PANA Working Group
Internet-Draft
Expires: December 25, 2006

Y. El Mghazli
Alcatel
Y. Ohba
Toshiba
J. Bournelle
GET/INT

June 23, 2006

SNMP usage for PAA-EP interface
draft-ietf-pana-snmp-06

Status of this Memo

Copyright Notice

Abstract

The PANA Authentication Agent (PAA) does not necessarily act as an Enforcement Point (EP) to prevent unauthorized access or usage of the network. When a PANA Client successfully authenticates itself to the PAA, EP(s) (e.g., access routers) will need to be suitably notified. The PANA working group recommends the use of Simple Network

Management Protocol Version 3 (SNMPv3) to deliver the authorization information to one or more EPs when the PAA is separated from EPs.

The present document provides the necessary information and extensions needed to use SNMPv3 as the PAA-EP protocol.

Table of Contents

1. Terminology and Definitions

PANA: Protocol for Carrying Authentication for Network Access.

PaC (PANA Client):

The client side of the protocol that resides in the host device, which is responsible for providing the credentials to prove its identity for network access authorization. A PaC is responsible for requesting network access and engaging in authentication process using the PANA protocol.

DI (Device Identifier):

The identifier used by the network as a handle to control and police the network access of a client. Depending on the access technology, this identifier might contain any of IP address, link-layer address, switch port number, etc. of a connected device.

PAA (PANA Authentication Agent):

The access network (server) side entity of the PANA protocol. A PAA is in charge of interfacing with the PaCs for authenticating and authorizing them for the network access service. To this end, the PAA verifies the credentials provided by a PANA client and grants network access service to the device associated with the client and identified by a DI.

The PAA is also responsible for updating the access control state (i.e., filters) depending on the authorization. The PAA communicates the updated state to the enforcement points in the network. When the PAA and the EP are separated, a protocol is required to carry the authorized client attributes from the PAA to the EP. SNMPv3 is used to this end.

EP (Enforcement Point):

A node on the access network where per-packet enforcement policies are applied on the inbound and outbound traffic of client devices. The EP uses non-cryptographic or cryptographic filters to selectively allow and discard data packets. These filters may be applied at the link-layer or the IP-layer. An EP learns the attributes of the authorized clients (DI and optionally cryptographic keys) from the PAA.

Authentication Server (AS):

The server implementation that is in charge of verifying the credentials of a PaC that is requesting the network access service. The AS receives requests from the PAA on behalf of the PaCs, and responds with the result of verification together with the authorization parameters (e.g., allowed bandwidth, IP configuration, etc). The AS might be hosted on the same node as the PAA, on a dedicated node on the access network, or on a central server somewhere in the Internet.

2. Introduction

2.1. PAA/EP separation context

PANA enables access control by identifying legitimate clients and generating filtering information for access control mechanisms. [I-D.ietf-pana-framework] defines a general AAA and access control framework. The PANA protocol itself provides client authentication and authorization functionality for securing network access. Access control ensures that only authenticated and authorized clients can gain access to the network.

Access control can be achieved by placing EPs (Enforcement Points) in the network for policing the traffic flow. EPs should prevent data traffic from and to any unauthorized client unless it's either PANA or one of the other allowed traffic types (e.g., ARP, IPv6 neighbor discovery, DHCP, etc.).

Figure 1 below illustrates the functional entities and the interfaces (protocols, APIs) among them.

```
                                           RADIUS/
                                           Diameter/
     +-----+       PANA        +-----+     LDAP/ API    +-----+
     | PaC |<----------------->| PAA |<---------------->| AS  |
     +-----+                   +-----+                  +-----+
        ^                          ^
        |                          |
        |         +-----+          |
   IKE/ +-------->| EP  |<--------+ SNMP/ API
4-way handshake   +-----+
```

Figure 1: PANA Functional Model

Some of the entities may be co-located depending on the deployment scenario.

The EP on the access network allows general data traffic from any authorized PaC, whereas it allows only limited type of traffic (e.g., PANA, DHCP, router discovery) for the unauthorized PaCs. This ensures that the newly attached clients have the minimum access service to engage in PANA and get authorized for the unlimited service.

If the PaC is authorized to gain the access to the network, the PAA also sends the PaC-specific attributes (e.g., IP address, cryptographic keys, etc.) to the EP by using SNMP. The EP uses this information to enforce policy rules allowing data traffic from and to

the PaC to pass through.

In case a cryptographic access control needs to be enabled after the PANA authentication, a secure association protocol runs between the PaC and the EP. The PaC should already have the input parameters to this process as a result of the successful PANA exchange. Similarly, the EP should have obtained them from the PAA via SNMP. Secure association exchange produces the required security associations between the PaC and the EP to enable per-packet protection.

Finally data traffic can start flowing from and to the newly authorized PaC.

In this document, it is assumed that PAA and EP are pre-configured to be able to communicate each other with a required security association. In case where dynamic discovery is needed for PAA and EP to know each other's address, then SNMPv3 engine-id discovery could be used. Such dynamic discovery is out of scope of this document.

2.2. Scope

Section 3 gives references for the SNMP framework.

Section 4 provides a general statement with regards to the applicability of SNMP as the PAA-EP protocol.

IPsec configuration MIB modules were found to have general applicability and varying levels of re-usability for PANA EP authorization configuration using SNMP. Section 5 details the applicability of this MIB set to the EP configuration and Section 6 defines some additional PANA-specific objects that extend the IPsec SPD-MIB module [I-D.ietf-ipsp-spd-mib] in order to entirely satisfy the PAA-EP interface requirements.

In the same manner, RTFM Meter MIB module was found to have general applicability of re-usability for PANA EP accounting configuration using SNMP. Section 7 details the applicability of this MIB to the EP configuration.

Finally, Section 9 addresses the security considerations.

3. The Internet-Standard Management Framework

For a detailed overview of the documents that describe the current Internet-Standard Management Framework, please refer to section 7 of [RFC3410].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. MIB objects are generally accessed through the Simple Network Management Protocol (SNMP). Objects in the MIB are defined using the mechanisms defined in the Structure of Management Information (SMI). This memo specifies a MIB module that is compliant to the SMIv2, which is described in STD 58 [RFC2578], STD 58 [RFC2579] and STD 58 [RFC2580].

4. SNMP Applicability with the PANA framework

This section provides a general statement with regards to the applicability of SNMP as the PAA-EP protocol. This analysis of SNMP is specific to SNMPv3, which provides the security required for PANA usage. SNMPv1 and SNMPv2c would be inappropriate for PANA since they have been declared Historic, and because their messages have only trivial security.

4.1. SNMPv3 General applicability

SNMPv3 is that it is a mature, well understood protocol, currently deployed in various scenarios, with mature toolsets available for SNMP managers and agents.

