IEEE P802.15
Wireless Personal Area Networks

Project	IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs)
Title	**Draft running comment resolution**
Date Submitted	[9 July, 2004]
Source	[James P. K. Gilb] [Appairent Technologies] [16990 Via Tazon, #125, San Diego, CA 92127] Voice: [858-485-6401] Fax: [858-485-6406] E-mail: [last name at ieee dot org]
Re:	[]
Abstract	[This document is a record of comment resolutions and proposals for draft development of 802.15.3b.]
Purpose	[To provide a record of the comment resolution and proposals for draft development of 802.15.3b.]
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. Comment resolution in Portland

1.1 Wednesday, July 14, 2004

1.1.1 PNID/BSID/Open scan

Change Table 5 as indicated below.

Table 1—MLME-SCAN primitive parameters

Name	Type	Valid range	Description
~~OpenScan~~ ScanType	~~Boolean~~ Enumeration	~~TRUE, FALSE~~ OPEN, BSID, PNID, BOTH	~~Indicates whether scan is an open scan or not. Open scan is defined in 8.2.1.~~ Indicates the type of scan to be performed, either open as defined in <xref 8.2.1>, or for a specific PNID, BSID or both.

1.1.2 Catch-all reason code

CID 51

Clause 7.5.1.2 Association response

- 9-~~255~~254 -> Reserved
- 255 -> Other failure

Clause 7.5.1.3 Disassociation request

- 5-~~255~~254 -> Reserved
- 255 -> Other failure

Clause 7.5.6.2 Channel time response

- 13-~~255~~254-> Reserved
- 255-> Other failure

Clause 7.5.7.4 Remote scan response

- 3-~~255~~254-> Reserved
- 255-> Other failure

Clause 7.5.8.4 SPS configuration response

- 5-~~255~~254-> Reserved
- 255-> Other failure

1.1.3 PNID selection.

~~The PNID is chosen by the PNC when it starts the piconet and shall only be changed if the PNC detects another piconet with the same PNID on any channel~~. The PNC shall choose a PNID when it starts a piconet;

the PNID should be selected randomly. An existing piconet’s PNID shall be changed only if the PNC detects another piconet with same PNID on any channel. The same PNID may be persistent when the PNC restarts a piconet that ended without handing over control to a PNC capable DEV.

1.2 Multicast

DEV needs to leave an join a multicast group based on starting and stopping an application.

Idea:

Have DEVs send a request to the PNC to join a multicast group.

If the group does not exist, the PNC assigns a DEVID to the group.

If the group does exist, the PNC will use the existing DEVID.

The PNC responds to the DEV with this DEVID.

The PNC can also refuse the request due to lack of DEVIDs, handover in progress, size of group, too many groups, resources unavailable, other failure.

The DEV can leave a multicast group, this always succeeds, but the PNC does send a response. When the last DEV leaves the multicast group, the PNC deletes the DEVID and any CTAs.

The PNC deletes a DEV from a multicast group when it is disassociated.

This is optional for a DEV but mandatory for a PNC. The PNC is not required to support any multicast groups in any number.

In the PNC Information command the multicast group DEVIDs (McstGrpID) does not appear.

During PNC handover, the current PNC uses the Announce command to send the multicast group information in the Multicast Group IE. This has the 8 octet Multicast Address, 1 octet assigned DEVID. DEVs may request this IE from the PNC, they shall request it from another DEV, a PNC shall not request it from a DEV. The PNC may send this IE in a Announce command, a DEV shall send this IE in an Announce command.

The new PNC may delete any or all multicast groups. If so, it deletes the McstGrpID and terminates all of the streams. All DEVs in multicast groups shall rejoin the multicast group following a PNC handover. A DEV rejoins a multicast group by sending the Multicast Group Request command with the appropriate fields.

Need to add DestID to MLME_MULTICAST_SETUP, and an enumeration that indicates which one is to be filtered.

***Change Table 48 as shown in Table 2:*.**

Add the following subclause to 7.4 prior to 7.4.17.

Table 2—Information elements

Element ID hex value	Element	Subclause	Present in beacon
0x0F	Piconet Services	7.4.16	Non-beacon IE
0x10	Multicast Group	7.4.17	Non-beacon IE
0x11-0x7F	Reserved		
0x80-0xFF	Vendor Specific	7.4.18	As needed

1.2.1 Multicast Group

The Multicast Group IE is used to list the DEVs that are a member of a multicast group. The Multicast Group IE shall be formatted as illustrated in Figure 1.

octets: 1-32	1	1	8	1	1
Group IDs	Start DEVID	McstGrpID	Multicast address	Length (=11 to 42)	Element ID

Figure 1—Multicast Group information element format

The Multicast Address field is the is a 64 bit MAC address that is used for multicast traffic as defined in IEEE Std. 802.0.

