ACL information command to the new PNC with the ACL information of all of the authenticated DEVs in it and by sending a broadcast ACL information command or a directed ACL information command to each authenticated DEV with the ACL information of the new PNC. If the DME of each DEV chooses to accept this ACL information, the

authentication process between the new PNC and each authenticated DEV may proceed without any interruption of service. “

111(Gilb, TR) [SEC] Table 57: It needs to be indicated either here or in clause 7 that the probe command, piconet services command and disassociate command may be sent insecurely before authentication has taken place. Add text that indicates that the probe command, piconet services command and disassociate command may be sent insecurely before authentication has taken place. **Suggest accept in principle.** See proposed resolution to 101.

112 (Gilb, TR) [SEC] There needs to be a clear delineation between the aspects of the certificate usage that are within scope and those that are out of scope. If we are specifying the exact format of the entity certificate, it seems that the format of the CA key and the other information should be specified as well. It should be made clear what checks, if any, are performed by the MLME and what checks should be pushed up to the higher layer. **Suggest accept in principle.** Resolve as proposed in 93. As in the proposed resolution to 93, the manner in which the public key is accepted for the authentication protocol is out of scope. It is appropriate to clearly define the manner in which the certificate itself can be verified, so the definition of how to create and verify the certificate should not be removed. Instead, make changes indicated in 93 to state that devices should (instead of shall) verify the authenticity of the public key by performing the certificate checking operations. The resolution to 102 provides the ability for each security suite to define ACL entries, which may include CA certificates, which include CA keys.

116 (Gilb, TR) [SEC] Figure 168: This figure does not reflect the current version of secure beacons. This figure should be updated to match the secure beacon frame format specified in 7.3.1.2. Better still, replace this with a cross reference to the correct figure. **Suggest accept in principle.** Replace figure 168 with the following figure.

Enc Data Length *l(m)*	**Auth Data Length *l(a)***
0	$27+L_1+...+L_{n-1}$

L_{n-1}	...	L_1	13	2	2	Octets: 10
Information element-(n-1)	...	Information element-1	Piconet synch. parameters	Secure frame counter	SECID	Frame header

Figure 5—CCM input for secure beacons

117 (Gilb, E) The ECMQV security suite is not presented in a similar fashion to the NTRUEncrypt and RSA-OAEP security suites. Change this security suite to better match the other security suites and/or change the other security suites to match this security suite. Recommend combining 10.3.1.1 to 10.3.1.4 into one sub-clause (making ECMQV a level 5 heading and removing the heading for ECC certificates) and combining other sub-clauses of 10.3 for clarity. **Suggest accept in principle. Any other suggestions?** Reduce the number of sub-headings in the ECMQV security suite.

124 (Gilb, TR) [SEC] It seems that the types of security support that are listed here are limited only to the methods that are explicitly defined in the standard. There may be additional methods that should be allowed and there should be a means for vendors to indicate that there are vendor specific methods implemented. For instance, certificates that are not in the format specified here, such as X.509 certificates used in browsers, may be useful to use to provide evidence of the validity of a public key while in mode 1 or mode 2. Change the ECC and RSA X.509 certificates to be simply an X.509 certificate. The certificate indicates the method

used for authentication. **Suggest accept in principle.** Add PIC entries for supporting various kinds of ACL information in the ACL information command.

362 (Schrader, T) [SEC] The way that states are numbered and and state transitions can lead to confusion and difficulty understanding the transitions. Label states 1.0, 2.0, 3.0, ... N.0 and label state transitions as n.m to indicate a transition from state n.0 to state m.0. This impacts a lot of diagrams and text, but it would be a major improvement. Use "x" as the "any" state indicator. **Suggest accept.**

370 (Shvodian, TR) [SEC] Why can't a mode 0 PNC use the ACL? I thought this is how we got rid of mode 1. Maybe this is just an oversight. Change to "A device operating in mode 0 shall not perform any security related operations on MAC frames." **Suggest accept in principle.** Change text in 9.3.1 to the following:

“A device operating in security mode 0 shall not perform any cryptographic operations on MAC frames. While in this mode, if the MAC receives a frame with the SEC field set to 1, the MAC shall discard the frame and the MLME shall return an MLME-SECURITY-ERROR.indication to the higher layer with the ReasonCode set to UNAVAILABLE-KEY.”

431 (Gubbi, TR) [SEC] Text in ln 19:22 and Figure-41 are utterly confusing. Is this trying to tellthat Public-key objects larger than 254 octets can be fragmented and can be sent in multiple IEs that have appropriate indices? In any case, state clearly. Change the text in ln 19:22 and figure 41 to following:Text:If the public-key object is larger than 254 octets, it can be fragementedwith fragment size of 254 and sent in atmost 4 IEs. The fragmentation isonly due to the reason that IEs do not accommodate more than 254 octets.For this purpose there are four public-key object IE indices defined for this in Table-47. They are - Public-key object carrying first fragment or the entire public key object if it is less than 255 octets long - Public-key object-1 carrying second fragment, if present - Public-key object-2 carrying third fragment, if present - Public-key object-3 carrying fourth fragment, if presentWhen fragmentation is performed, the corresponding IEs shall be placedtogether in the frame carrying them and they shall appear in the order ofthe fragment they are carrying with fragment-0 appearing first.Figure: four different boxes, one for each IE, with their payload joining to form overall Public-key-object **Suggest accept in principle.** See resolution proposed in 02/399r2. It seems that perhaps it would be more flexible to simply have one public-key object IE and simply have two indication bytes at the beginning. The first indication byte would say the number in the sequence. So the first public key object IE of the extended group would have a value of 1 for that byte and the 3rd would have a value of 3. The second indication byte would be a TRUE or FALSE byte where 1 indicates that it is the last IE for this public-key object and 0 indicates that it is not the last. This simplifies the IEs and also allows for longer public-key objects if they are ever needed.