Application intelligence is captured in MIB modules, rather than in the messaging protocol. MIB modules define a data model of the information that can be collected and configured for a managed functionality. The SNMP messaging protocol transports the data in a standardized format without needing to understand the semantics of the data being transferred. The endpoints of the communication understand the semantics of the data.

Partly due to the lack of security in SNMPv1 and SNMPv2c, and partly due to variations in configuration requirements across vendors, few MIB modules have been developed that enable standardized configuration of managed devices across vendors. Since monitoring can be done using only a least-common-denominator subset of information across vendors, many MIB modules have been developed to provide standardized monitoring of managed devices. As a result, SNMP has been used primarily for monitoring rather than for configuring network nodes.

SNMPv3 builds upon the design of widely-deployed SNMPv1 and SNMPv2c versions. Specifically, SNMPv3 shares the separation of data modeling (MIBs) from the protocol to transfer data, so all existing MIBs can be used with SNMPv3. SNMPv3 also uses the SMIv2 standard, and it shares operations and transport with SNMPv2c. The major difference between SNMPv3 and earlier versions is the addition of strong message security and controlled access to data.

SNMPv3 uses the architecture detailed in [RFC3411], where all SNMP entities are capable of performing certain functions, such as the generation of requests, response to requests, the generation of asynchronous notifications, the receipt of notifications, and the proxy-forwarding of SNMP messages. SNMP is used to read and manipulate virtual databases of managed-application-specific operational parameters and statistics, which are defined in MIB

modules.

4.2. Compliancy of SNMP against the PANA requirements

The following sections detail how the PAA-EP protocol requirements are fully supported by SNMP:

4.2.1. Authorization Consideration

This section discusses PAA-EP communication in terms of authorization aspects.

Binary Authorization:

Filtering rules to be installed on EP generally include a device identifier of PaC, and also some cryptographic keying materials (e.g., IKE [RFC2409] pre-shared key) when cryptographic data traffic protection is needed. Each keying material is uniquely identified with a keying material name (e.g., ID_KEY_ID in IKE) and has a lifetime for key management, accounting, access control and security reasons in general.

PANA authorization management can be modeled in a way that is consistent with existing standard MIB modules, this is detailed in Section 5. Additional PANA-specific objects may be needed and are defined in Section 6.

Profile-based authorization:

In addition to the device identifier and keying material, the PAA may provide the EP with additional authorization information. For instance, a user may be authorized to access the network within a given class of service or for a maximum amount of units. The type of units can be time (e.g., authorization lifetime), volume (e.g., the number of incoming and/or outgoing packets and/or bytes), service specific or money depending on the type of service event [I-D.ietf-aaa-diameter-cc].

This type of authorization might also require that the communicating pair of PAA and EP to detect a dead or rebooted peer in order to avoid possible inaccurate accounting. This aspect is discussed in Section 4.2.6.

In any case, even if it is very likely to occur between the PAA and EP, this kind of profile-based authorization is beyond the scope of the PANA working group. Hence, the specification of the MIB modules (existing or not) necessary to provide such policy information is outside the scope of the present document, which

deals only with the so-called binary authorization.

4.2.2. PAA-EP relation

A number of deployment options are envisaged within the PANA framework. See [I-D.ietf-pana-framework] for further details.

The SNMP framework [RFC3410] supports one-to-many, many-to-one, and many-to-many relationships between the SNMP managers (PAAs) and agents (EPs).

4.2.3. Secure Communication

SNMPv3 includes the User-based Security Model (USM, [RFC3414]), which provides authentication, confidentiality, and integrity.

Additionally, USM has specific built-in mechanisms for preventing replay attacks including unique protocol engine IDs, timers and counters per engine and time windows for the validity of messages.

See [RFC3410] for the security features provided by the SNMPv3 framework.

4.2.4. Notification of PaC presence

The PaC may also choose to start sending packets before getting authenticated. In that case, the network should detect this and the PAA must send an unsolicited PANA-Start-Request message to the PaC. The EP is the node that can detect such activity. In case they are separate, there needs to be an explicit message to prompt the PAA.

Such a presence notification is done by using the SNMP notification operation. See Section 6 and Section 6.2 for details on how new and existing SNMP objects provide this feature.

The presence notification may contain both an IP address and a link-layer address, or only one of them.

A presence notification without containing a link-layer address may be generated where access control is performed using IPsec [I-D.ietf-pana-ipsec].

A presence notification without containing an IP address may be generated when a link-layer frame that does not contain an IP address is used as the notification trigger. If the PAA that received such a notification from the EP has already a PANA session associated with the link-layer address, it may use the notification as a trigger to install filtering information to the EP. This can happen if the PaC

is roaming from one EP to another under the same PAA.

4.2.5. Accounting Consideration

Since authentication and authorization are closely related to accounting in many cases, accounting aspects need to be considered in the PAA-EP protocol.

The PAA device acts as an accounting client of a AAA protocol. The PAA collects accounting information from the EP(s) it controls, and sends the gathered data to the accounting server by using the AAA protocol.

PANA accounting management can be done using RTFM (Real-Time Flow Measurement) standard MIB module [RFC2720], this is detailed in Section 7.

The authentication/authorization session identifier of a AAA protocol is used by the accounting server to associate accounting information with a particular authentication/authorization session to calculate bills. The authentication/authorization session identifier may or may not be the same as the PANA session identifier and it is the responsibility of the PAA to organize the correlation between the meters/filters at the EP and the PANA/AAA sessions at the PAA.

The communicating pair of the PAA and the EP might need to detect a rebooted peer to avoid possible inaccurate accounting. This aspect is discussed in Section 4.2.6.

4.2.6. Peer Liveness Test and Rebooted Peer Detection

PAA-EP protocol implementations need to be stateful, when considering the authorization and accounting aspects as described in the previous sections. The stateful nature provides the functionality to detect a dead or rebooted peer in a timely fashion. On the other hand, this does not mean that the PAA-EP protocol itself needs to be stateful. For example, an SNMP entity (i.e., an SNMP engine plus SNMP applications) can generate SNMP queries on a particular MIB at an interval short enough to perform peer liveness test and rebooted peer detection.

Also, the peer liveness test and rebooted peer detection need to be performed securely.

When SNMPv3 is used as the PAA-EP protocol, the SNMP management framework supports snmpEngineBoots MIB [RFC3411]. By periodically sending SNMP query to the peer to check the current value of this MIB with the use of SNMP Security Subsystem, it is possible for an SNMP

entity to securely perform peer liveness test and rebooted peer detection between PAA and EP.