The McstGrpID field contains the DEVID that has been assigned by the PNC for the address in the Multicast Address field.

The Start DEVID field indicates the DEVID that corresponds to the first bit in the Group IDs field.

The Group IDs field contains a bitmap of 1 to 32 octets in length. Each bit of the Group IDs field when set to one indicates the DEV whose DEVID is equal to the start DEVID plus the bit position in the Group ID bitmap is a member of the multicast group identified by the Multicast Address and McstGrpID fields. The bits in the Group IDs field is set to zero otherwise. The bit position 0, i.e. the first bit or lsb of the bitmap corresponds to the start DEVID.

The bits corresponding to the PNCID, UnassocID, BcstID, McstID, NbrIDs and the reserved DEVIDs, 7.2.3, shall be set to zero upon transmission by the PNC and shall be ignored upon reception.

Add the two rows in Table 3 to Table 50.

Table 3—Command types

Command type hex value b15-b0	Command name	Subclause	Associated	Secure membership (if required)
0x001D	Multicast configuration request	7.5.10.1	X	X
0x001E	Multicast configuration response	7.5.10.2	X	X

Add the following subclause at 7.5.10 or later.

1.2.2 Multicast configuration commands

1.2.2.1 Multicast configuraton request

The Multicast Configuraton Request command by a DEV to to request a McstGrpID, <xref 7.2.3>. The DestID shall be set to the PNCID. The Multicast Configuration Request command shall be formatted as illustrated in Figure 2.

octets: 8	**1**	**2**	**2**
Multicast address	Action	Length (=9)	Command type

Figure 2—Multicast configuration request command format

The Action field shall be set as indicated in Table 4.

Table 4—Action field values.

Action field value	Meaing	Description
0	Join	The request is for the DEV to join the multicast group
1	Leave	The DEV is leaving the multicast group
2-255	Reserved	

The Multicast Address field is defined in 7.4.17.

1.2.2.2 Multicast configuration response

The Multicast Configuraton Response command is used by the PNC to respond to a request for a multicast DEVID. The SrcID shall be set to the PNCID. The Multicast Configuration Response command shall be formatted as illustrated in Figure 3.

octets: 1	**1**	**8**	**2**	**2**
Reason code	McstGrpID	Multicast address	Length (=10)	Command type

Figure 3—Multicast configuration response command format

The Multicast Address field is defined in 7.4.17.

If the request for a multicast ID was successful, the McstGrpID field is the DEVID, <xref 7.2.3>, that has been assigned by the PNC for the address in the Multicast Address field. Otherwise, the McstGrpID field shall be set to zero.

The valid values of the Reason Code are:

- 0 -> Success
- 1 -> Failure, lack of DEVIDs
- 2 -> Failure, handover in progress

- 3 -> Failure, resources unavailable
- 4 -> Failure, not a valid multicast address.
- 5-254 -> Reserved
- 255 -> Other failure

Add new subsection to 8.5 as 8.5.3.

1.2.3 Multicast group configuration

Multicast addresses are defined in IEEE Std. 802-2001. Because this standard uses DEVIDs for addressing, the PNC needs to assign a DEVID to be used for a multicast address. The PNC also keeps track of all of the DEVs that request the use of a particular multicast address by maintaing a list of the their DEVIDs and the associated multicast address. A group of DEVs that have been registered with the PNC using a particular multicast address are called a multicast group.

A DEV requests a DEVID for a multicast address, called a McstGrpID, from the PNC using the Multicast Configuration Request command, 1.2.2.1, with the Multicast Address field set to the desired multicast address and the Action field set to “Join.” If a McstGrpID is not currently assigned as a DEVID for that Multicast Address and the PNC has the resources available, the PNC should assign an McstGrpID for the Multicast Address and respond to the originating DEV with the Multicast Configuration Response command, 1.2.2.2. If the request was successful, the PNC adds the originating DEV to the multicast group associated with the McstGrpID.

If the PNC has already assigned a McstGrpID for the address in the Multicast Address field and the PNC has the resources available, it shall add the originating DEV’s DEVID to the multicast group.

If the originating DEV’s request is granted, the PNC shall send the Multicast Configuration Response command to the originating DEV with the McstGrpID field set to the value assigned to that multicast address and the Reason Code set to “Success.”

