<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>ASN.1 coding for AAI-PKM-REQ/RSP message over IEEE 802.16.1a</td>
<td></td>
</tr>
<tr>
<td>Date Submitted</td>
<td>2012-05-04</td>
<td></td>
</tr>
<tr>
<td>Source(s)</td>
<td>Eunkyung Kim, Sungcheol Chang, Won-Ik Kim, Seokki Kim, Sungkyung Kim, Miyoung Yun, Hyun Lee, Chulsik Yoon, Jaesun Cha, Soojung Jung, Anseok Lee, Wooram Shin, Kwangjae Lim</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ETRI</td>
<td></td>
</tr>
<tr>
<td>Voice</td>
<td>+82-42-860-5415</td>
<td></td>
</tr>
<tr>
<td>E-mail</td>
<td><a href="mailto:ekkim@etri.re.kr">ekkim@etri.re.kr</a></td>
<td></td>
</tr>
<tr>
<td>Re:</td>
<td>“IEEE 802.16-12-271,” in response to Letter Ballot Recirc #38a on P802.16.1a/D2</td>
<td></td>
</tr>
<tr>
<td>Abstract</td>
<td>ASN.1 coding of AAI-PKM-REQ/RSP message on GRIDMAN Draft Standard</td>
<td></td>
</tr>
<tr>
<td>Purpose</td>
<td>To discuss and adopt the proposed text in the draft amendment document on GRIDMAN</td>
<td></td>
</tr>
<tr>
<td>Notice</td>
<td>This document does not represent the agreed views of the IEEE 802.16 Working Group or any of its subgroups. It represents only the views of the participants listed in the “Source(s)” field above. It is offered as a basis for discussion. It is not binding on the contributor(s), who reserve(s) the right to add, amend or withdraw material contained herein.</td>
<td></td>
</tr>
<tr>
<td>Copyright Policy</td>
<td>The contributor is familiar with the IEEE-SA Copyright Policy</td>
<td></td>
</tr>
<tr>
<td>Patent Policy and Procedures</td>
<td>The contributor is familiar with the IEEE-SA Patent Policy and Procedures:</td>
<td></td>
</tr>
<tr>
<td></td>
<td><a href="http://standards.ieee.org/guides/bylaws/sect6-7.html#6">http://standards.ieee.org/guides/bylaws/sect6-7.html#6</a> and</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Further information is located at <a href="http://standards.ieee.org/board/pat/pat-material.html">http://standards.ieee.org/board/pat/pat-material.html</a> and <a href="http://standards.ieee.org/board/pat">http://standards.ieee.org/board/pat</a>.</td>
<td></td>
</tr>
</tbody>
</table>
1. Introduction

This document provides ASN.1 coding of AAI-PKM-REQ/RSP messages.

2. References


3. Proposed Text on the IEEE 802.16.1a Amendment Draft Standard

[-------------------------------------Start of Text Proposal---------------------------------------------------]

[Remedy: Add the following text in line #51 in page 223 on P802.16.1a/D2]

.........

- Group Configuration
- Group Configuration
AAI-GRP-CFG ::= SEQUENCE {
    deletionFlag ENUMERATED {
        flowAdded, flowDeleted
dlULIndicator ENUMERATED {
    dlAllocation,
    ulAllocation
},
flowID FID,
burstSize INTEGER (0..31) OPTIONAL,
graInfo CHOICE {
    graInfoForDeletededFlow NULL,
    graInfoForAddedFlow GroupResourceAllocInfo
},
...

-- Security Messages
PKMID ::= INTEGER (0..255)
AKID ::= BIT STRING (SIZE (64))
SAID ::= INTEGER (0..255)
KeyLifetime ::= INTEGER (0..4294967295)
CounterTEK ::= INTEGER (0..65535)
EKS ::= INTEGER (0..3)
Nonce ::= BIT STRING (SIZE (64))
MulticastNonce ::= BIT STRING (SIZE (128))
TimeStamp ::= INTEGER (0..4294967295)
EncryptedDMK ::= BIT STRING (SIZE (1024))
Signature ::= BIT STRING (SIZE (1024))
Certificate ::= BIT STRING (SIZE (1024))
PKM-ReauthRequest ::= SEQUENCE {
    cmacIndicator CMACI,
    ...
}
PKM-EAPTransfer ::= SEQUENCE {
    eapPayload OCTET STRING (SIZE (1..1400)),
    ...
}
PKM-KeyAgreementMsg1 ::= SEQUENCE {
    nonceABS Nonce,
    akID AKID,
    keyLifeTime KeyLifetime,
    cmacIndicator CMACI,
    ...
}
PKM-KeyAgreementMsg2 ::= SEQUENCE {
    nonceABS Nonce,
    nonceAMS Nonce,
    akID AKID,
    securityNegoParameters SecurityNegotiationPara OPTIONAL,
    cmacIndicator CMACI,
    ...
}
PKM-KeyAgreementMsg3 ::= SEQUENCE {
    nonceABS Nonce,
    nonceAMS Nonce,
supportingSAs SupportingSAs OPTIONAL,
securityNegoParameters SecurityNegotiationPara OPTIONAL,
cmacIndicator CMACI,
...