5. Applicability of IPsec configuration MIBs

This section details the applicability of existing IPsec configuration MIB modules to the EP configuration. These were found to have general applicability and a fair level of re-usability for the PANA EP configuration:

IPSec Security Policy Database (SPD) Configuration MIB:

[I-D.ietf-ipsp-spd-mib] defines a MIB module (IPSEC-SPD-MIB) for configuration of an IPSec Security Policy Database (SPD). No IPsec or IKE specific actions are defined within this document.

IPsec Security Policy IKE Action MIB:

[I-D.ietf-ipsp-ikeaction-mib] defines a MIB module (IPSEC-IKEACTION-MIB) for configuration of an IKE action within the IPsec SPD.

IPsec Security Policy IPsec Action MIB:

[I-D.ietf-ipsp-ipsecaction-mib] defines a MIB module (IPSEC-IPSECACTION-MIB) for configuring IPsec actions within the IPsec SPD.

The EP enforces binary authorization by filtering data traffic on the basis of the Device Identifier (DI) of the PaC. The PAA must provision its EPs with DI-based filters in order to control and police the network access of a PaC. According to the definition of the Device Identifier in [RFC4058], such filters - depending on the access technology - might be either a IPv4/6 address or a link-layer address of a connected device.

Do note also that a keying material might be provisioned. The particular case where access control is performed using IPsec is specified in [I-D.ietf-pana-ipsec]. The configuration aspects in this case are detailed in Section 5.2.

5.1. General IP Access Control

The IPsec SPD MIB module (SPD-MIB) is designed to configure an IPsec security policy database in a policy and rule oriented fashion. This module is divided into 3 portions (Rules, Filters, Actions). Specifically, SPD-MIB provides a generic mechanism for performing packet processing based on a rule set.

The policy-based packet filtering and the corresponding execution of actions is of a more general nature than for IPsec configuration

only, such as for configuration of a firewall. Rules within the IPsec SPD-MIB are generic and simply bind a filter to an action. Filters provided within the SPD-MIB itself are numerous and fairly complete for most common packet filtering usage but externally defined filters are supported.

For IPv4/v6 address-based filters provisioning, the DIFFSERV-MIB module defined in [RFC3289] provides means to filter the traffic based on the IP header information. DiffServ Multi-field Classifier table provides such facilities: one can define the various tests that are used when evaluating a given IP packet. The various tests definable in this table are as follows:

- IP source address prefix, including host, CIDR Prefix, and "any source address"

- IP destination address prefix, including host, CIDR Prefix, and "any destination address"

- IPv6 Flow ID

- IP protocol or "any"

- TCP/UDP/SCTP source port range, including "any"

- TCP/UDP/SCTP destination port range, including "any"

- Differentiated Services Code Point

The results of each test are ANDed together to produce the result of the entire filter.

The actions encapsulated within the SPD-MIB module are basic drop/accept actions. These are sufficient to perform EP binary authorization enforcement at the EP.

When profile-based authorization information is provided to the EP, more advanced actions like classifiers, meters and schedulers might be configured by the PAA. This is out of the scope of the present document.

5.2. Network Layer Secure Access Control (IPsec)

The PANA protocol authenticates the client and also establishes a PANA security association between the PANA client and PANA authentication agent at the end of successful authentication. The PAA indicates the results of the authentication using the PANA-Bind-Request (PBR) message wherein it can indicate the access control

method enforced by the access network and the IP address of the corresponding EP.

When IPsec is used to perform access control, the PANA protocol [I-D.ietf-pana-pana] does not discuss any details of IPsec [RFC2401] SA establishment. Indeed, [I-D.ietf-pana-ipsec] discusses the details for establishing IPsec security associations between the PaC and the EP. When the IPsec SAs are successfully established, it can be used to enforce access control and specifically used to prevent the service theft mentioned in [RFC4016].

In this particular context, one assumes that the following have already happened before the IPsec SAs are established:

1. PANA client (PaC) and PAA mutually authenticate each other using EAP methods that derive the AAA-Key [I-D.ietf-eap-keying].

2. PaC learns the IP address of the Enforcement point (EP) during the PANA exchange.

3. PaC learns that the network uses IPsec [RFC2401] for securing the link between PaC and EP during the PANA exchange (PBR message).

4. PaC configures an IP address address before the PANA protocol begins (the pre-PANA Address (PRPA), see [I-D.ietf-pana-pana]).

The IPsec IKE Action MIB module (IKEACTION-MIB) works within the framework of the IPsec SPD-MIB. It can be referenced as an action by the SPD-MIB and is used to configure IKE negotiations between network devices. Hence, together with the SPD-MIB, the IKEACTION-MIB module enables the PAA to configure IPSEC-based access control at the EP.

The PAA is then responsible to communicate to EP the following information before IKE phase 1 exchange begins between PaC and EP:

The IKE pre-shared key:

> To this end, the PAA must set a row in the IKE Credential Filter table of the IKEACTION-MIB. This table defines filters, which can be used to match credentials of IKE peers, where the credentials in question have been obtained from an IKE phase 1 exchange. They may be X.509 certificates, Kerberos tickets, or Pre-shared keys, etc.

The PRPA of the PaC:

The DiffServ Multi-Field Classifier of the DIFFSERV-MIB is used for configuring the SPD in a similar manner than what is described in Section 5.1.

The Key-Id and PANA session ID:

[I-D.ietf-pana-ipsec] states that PaC and EP should use the PANA session ID concatenated with the AAA-key as the value of the ID_KEY_ID in aggressive mode for establishing the phase 1 SA. Section 8 details usage examples that illustrate the way the IKEACTION-MIB is used for this purpose.

6. PANA extension to the IPsec SPD MIB

Many existing MIB objects defined in the IPsec Configuration MIB modules can be efficiently re-used for the PANA-specific needs. This is detailed in Section 5.

The present section defines additional PANA-specific objects that extend the IPsec SPD MIB module in order to entirely satisfy the PAA-EP interface requirements.

6.1. Bootstrapping link layer ciphers

PANA can bootstrap not only network layer secure access control but also any type of link layer secure access control. The following parameters are defined to support bootstrapping link layer secure access control:

panaL2FiltPmk:

A PMK (Pairwise Master Key) that is derived from AAA-Key and used as the shared secret needed for executing a secure association protocol between the PaC and EP in order to generate TSKs (Transient Session Keys) [I-D.ietf-eap-keying]. In the case of IEEE 802.11i [802.11i], an 802.11i PMK is carried in this MIB object.

PMK Name:

The name fo the PMK. In the case of IEEE 802.11i, a 802.11i PMKID is carried in this MIB object.

PMK Lifetime:

The lifetime of the PMK. The PMK must be invalidated when the PMK lifetime expires. A new PMK with a new PMK lifetime may be installed before the lifetime of the current PMK expires. The PMK lifetime may be calculated from a RADIUS Session-Timeout attribute or a Diameter Authorization-Lifetime AVP.