If the address in the Multicast Address field does not correspond to a valid multicast address, IEEE Std. 802-2001, the PNC shall not assign a McstGrpID and shall send the Multicast Configuration Response command to the originating DEV with the McstGrpID set to zero and the Reason Code field set to “Failure, not a valid multicast address.”

If the PNC is unable to fullfill the originating DEV’s request for a McstGrpID, the PNC shall send the Multicast Configuration Response command to the originating DEV with the McstGrpID set to zero and the Reason Code field set to the appropriate value.

When a DEV no longer needs to use the multicast address, it shall send the Multicast Configuration Request command to the PNC with the Multicast Address field set to the address and the Action field set to “Leave.” When the PNC receives this command, it shall remove the DEV from the multicast group and respond with the Multicast Configuration Response command with the McstGrpID field set to zero, the Multicast Address field set to the same value as in the request command and the Reason Code field set to “Success.” The PNC shall always respond to a properly formatted Muticast Configuration Request commad with the Action field set to “Leave” with a Multicast Configuration Response command with the Reason Code set to “Success.” If the address in the Multicast Address field corresponds to an multicast group that has the originating DEV as a member, the PNC shall remove the DEV from the multicast group.

(Suggestion: Use the Association IE to indicated that the PNC is disassociating DEVs from a multicast group).

If the PNC is unable to support an existing multicast group, it shall send the Multicast Configuration Response command to the members of the multicast group with the Multicast Address field set to the address for that group, the McstGrpID set to zero and the Reason Code field set to the appropriate error value.

If a multicast group no longer has any members, either due to disassication or requests from the DEVs to leave the group, the PNC shall de-allocate the McstGrpID. A McstGrpID shall be allocated and re-used according to the rules for assigning DEVIDs in 8.3.1. A McstGrpID shall not be reported in the PNC Information command.

During PNC handover, the old PNC shall send one or more Announce commands, <xref>, to the new PNC with the Multicast Group IEs, <xref>, that correspond to the McstGrpIDs that are currently in use.

1.3 Broadcast-to-Broadcast allocations

Summary:

Non-unicast source CTAs are assigned by the PNC when it feels like it. They are allocations in the regular sense.

No one except the PNC can terminate a non-unicast source stream.

The PNC determines the access method (slotted-aloha or CSMA/CA) for a CTA when the SrcID is one of BcstID, McstID or McstGrpID and indicates this in the beacon. DEVs can request the frequency and duration of these types of CTAs with a modified channel time request command. Because the SrcID and TrgtID are both specified, the PNC can determine the type of request easily.

A request for a non-unicast source CTA can use any stream index.

The standard will have the following definitions:

— CTA - Time allocated in the superframe
— Regular CTA - non-contention based CTA (at least initially)
— Open CTA - Contention based CTA, there are multiple sources possible.
— Private CTA - Unicast SrcID that is also the DestID.
— Association CTA - Contention based CTA, only used for associating DEVs, SrcID = UnassocID.

Clause 7:

Add the following definitions for the CTRq command:

— A request with the PNCID as the DestID and the stream index set to the MCTA index indicates that the DEV is requesting a change in the frequency and/or size of allocations that the PNC provides with the DEV’s.

Clause 8:

— The PNC can allocate CTAs with the both the SrcID and DestID set to be the BcstID. These are open CTAs that use contention based channel access rather than having a single DEV with the transmit control. If the MCTA Used bit in the Piconet Synchronization Parameters field in the beaon is set to one, then the open CTAs will use slotted-aloha as the access method. Otherwise the open CTAs will use the CSMA-CA as the access method.
— DEVs cannot request termination of an open CTA. They can modify their request so that they are not requesting any time, however.

— A DEV can request that the PNC adjust the frequency and duration of open CTAs by sending a CTRq command with the destination address set to the broadcast address and the stream index set to the

Types of CTAs used in the piconet are:

Table 5—Types of CTAs in the standard

CTA type	SrcID/DestID	Stream Index	Access method(s)
CAP	N/A	N/A	Uses CSMA/CA, not a real CTA, but it is assigned time in the superframe.
Regular CTA	Any valid single DEVID[a]	A regular stream index	TDMA with transmit control transfer
Regular MCTA	PNCID and any valid single DEVID	MCTA stream index	TDMA with transmit control transfer. This is the same functionality as a regular CTA.
Association MCTA	UnassocID/PNCID	MCTA stream index	Slotted aloha[b]
Private CTA	Both IDs are any valid single DEVID	A regular stream index	Not defined by PNC, handled by by DEV that has control of the CTA
Open MCTA	BcstID/BcstID	MCTA stream index	Slotted aloha or CSMA/CA as index the PNC in the beacon.
Open CTA	BcstID/BcstID	Asynchronous stream index	Slotted aloha or CSMA/CA as index the PNC in the beacon

[a] A single DEVID is a DEVID that corresponds to a single physical devices
[b] Could we let a CSMA/CA PNC uses these and have them as CSMA/CA?