PKM-TEKRequest ::= SEQUENCE {
  said SAID,
tekRefreshFlag ENUMERATED {
    secondTEKUpdate,
    firstTEKUpdate
  } OPTIONAL,
  cmacIndicator CMACI,
...
}

PKM-TEKReply ::= SEQUENCE {
  said SAID,
  counterTEK CounterTEK,
  eks EKS,
  cmacIndicator CMACI,
...
}

PKM-TEKInvalid ::= SEQUENCE {
  said SAID,
  cmacIndicator CMACI,
...
}

-- for HR-Network

Peer-KeyAgreementMsg1 ::= SEQUENCE {
  keyAgreementType CHOICE {
    preSharedKey PreSharedKey1,
    pki PKI1,
  },
  cmacIndicator CMACI,
...
}

Peer-KeyAgreementMsg2 ::= SEQUENCE {
  keyAgreementType CHOICE {
    preSharedKey PreSharedKey2,
    pki PKI2,
  },
  cmacIndicator CMACI,
...
}

Peer-KeyAgreementMsg3 ::= SEQUENCE {
  keyAgreementType CHOICE {
    preSharedKey PreSharedKey3,
    pki PKI3,
  },
  cmacIndicator CMACI,
...
}

PKM-MulticastKeyRequest ::= SEQUENCE {
  multicastGroupID MulticastGroupID,
  fid FID,
  cmacIndicator CMACI,
PKM-MulticastKeyReply ::= SEQUENCE {
  multicastGroupID  MulticastGroupID,
  fid             FID,
  mcNonce         MulticastNonce,
  counterMtek      CounterTEK,
  meks            EKS,
  cmacIndicator   CMACI.
}

SecurityNegotiationPara ::= SEQUENCE {
  sizeOfICV ENUMERATED {
    thirtyTwoBits,
    sixtyFourBits
  },
  windowSize     INTEGER (0..65535)
}

SupportingSAs ::= BIT STRING {
  nullSASupported (0),
  said1Supported  (1),
  said2Supported  (2)
} (SIZE (3))

PreSharedKey1 ::= SEQUENCE {
  nonceHRMS1      Nonce,
  dakID           AKID,
  keyLifetime     KeyLifetime
}

PKI1 ::= SEQUENCE {
  timestampHRMS1  TimeStamp,
  nonceHRMS1      Nonce,
  macAddressHRMS2  MACAddress,
  macAddressHRMS1  MACAddress,
  sigHRMS1        Signature,
  certificateHRMS1 Certificate
}

PreSharedKey2 ::= SEQUENCE {
  nonceHRMS1      Nonce,
  nonceHRMS2      Nonce,
  dakID           AKID,
  securityNegoParameters  SecurityNegotiationPara
}

PKI2 ::= SEQUENCE {
  timestampHRMS2  TimeStamp,
  nonceHRMS2      Nonce,
  macAddressHRMS1  MACAddress,
  macAddressHRMS2  MACAddress,
  nonceHRMS1      Nonce,
  encryptedDMK    EncryptedDMK,
  sigHRMS2        Signature,
  certificateHRMS2 Certificate
}

PKI3 ::= SEQUENCE {
  nonceHRMS2      Nonce,
  macAddressHRMS2  MACAddress,
  macAddressHRMS1  MACAddress
}
-- Privacy Key Management Request

-- Privacy Key Management Request

AAI-PKM-REQ ::= SEQUENCE {
    pkmid PKMID,
    pkmMessage CHOICE {
        reauthRequest PKM-ReauthRequest,
        eapTransfer PKM-EAPTransfer,
        keyAgreementMsg2 PKM-KeyAgreementMsg2,
        tekRequest PKM-TEKRequest,
        tekInvalid PKM-TEKInvalid,
        peerKeyAgreementMsg2 Peer-KeyAgreementMsg2,
        multicastKeyRequest PKM-MulticastKeyRequest,
        ...
    }
}

-- Privacy Key Management Response

-- Privacy Key Management Response

AAI-PKM-RSP ::= SEQUENCE {
    pkmid PKMID,
    pkmMessage CHOICE {
        eapTransfer PKM-EAPTransfer,
        keyAgreementMsg1 PKM-KeyAgreementMsg1,
        keyAgreementMsg3 PKM-KeyAgreementMsg3,
        tekReply PKM-TEKReply,
        tekInvalid PKM-TEKInvalid,
        peerKeyAgreementMsg1 Peer-KeyAgreementMsg1,
        peerKeyAgreementMsg3 Peer-KeyAgreementMsg3,
        multicastKeyReply PKM-MulticastKeyReply,
        ...
    }
}

---------------------End of Text Proposal--------------------------------------