These parameters are contained in PanaL2FilterEntry together with other link layer filtering parameters.

6.2. Notification of PaC presence

The SPD-MIB provides a means to notify to the SNMP manager (PAA) information on packets matching/not matching the filters of given rule. Such notification mechanisms and objects can be re-used for notifying the PAA that unauthorized packets are trying to pass

through the EP.

If reliability needs to be guaranteed for the notification (panaNewPacNotification), hence Inform notification (which is acknowledged) MUST be used. Then the PAA needs to have engine-id to be the authoritative of SNMP clock between EP and PAA (for inform operation the responder becomes the authoritative).

6.3. PANA MIB Overview

Link-Layer filter table

[I-D.ietf-pana-pana] says the Device-Id AVP (code 1025) is of Address Type [RFC3588]. The content for link-layer addresses is expected to be specified in specific documents that describe how IP operates over different link-layers. For instance, IPv6 over Ethernet is described in [RFC2464]. To this end, additional filters are designed in the present document. The link-layer filter test the L2 address (e.g. MAC address, port, DSL line) and also defines a set of parameters needed for bootstrapping link-layer ciphering, including a PMK (Pair-wise Master Key), the PMK name and the PMK lifetime. Note that the definition of this table does not assume the usage of any particular link-layer.

When the MIB module definition for a particular link-layer defines MIB objects for the parameters specific to that link-layer, the link-layer specific parameters SHOULD be accessed via the PANA MIB using the link-layer filter table, instead of accessing them via the link-layer specific MIB.

New PaC notification tables

This table defines a new notification, which aims at satisfying this requirement. It re-uses existing notification variable objects pre-defined in the SPD-MIB.

The "New PaC" notification is triggered when the EP detects traffic coming from an unauthorized source.

If the traffic detected is an IP flow, the objects sent must include spdIPSourceType, spdIPSourceAddress, spdIPDestinationType, and spdIPDestinationAddress objects to indicate the packet source and destination of the packet that triggered the action. Additionally, the spdIPInterfaceType and spdIPInterfaceAddress objects are included to indicate which interface the action was executed in association with and if the packet was inbound or outbound through the endpoint. See [I-D.ietf-ipsp-spd-mib] for further details.

If the traffic detected is Link-layer traffic, the objects sent must include the index of the interface which detected such traffic and potentially the L2 address source of the traffic.