Different allocations can be requested with the CTRq command. These are listed in Table 6.

Table 6—Interpetation of Parameters in a Channel Time Request command

DestID	Stream Index	CTA Type	Description
Any DEVID	Unassigned	Regular CTA	New CTA
Any DEVID	Assigned stream index	Regular CTA	Modify or terminate existing CTA
Any DEVID	Asynchronous stream index	Regular CTA	Create or modify asynchronous CTA
Same as SrcID	Unassigned	Private CTA	New private CTA
Same as SrcID	Assigned stream index	Private CTA	Modify or terminate existing private CTA
DEVID different than SrcID	Stream index previously assigned to private CTA	Private CTA	Handover control of the private CTA to the DEV indicated in the DestID field
UnassocID	Any	N/A	Not allowed in a request, only the PNC assigns association CTAs
BcstID	MCTA stream index	Open CTA	Modify request for an open MCTA, PNC takes this as a suggestion
BcstID	Channel forward (CF) stream index (=FF)	Open CTA	Modify request for open CTAs, PNC takes this as a suggestion.
PNCID	MCTA stream index	Regular MCTA	Modify request for DEV to PNC CTAs, PNC takes this as a suggestion

2. Proposals

This section contains proposals for solutions. These proposals have not necessarily been approved or disapproved.

2.1 2-way CTAs

Need to add a new command, suggested format is as follows:

Added the following to 7.5.9 as 7.5.9.3

2.1.0.1 Relinquish CTA time command

The Relinquish CTA Time command enables a DEV to release a period of time in a CTA to be used by another DEV, <xref 8.4.3.3>. The ACK Policy field in the MAC header shall be set to no-ACK. The Relinquish CTA Time command shall be formatted as illustrated in Figure 4.

octets: 2	**2**	**2**
Relinquish end time	Length (=2)	Command type

Figure 4—Relinquish CTA command format

The Relinquish End Time field indicates the time in μs measured from the beginning of the superframe by which the DEV that is the DestID of this command will no longer be able to transmit in the current CTA. The rules for using this command are specified in <xref 8.4.3.3>.

Add the following text as a new subclause, 8.4.3.3 (subsequent subclauses will be renumbered) or as the last paragraphs in 8.4.3.2.

2.1.0.2 Relinquishing CTA time to another DEV

The PNC gives transmit control to the DEV that is the SrcID of a CTA for the duration of the CTA. The DEV that has transmit control in a CTA may, subject to the restrictions in this subclause, relinquish a soem or all of the remaining time in a CTA to another DEV. The DEV that relinquishes the channel time is referred to as the originating DEV while the DEV which is given the transmit control of the time in the CTA is referred to as the target DEV. The DEV that is the SrcID of the CTA begins the CTA with transmit control for the CTA.

The originating DEV relinquishes transmit control of the time in the CTA to the target DEV by sending the Relinquish CTA Time command to the target DEV with the Relinquish End Time field set appropriately, <7.5.x.x>. The originating DEV shall have control over access to the CTA when it has received the command.

The originating DEV may relinquish any portion of the time in the CTA up to the end of the CTA. If the value of the Relinquish End Time field is less than the end time for the CTA, the target DEV is given transmit control for only a portion of the CTA. Transmit control returns to the originating DEV either when the time in the Relinquish End Time field occurs or when it receives a Relinqush CTA Time command from the target DEV prior to the time indicated in the Relinquish End Time field.

(note: Add note about guard time).

If the value of the Relinquish End Time field is equal to or greater than the end time for the CTA, the target DEV has been given transmit control for the remainder of the CTA. Regardless of the value of the Relinquish End Time field, transmit control returns to the PNC at the end of the CTA.

(Note the originating DEV shall not set it to longer than Relinquish End Time. If a DEV has not received a beacon, it shall not send the Relinquish End Time command.)