6.4. PANA MIB Objects Definition

```
PANA-EP-MIB DEFINITIONS ::= BEGIN

IMPORTS

MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE
FROM SNMPv2-SMI

TEXTUAL-CONVENTION, RowStatus, PhysAddress, StorageType,
TimeStamp, TimeInterval
FROM SNMPv2-TC

MODULE-COMPLIANCE, OBJECT-GROUP, NOTIFICATION-GROUP
FROM SNMPv2-CONF

InterfaceIndex
FROM IF-MIB

spdMIB, spdIPEndpointAddType, spdIPEndpointAddress,
spdActionExecuted, spdIPSourceType, spdIPSourceAddress,
spdIPDestinationType,spdIPDestinationAddress
FROM IPSEC-SPD-MIB;

--
-- Module identity
--

panaMIB MODULE-IDENTITY
LAST-UPDATED
"200605280000Z"            -- 28 may 2006
ORGANIZATION
"IETF PANA Working Group"
CONTACT-INFO
"Yacine El Mghazli
Alcatel
Route de Nozay
91460 Marcoussis
France
Email: yacine.el_mghazli@alcatel.fr

Yoshihiro Ohba
```

```
Toshiba America Research, Inc.
1, Telcordia Drive
Piscataway, NJ  08854
USA
Email: yohba@tari.toshiba.com"
DESCRIPTION
"The MIB module for defining additional PANA-specific objects to
the IPsec SPD MIB. Copyright (C) The Internet Society (2003).
This version of this MIB module is part of RFC XXXX, see the
RFC itself for full legal notices."

-- Revision History
REVISION
"200605280000Z"            -- 28 may 2006
DESCRIPTION
"Removed L2 notif"
REVISION
"200512280000Z"            -- 22 december 2005
DESCRIPTION
"L2 Filter indexing modified"
REVISION
"200506280000Z"            -- 28 Juin 2005
DESCRIPTION
"L2 protection generic parameters"
REVISION
"200502050000Z"            -- 05 February 2005
DESCRIPTION
"L2 generic filters"
REVISION
"200410220000Z"            -- 22 October 2004
DESCRIPTION
"Version 02, draft-ietf-pana-snmp-02.txt"
REVISION
"200402050000Z"            -- 05 February 2004
DESCRIPTION
"Version 01, draft-yacine-pana-paa2ep-snmp-01.txt"
REVISION
"200310310000Z"            -- 31 October 2003
DESCRIPTION
"Initial version, draft-yacine-pana-paa2ep-snmp-00.txt"
::= { spdMIB  XXX } -- XXX to be assigned by IANA

--
-- groups of related objects
--

panaConfigObjects         OBJECT IDENTIFIER
::= { panaMIB 1 }
```

```
               panaNotificationObjects   OBJECT IDENTIFIER
               ::= { panaMIB 2}
               panaConformanceObjects    OBJECT IDENTIFIER



               ::= { panaMIB 3 }




               --
               -- Textual Conventions
               --

               PanaKey ::= TEXTUAL-CONVENTION
               STATUS   current
               DESCRIPTION
               "The PanaKey is used to carry a key.  When the key does
               not exist, the length of the key becomes zero."
               SYNTAX      OCTET STRING (SIZE(0..255))

               PanaKeyName ::= TEXTUAL-CONVENTION
               STATUS   current
               DESCRIPTION
               "The PanaKeyName is used to carry the name of a PanaKey.
               When the key name does not exist, the length of the
               key name becomes zero."
               SYNTAX      OCTET STRING (SIZE(0..255))




               --
               -- PANA Additional Filters Objects
               --


               --
               -- The Link-layer Filter Table
               --


               panaL2FilterTable OBJECT-TYPE
               SYNTAX      SEQUENCE OF PanaL2FilterEntry
               MAX-ACCESS  not-accessible
               STATUS      current
               DESCRIPTION
               "Link-layer filter definitions."
               ::= { panaConfigObjects 1 }
```

```
panaL2FilterEntry OBJECT-TYPE
SYNTAX      PanaL2FilterEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
"An entry in the Link-layer filter table."
INDEX       { panaL2FiltEpIfIndex, panaL2FiltAddr }
::= { panaL2FilterTable 1 }

PanaL2FilterEntry ::= SEQUENCE {
panaL2FiltEpIfIndex         InterfaceIndex,
panaL2FiltAddr              PhysAddress,
panaL2FiltPmk               PanaKey,
panaL2FiltPmkName           PanaKeyName,
panaL2FiltPmkLifetime       TimeInterval,
panaL2FiltLastChanged       TimeStamp,
panaL2FiltStorageType       StorageType,
panaL2FiltRowStatus         RowStatus
}

panaL2FiltEpIfIndex OBJECT-TYPE
SYNTAX         InterfaceIndex
MAX-ACCESS     read-create
STATUS         current
DESCRIPTION
"The index identifying the EP interface where the filter
policy must be enforced on."
::= { panaL2FilterEntry 1 }

panaL2FiltAddr OBJECT-TYPE
SYNTAX         PhysAddress
MAX-ACCESS     read-create
STATUS         current
DESCRIPTION
"The authorized device Link-layer address (DI). For
example, for a 802.x interface, this object normally
contains a MAC address. For interfaces which do not have such
an address (e.g., a serial line), this object should contain
an octet string of zero length."
::= { panaL2FilterEntry 2 }

panaL2FiltPmk OBJECT-TYPE
SYNTAX      PanaKey
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
"This is PMK (Pairwise Master Key) used for bootstraping
link-layer ciphers."
```

```
               ::= {  panaL2FilterEntry 3 }

               panaL2FiltPmkName OBJECT-TYPE
               SYNTAX      PanaKeyName
               MAX-ACCESS  read-create
               STATUS      current
               DESCRIPTION
               "This is the name of the panaL2Pmk."
               ::= { panaL2FilterEntry 4 }

               panaL2FiltPmkLifetime OBJECT-TYPE
               SYNTAX      TimeInterval
               MAX-ACCESS  read-create
               STATUS      current
               DESCRIPTION
               "This is the lifetime of panaL2Pmk."
               ::= { panaL2FilterEntry 5 }

               panaL2FiltLastChanged OBJECT-TYPE
               SYNTAX      TimeStamp
               MAX-ACCESS  read-only
               STATUS      current
               DESCRIPTION
               "The value of sysUpTime when this row was last modified or
               created either through SNMP SETs or by some other external
               means."
               ::= { panaL2FilterEntry 6 }

               panaL2FiltStorageType OBJECT-TYPE
               SYNTAX      StorageType
               MAX-ACCESS  read-create
               STATUS      current
               DESCRIPTION
               "The storage type for this row. Rows in this table which were
               created through an external process may have a storage type
               of readOnly or permanent."
               DEFVAL { nonVolatile }
               ::= { panaL2FilterEntry 7 }

               panaL2FiltRowStatus OBJECT-TYPE
               SYNTAX      RowStatus
               MAX-ACCESS  read-create
               STATUS      current
               DESCRIPTION
               "This object indicates the conceptual status of this row."
               ::= { panaL2FilterEntry 8 }
```

```
--
--
-- Notification objects information



--
--

panaNotificationVariables OBJECT IDENTIFIER ::=
{ panaNotificationObjects 1 }

panaNotifications OBJECT IDENTIFIER ::=



{ panaNotificationObjects 0 }

panaEpIfIndex OBJECT-TYPE
SYNTAX      InterfaceIndex
MAX-ACCESS  accessible-for-notify
STATUS      current
DESCRIPTION
"Contains the interface index on which the packet triggered
the notification in question."
::= { panaNotificationVariables 1 }

panaL2SourceAddress OBJECT-TYPE
SYNTAX      PhysAddress
MAX-ACCESS  accessible-for-notify
STATUS      current
DESCRIPTION
"Contains the source Link layer address of the packet which
triggered the notification in question. For
example, for a 802.x frame, this object normally
contains a MAC address. For interfaces which do not have such
an address (e.g., a serial line), this object should contain
an octet string of zero length."
::= { panaNotificationVariables 2 }


panaNewPacNotification NOTIFICATION-TYPE
OBJECTS {
spdActionExecuted,
spdIPEndpointAddType,
spdIPEndpointAddress,
spdIPSourceType,
spdIPSourceAddress,
```

```
               spdIPDestinationType,
               spdIPDestinationAddress,
               panaEpIfIndex,
               panaL2SourceAddress}
               STATUS  current
               DESCRIPTION
               "Notification that EP detected traffic coming from an
               unauthorized source.  When source and destination IP addresses
               of the traffic is unknown, spdIPSourceType and
               spdIPDestinationType must be zero.  When source L2 address of
               the traffic is unknown, panaL2SourceAddress must be
zero.Notification
               that EP detected traffic coming from an
               unauthorized source."
               ::= { panaNotifications 1 }

               --
               --
               -- Conformance information
               --
               --

               panaGroups OBJECT IDENTIFIER
               ::= { panaConformanceObjects 1 }
               panaCompliances OBJECT IDENTIFIER
               ::= { panaConformanceObjects 2 }

               --
               -- Compliance Groups Definitions
               --

               panaL2FilterGroup OBJECT-GROUP
               OBJECTS {
               panaL2FiltEpIfIndex,
               panaL2FiltAddr,
               panaL2FiltPmk,
               panaL2FiltPmkName,
               panaL2FiltPmkLifetime,
               panaL2FiltLastChanged,
               panaL2FiltStorageType,
               panaL2FiltRowStatus }
               STATUS current
               DESCRIPTION
               "The Link-layer Filter Group."
               ::= { panaGroups 1 }

               panaNewPacNotificationObjectsGroup OBJECT-GROUP
               OBJECTS {
```

```
               panaEpIfIndex,
               panaL2SourceAddress }
               STATUS current
               DESCRIPTION
               "PaC Presence Notification Objects Group."
               ::= { panaGroups 2 }

               panaNewPacNotificationGroup NOTIFICATION-GROUP
               NOTIFICATIONS {
               panaNewPacNotification}
               STATUS current
               DESCRIPTION
               "PaC Presence Notification Group."
               ::= { panaGroups 3 }

               --
               -- Compliance statements
               --




               panaFilterCompliance MODULE-COMPLIANCE
               STATUS      current
               DESCRIPTION
               "The compliance statement for SNMP entities that support
               PANA DI-based filtering."

               MODULE -- This Module

               MANDATORY-GROUPS { panaL2FilterGroup }

               OBJECT      panaL2FiltRowStatus
               SYNTAX      RowStatus { active(1), createAndGo(4), destroy(6) }
               DESCRIPTION
               "Support of the values notInService(2), notReady(3),
               and createAndWait(5) is not required."

               OBJECT      panaL2FiltLastChanged
               MIN-ACCESS  not-accessible
               DESCRIPTION
               "This object not required for compliance."


               MODULE IPSEC-SPD-MIB

               MANDATORY-GROUPS {
               spdEndpointGroup,
               spdGroupContentsGroup,
```

```
               spdRuleDefinitionGroup,
               spdStaticFilterGroup,
               spdStaticActionGroup }

               OBJECT      spdEndGroupRowStatus
               SYNTAX      RowStatus { active(1), createAndGo(4), destroy(6) }
               DESCRIPTION
               "Support of the values notInService(2), notReady(3),
               and createAndWait(5) is not required."

               OBJECT      spdEndGroupLastChanged
               MIN-ACCESS  not-accessible
               DESCRIPTION
               "This object not required for compliance."

               OBJECT      spdGroupContComponentType
               SYNTAX      INTEGER { rule(2) }
               DESCRIPTION
               "Support of the value group(1) is only required for
               implementations which support Policy Groups within
               Policy Groups."



               OBJECT      spdGroupContRowStatus
               SYNTAX      RowStatus { active(1), createAndGo(4), destroy(6) }
               DESCRIPTION
               "Support of the values notInService(2), notReady(3),
               and createAndWait(5) is not required."

               OBJECT      spdGroupContLastChanged
               MIN-ACCESS  not-accessible
               DESCRIPTION
               "This object not required for compliance."

               OBJECT      spdRuleDefRowStatus
               SYNTAX      RowStatus { active(1), createAndGo(4), destroy(6) }
               DESCRIPTION
               "Support of the values notInService(2), notReady(3),
               and createAndWait(5) is not required."

               OBJECT      spdRuleDefLastChanged
               MIN-ACCESS  not-accessible
               DESCRIPTION
               "This object not required for compliance."

               ::= { panaCompliances 1 }
```

```
panaNewPacNotificationCompliance MODULE-COMPLIANCE
STATUS      current
DESCRIPTION
"The compliance statement for SNMP entities that support
new PaC presence Notification."

MODULE -- This Module

MANDATORY-GROUPS {
panaNewPacNotificationObjectsGroup,
panaNewPacNotificationGroup
}

MODULE IPSEC-SPD-MIB

MANDATORY-GROUPS { spdActionLoggingObjectGroup }

::= { panaCompliances 2 }

END
```

7. Applicability of RTFM Meter MIB

RTFM provides for the measurement of network traffic flows:

- a method of specifying traffic flows within a network

- a hierarchy of devices (meters, meter readers, managers) for measuring the specified flows

- a mechanism for configuring meters and meter readers, and for collecting the flow data from remote meters

RTFM provides high time resolution for flow first- and last-packet times. Counters for long-duration flows may be read at intervals determined by a manager. The RTFM Meter is designed so as to do as much data reduction work as possible, which minimizes the amount of data to be read and the amount of processing needed to produce useful reports from it.

RTFM flow data can be used for a wide range of purposes, such as usage accounting, long-term recording of network usage (classified by IP address attributes) and real-time analysis of traffic flows at remote metering points.

PANA recommends the use of the following RTFM module and architecture for the PAA to collect accounting information from the EP(s). It is the responsability of the PAA to organise the correlation between the EP filters, RTFM flows, the PANA and the AAA sessions.

RTFM Meter MIB [RFC2720] describes the SNMP Management Information Base for an RTFM meter, including its flow table, rule table (storing the meter's rulesets) and the control tables used for managing a meter and reading flow data from it.

[RFC2722] defines the RTFM Architecture, giving descriptions of each component. Explains how traffic flows are viewed as logical entities described in terms of their address-attribute values, so that each is defined by the attributes of its end-points. Gives a detailed description of the RTFM traffic meter, with full details of how flows are stored in the meter's flow table, and how packets are matched in accordance with rules stored in a ruleset.

8. EP Configuration Example

Below are usage examples of the MIB modules in the PANA context.

8.1. Common setup

The "EndPoint to Group" table is used to map policy (groupings) onto an endpoint (EP-ADDR is the IP address) where traffic is to pass by. Any policy group assigned to an endpoint is then used to control access to the traffic passing by it.

So far, we define below two policy groups at the EP interface ("EP-SPD-IN" and "EP-SPD-OUT").

spdEndpointToGroupTable, row 1:

- spdEndGroupDirection = incoming;
- spdEndGroupIdentType = IPv4;
- spdEndGroupAddress = EP-ADDR;
- spdEndGroupName = "EP-SPD-IN";

spdEndpointToGroupTable, row 2:

- spdEndGroupDirection = outgoing;
- spdEndGroupIdentType = IPv4;
- spdEndGroupAddress = EP-ADDR;
- spdEndGroupName = "EP-SPD-OUT";

Within each of the policy group defined above, we define a sub-group (a compound filter) dedicated to the treatment of traffic that is allowed prior to any configuration. The following configuration illustrates the incoming allowed unprotected traffic:

spdGroupContentsTable, row 1:

- spdGroupContName = "EP-SPD-IN";
- spdGroupContPriority = '10';
- spdGroupContFilter = spdTrueFilterInstance;

- spdGroupContComponentType = group;

- spdGroupContComponentName = "EP-ALLOWED-UNPROTECTED-IN";

Within this sub-group, we define a rule (a sub-filter) dedicated to each allowed traffic types (namely PANA, ARP, IPv6 Neighbor Discovery, DHCP, etc). The following configuration illustrates the case of incoming DHCP traffic:

spdGroupContentsTable, row 2:

- spdGroupContName = "EP-ALLOWED-UNPROTECTED-IN";

- spdGroupContPriority = '1';

- spdGroupContFilter = diffServMultiFieldClfrTable.10;

- spdGroupContComponentType = rule;

- spdGroupContComponentName = "EP-DHCP-ACCEPT-IN";

We define the corresponding filter in the DiffServ Multi-Field Classifier table (DIFFSERV_MIB, [RFC3289]), which matches the DHCP packets:

diffServMultiFieldClfrTable, row 10:

- diffServMultiFieldClfrAddrType = v4;

- diffServMultiFieldClfrSrcL4PortMin = '546' (DHCP);

- diffServMultiFieldClfrSrcL4PortMax = '547' (DHCP);

- diffServMultiFieldClfrProtocol = '17' (UDP);

The "Rule Defininition" table links a rule with a given action in the SPD action MIB. E.g. the following entries links the filter defined above with the "accept" action statically defined in the SPD MIB (spdStaticActions.3).

spdRuleDefinitionTable, row 1:

- spdRuleDefName = "EP-DHCP-ACCEPT-IN";

- spdRuleDefDescription = "Allow Incoming DHCP packets";

- spdRuleDefFilter = spdTrueFilterInstance;

- o spdRuleDefFilterNegated = false (default);

- o spdRuleDefAction = spdAcceptAction;

8.2. General IP-based Access Control

In this section, we need to configure the SPD so that EP accepts IP packets coming from/going to the client 'PaC1'. In the whole section, 'PaC1-IP@' is the IP address of 'PaC1'.