If target DEV has been given transmit control for only a portion of a CTA, it shall not hand over transmit control to any other DEV in the piconet. In this case, the target DEV may return transmit control to the originating DEV using the Relinquish CTA Time command. In this case, the originating DEV ignores the value of the Relinquish End Time field.

If a DEV has been given transmit control for the remainder of the CTA, it may handover transmit control to another DEV in the piconet.

(note: new field here, modify figure 5 with new field.).

If the destination DEV has data that it needs to send, it may use the time provided by the source DEV to send data frames, as illustrated in Figure 5. The destination DEV that has transmit control in a CTA is not required to use it only for communcation with the source DEV. It may send frames to any device in the piconet, but it should only send frames if it determines that the destination of its frames will be listening during that time.

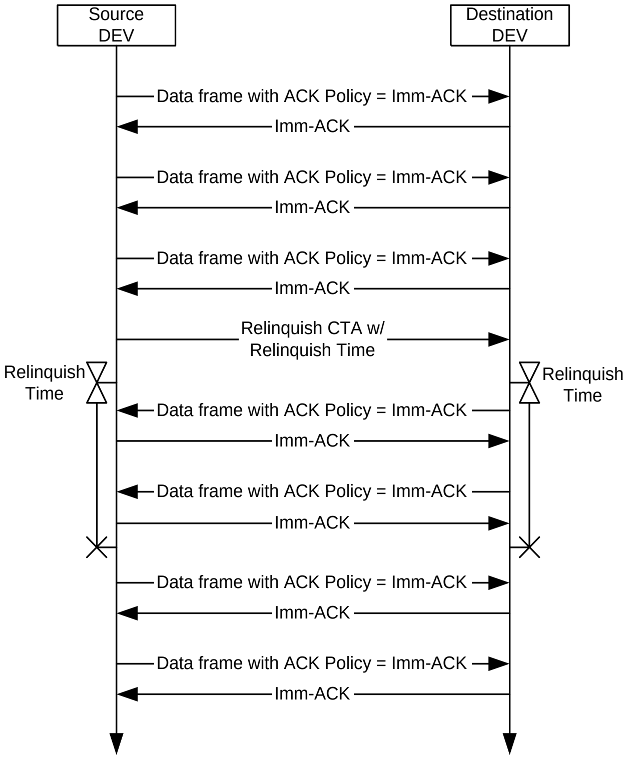
Source DEV
Destination DEV
Data frame with ACK Policy = Imm-ACK
Imm-ACK
Data frame with ACK Policy = Imm-ACK
Imm-ACK
Data frame with ACK Policy = Imm-ACK
Imm-ACK
Relinquish CTA w/ Relinquish Time
Relinquish Time
Relinquish Time
Data frame with ACK Policy = Imm-ACK
Imm-ACK
Data frame with ACK Policy = Imm-ACK
Imm-ACK
Data frame with ACK Policy = Imm-ACK
Imm-ACK
Data frame with ACK Policy = Imm-ACK
Imm-ACK

Figure 5—Message sequence chart for relinquishing CTA time when the destination DEV has data to send.

If the destination DEV does not have frames to send, then it may either hand the transmit control back to the source DEV using the Relinquish CTA Time command <xref 7.5.x.x> or control will return to the source DEV when the relinquish time has expired.

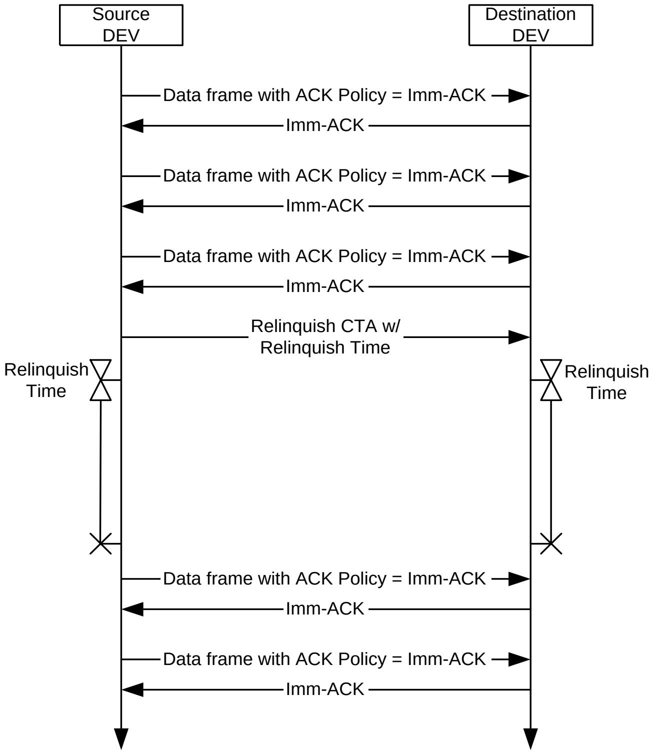
Source DEV
Destination DEV
Data frame with ACK Policy = Imm-ACK
Imm-ACK
Data frame with ACK Policy = Imm-ACK
Imm-ACK
Data frame with ACK Policy = Imm-ACK
Imm-ACK
Relinquish CTA w/ Relinquish Time
Relinquish Time
Relinquish Time
Data frame with ACK Policy = Imm-ACK
Imm-ACK
Data frame with ACK Policy = Imm-ACK
Imm-ACK