Within the incoming policy group defined above ("EP-SPD-IN"), we define a rule dedicated to the treatment of packets coming from "PaC1" and the EP we are provisioning.

spdGroupContentsTable, row 3:

- o spdGroupContName = "EP-SPD-IN";

- o spdGroupContPriority = '100';

- o spdGroupContFilter = diffServMultiFieldClfrTable.1;

- o spdGroupContComponentType = rule;

- o spdGroupContComponentName = "EP-PaC1-ACCEPT-IN";

We define the filter in the DiffServ Multi-field Classifier table, which matches IP packets coming from the PaC:

diffServMultiFieldClfrTable, row 1:

- o diffServMultiFieldClfrAddrType = v4;

- o diffServMultiFieldClfrSrcAddr = 'PaC1-IP@';

diffServMultiFieldClfrTable, row 2:

- o diffServMultiFieldClfrAddrType = v4;

- o diffServMultiFieldClfrDstAddr = 'PaC1-IP@';

The following entries links the two filters defined above with the "accept" action.

spdRuleDefinitionTable, row 2:

- o spdRuleDefName = "EP-PaC1-ACCEPT-IN";

- spdRuleDefDescription = "Allow Incoming IP packets from PaC1";

- spdRuleDefFilter = spdTrueFilterInstance;

- spdRuleDefFilterNegated = false (default);

- spdRuleDefAction = spdAcceptAction;

spdRuleDefinitionTable, row 3:

- spdRuleDefName = "EP-PaC1-ACCEPT-OUT";

- spdRuleDefDescription = "Allow Outgoing IP packets to PaC1";

- spdRuleDefFilter = spdTrueFilterInstance;

- spdRuleDefFilterNegated = false (default);

- spdRuleDefAction = spdAcceptAction;

The same thing must be done for the outgoing direction and this results in a policy such that any packet coming from/going to the PaC is allowed to go through the EP (via EP-ADDR endpoint).

8.3. IPsec-based Access Control

In this section we consider the case when IPsec is used to control access at the IP level. In order to avoid many redundancies, the previous configuration set is still valid. See below a usage example of IKEACTION-MIB and IPSECACTION-MIB.

-- IKE Phase 1 configuration (agressive mode):

We define a first-level filter in policy group "EP-SPD-IN" of the SPD MIB, using the "Group contents" table. This filter isolates IKE phase 1 traffic coming to the EP on this interface:

spdGroupContentsTable, row 4:

- spdGroupContName = "EP-SPD-IN";

- spdGroupContPriority = '1';

- spdGroupContFilter = ipiaIkePhase1Filter;

- spdGroupContComponentType = group;

- spdGroupContComponentName = "EP-IKE1-IN";

Within this IKE-specific policy sub-group, we specify a second-level filter to apply for for the traffic coming from PaC1.

spdGroupContentsTable, row 5:

- spdGroupContName = "EP-IKE-Phase1-IN";
- spdGroupContPriority = '1';
- spdGroupContFilter = diffServMultiFieldClfrTable.1;
- spdGroupContComponentType = group;
- spdGroupContComponentName = "EP-IKE1-PaC1-IN";

The diffServMultiFieldClfrTable.1 entry has been defined in the previous section. Within the lattest sub-group we finally specify the rule to apply for the IKE traffic coming from PaC1 and matching the right identity (id_key_id).

spdGroupContentsTable, row 6:

- spdGroupContName = "EP-IKE1-PaC1-IN";
- spdGroupContPriority = '1';
- spdGroupContFilter = ipiaPeerIdentityFilterTable.1;
- spdGroupContComponentType = rule;
- spdGroupContComponentName = "PaC1-IKE1-RULE";

The "Peer Identity Filter" table specifically informs the EP on the value of the idKeyId to use in IKE messages:

ipiaPeerIdentityFilterTable, row 1:

- ipiaPeerIdFiltName = "PaC1-IKE1-ID-FILTER";
- ipiaPeerIdFiltIdentityType = id_Key_Id;
- ipiaPeerIdFiltIdentityValue = 'PANA-Session-Id|PANA-Key-Id';

The "Rule Defininition" table links a rule with a given action in the IKE action MIB. This action will be triggered upon reception at the EP of an IKE phase 1 packet coming from PaC1 and matching the right

id_key_id.

spdRuleDefinitionTable, row 4:

- spdRuleDefName = "PaC1-IKE-RULE";

- spdRuleDefDescription = "IPsec Access Control for PaC1";

- spdRuleDefFilter = spdTrueFilterInstance;

- spdRuleDefFilterNegated = false (default);

- spdRuleDefAction = ipiaIkeActionTable.1;

The spdRuleDefAction attribute in the entry above points to a row in the ipiaIkeActionTable, defined below:

ipiaIkeActionTable, row 1:

- ipiaIkeActName = "PaC1-IKE";

- ipiaIkeActParametersName = "SA1-PaC1";

- ipiaIkeActThresholdDerivedKeys = '100' (default);

- ipiaIkeActExchangeMode = aggressive;

- ipiaIkeActAgressiveModeGroupId = 'xxx' (Diffie-Hellman values);

- ipiaIkeActIdentityType = id_Key_Id;

- ipiaIkeActIdentityContext = "PANA";

- ipiaIkeActPeerName = "PaC1";

The following entry links together the "PaC1" identity with the corresponding credentials table entry:

ipiaIkeIdentityTable, row 1:

- spdEndGroupIdentType = IPv4;

- spdEndGroupAddress = 'EP-ADDR';

- ipiaIkeActIdentityType = id_Key_Id;

- ipiaIkeActIdentityContext = 'PANA';

- ipiaIkeIdCredentialName = "PaC1-PSK";

Finally, the pre-shared key derivated at the PAA is set in the IPSA Credentials table (IPSECACTION-MIB):

ipsaCredentialTable, row 1:

- ipsaCredName = "PaC1-PSK";

- ipsaCredType = sharedSecret;

- ipsaCredCredential = 'PSK-derived-at-the-PAA';

- ipsaCredSize = 'xxx';

- ipsaMgnName = "xxx";

- ipsaRemoteID = 'PaC1';

The Ike Action entry (above) is indexed by a name (ipiaIkeActName attribute), which is used as the primary index into the ipiaIkeActionProposalsTable. The secondary index is a priority, which allows ordering of the proposals that will be sent to the peer (or accepted from the peer).

ipiaIkeActionProposalsTable, row 1:

- ipiaIkeActPropName = "PaC1-IKE-PROP1";

- ipiaIkeActPropPriority = '1';

The ipiaIkeActPropName points to a row in the ipiaIkeProposalsTable, which contains the actual IKE parameters for the Phase 1 exchanges:

ipiaIkeProposalsTable, row 1:

- ipiaIkeActPropName = "PaC1-IKE-PROP1";

- ipiaIkePropLifetimeDerivedKeys = xxx;

- ipiaIkePropCipherAlgorithm = 3DES;

- ipiaIkePropCipherKeyLength = xxx;

- ipiaIkePropHashAlgorithm = HMAC-SHA1;

- ipiaIkePropPrfAlgorithm = xxx;

- o ipiaIkePropVendorId = xxx;

- o ipiaIkePropDhGroup = 2;

- o ipiaIkePropAuthenticationMethod = xxx;

- o ipiaIkePropMaxLifetimeSecs = xxx;

- o ipiaIkePropMaxLifetimeKB = xxx;

There is a similar hierarchy for the proposals, which are used for Phase 2 negotiations, and result in a Phase 2 SA being created, upon successful negotiations. The parameters would be set up in the ipiaIpsecActionTable, ipiaIpsecProposalsTable and ipiaIpsecTransformsTable, along with the necessary related tables. One would then need a rule to trigger the row in the ipiaIpsecActionTable. See for further details on the IPSP MIBs usage.