Figure 6—Message sequence chart for relinquishing CTA time when the destination DEV does not have data to send.

(Need one with a Relinquish CTA time command responding and change timer finishes to be correct.)

2.2 DME-PAL SAP

Turns on and off facility, used to indicate the time a beacon arrived.

DME-BEACON-EVENT.request

DME-BEACON-EVENT.confirm

DME-BEACON-EVENT.indication

Start, stop and join

DME-START-PICONET

DME-DISASSOCIATE (how do we handle mulitple requests for join?)

DME-SCAN

DME-ASSOCIATE (how do we handle multiple requests for join?)

DME-RESET

DME-EXIT?

DME-PNC-HANDOVER

DME-NEW-PNC

Channel time

DME-CREATE-STREAM

DME-MODIFY-STREAM

DME-TERMINATE-STREAM

ASIE facility

DME-CREATE-ASIE.request, .indication, .response, .confirm

DME-ASIE.indication

Piconet Services

DME-PICONET-SERVICES.request, indication

Association

DME-DEV-ASSOCIATION-INFO (this handles PNC information command as well.)

Misc

DME-CHANNEL-STATUS.request, .indication., .response, .confirm.

DME-PICONET-PARM-CHARGE (channel, super duration, BSID)

2.3 Proposed Security Annex

Annex A

(informative)

Informal security analysis

A.1 Introduction

A useful number for this discussion is the number of μs in a year.

$$1\text{year} = 365 \times 24 \times 60 \times 60 \times 10^6 = 3.1536 \times 10^{13} \mu s \cong 2^{45} \mu s$$

A.2 Key usage

In general, a 128 bit AES key used in piconet should not be used more than 2^{64} times to produce an IC or to encrypt a frame. If a DEV sends a frame encrypted by a key once every microsecond, it would send approximately 2^{45} frames every year. Thus, to avoid security problems, an implementation should change its management keys at least once every 2^{19} = 524,288 years. More conservative implementations that are concerned with security should change management keys at least once every millenium.

Even if the DEV is able to send an encrypted frame once every nanosecond, it would transmit approximately 2^{45} frames every year and so the key should be changed at least once every 585 years. Of course, after 585 years, computation power will have increased dramatically and 128 bit AES keys likely will no longer be considered to be secure.

A.3 Replay attacks

The 802.15.3 symmetric key encryption suite

2.4 Montag in Berlin, 13.09.04

Database is 15-04-0334-05

CID 81 - Accept in principle

7.5.6.1 ????

Change xx paragraph as indicated

For instance, in the case where the CTA Rate Type field is set to zero, a value indicating a super-rate CTA request, and the CTA Rate Factor field contains a value N greater than zero, the requesting DEV is requesting super-rate CTAs from the PNC. If these super-rate CTAs, are allocated by the PNC, they will appear N times per superframe. A PNC shall support at least 8 CTAs per stream in the same superframe. The CTA Rate Type field set to zero and the CTA Rate Factor set to zero shall be reserved.

CID 77 and 78: Change to clarify 7.5.6.1. Tabled until Dienstag.

CID 79: Wanted to bring this to the communities attention. Take to the email reflector and off site people today for discussion.

What if the PNC can’t allocate the CTA Rate Factor, does the PNC respond with failure or let the DEV see the result and ternimate if the DEV doesn’t like the allocation.

Is the PNC curently required to notify a DEV if the PNC is no longer able to provide the minimum number of TUs in the allocation?

CID 80: Accept in principle

Make an exception in 7.4.1: "The exception is null CTAs (8.5.1.3), which shall be the last in the sequence of CTABs in the beacon."

Delete last 2 octets in Dly-ACK, no CID submitted. Table until Mittwoch, discuss off-line.