8.4. Link-layer Access control

The section below gives a usage example of the PANA module in the general L2-based access control case.

The example below assumes the configuration set detailed in Section 8.1 is valid. Here we need to configure the SPD so that EP accepts accepts 802 packets coming from/going to the client 'PaC1'? In the whole section 'PaC1-MAC@' is the MAC address of 'PaC1'.

Within the incoming policy group defined above ("EP-SPD-IN"), we define a rule dedicated to the treatment of packets coming from "PaC1" and the EP we are provisioning.

spdGroupContentsTable, row 3:

- o spdGroupContName = "EP-SPD-IN";

- o spdGroupContPriority = '101';

- o spdGroupContFilter = panaL2FilterTable.1;

- o spdGroupContComponentType = rule;

- o spdGroupContComponentName = "EP-PaC1-ACCEPT-IN-L2";

We define the filter in the Link-layer filter table, which matches L2 packets coming from the PaC to interface #5, which a 802.11 interface:

panaL2FilterTable, row 1:

- panaL2FiltEpIfIndex = 5;

- panaL2FiltAddr = "00-11-0A-80-4A-58";

- panaL2FiltPmk = "...";

- panaL2FiltPmkName = "...";

- panaL2FiltPmkLifetime = 10800;

The following entries links the filter defined above with the "accept" action.

spdRuleDefinitionTable, row 20:

- spdRuleDefName = "EP-PaC1-ACCEPT-IN-L2";

- spdRuleDefDescription = "Allow Incoming L2 packets from PaC1 via Interface #5";

- spdRuleDefFilter = spdTrueFilterInstance;

- spdRuleDefFilterNegated = false (default);

- spdRuleDefAction = spdAcceptAction;

The same thing must be done for the outgoing direction and this results in a policy such that any packet coming from/going to the PaC is allowed to go through the EP.

9. Security Considerations

The MIB defined in the present document relates to a system which will provide network access. As such, improper manipulation of the objects represented by this MIB may result in denial of service to a large number of end-users. In addition, manipulation of the panaL2FilterTable and diffServMultiFieldClfrTable may allow an end-user to gain network access, spoof their IP addresses, change the authorized device identifiers, or affect other end-users in either a positive or negative manner.

There are a number of management objects defined in this MIB module with a MAX-ACCESS clause of read-write and/or read-create. Such objects may be considered sensitive or vulnerable in some network environments. The support for SET operations in a non-secure environment without proper protection can have a negative effect on network operations. These are the tables and objects and their sensitivity/vulnerability:

- The use of panaL2FilterTable to specify which Link-layer addresses are authorized to access the network on which interface is considered to be only limited protection and does not protect against attacks which spoof the management station's IP address. The use of SNMPv3 security is mandated. Specifically, SNMPv3 VACM and USM MUST be used with any v3 agent which implements this MIB.

As described in Section 2.1, it is assumed that PAA and EP are pre-configured to be able to communicate each other with a required security association. The pre-configured SNMP user passphrase for the security association between PAA and EP should not be used for a long term and should be updated at a reasonable frequency.

It might be sufficient to use the same SNMP user identity and passphrase for all EPs controlled by the same PAA (the same thing can apply even when the EPs are controlled by multiple PAAs.) Note that even if the same SNMP user identity and passphrase for all EPs are used, the different SNMP authentication and encryption keys are derived for each EP, because the combination of SNMP user identity and engine-id is used for deriving the keys.

SNMP versions prior to SNMPv3 did not include adequate security. Even if the network itself is secure (for example by using IPsec), even then, there is no control as to who on the secure network is allowed to access and GET/SET (read/change/create/delete) the objects in this MIB module.

It is RECOMMENDED that implementers consider the security features as provided by the SNMPv3 framework (see [RFC3410], section 8),

including full support for the SNMPv3 cryptographic mechanisms (for authentication and privacy).

Further, deployment of SNMP versions prior to SNMPv3 is NOT RECOMMENDED. Instead, it is RECOMMENDED to deploy SNMPv3 and to enable cryptographic security. It is then a customer/operator responsibility to ensure that the SNMP entity giving access to an instance of this MIB module is properly configured to give access to the objects only to those principals (users) that have legitimate rights to indeed GET or SET (change/create/delete) them.

10. IANA Considerations

The only IANA considerations for this document is the node number allocation of the PANA-EP-MIB itself.

11. Acknowledgements

The authors would like to thank David Perkins for the MIB deep inspection and Robert Story, who provided very helpful recommendations on the IPSP MIBs usage.

This document originaly leverages on similar works done in the MIDCOM working group. Thanks to the authors of those IDs. Thanks to Thomas Moore and Olivier Marce for their grateful help during the edition of this document.

12. References

12.1. Normative References

[I-D.ietf-pana-pana]
Forsberg, D., "Protocol for Carrying Authentication for Network Access (PANA)", draft-ietf-pana-pana-11 (work in progress), March 2006.

[I-D.ietf-pana-ipsec]
Parthasarathy, M., "PANA Enabling IPsec based Access Control", draft-ietf-pana-ipsec-07 (work in progress), July 2005.

[I-D.ietf-ipsp-spd-mib]
Hardaker, W., "IPsec Security Policy Database Configuration MIB", draft-ietf-ipsp-spd-mib-06 (work in progress), April 2006.

[I-D.ietf-ipsp-ikeaction-mib]
Hardaker, W., "IPsec Security Policy IKE Action MIB", draft-ietf-ipsp-ikeaction-mib-01 (work in progress), August 2005.

[I-D.ietf-ipsp-ipsecaction-mib]
Hardaker, W., "IPsec Security Policy IPsec Action MIB", draft-ietf-ipsp-ipsecaction-mib-01 (work in progress), August 2005.

[RFC2578] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, April 1999.

[RFC2579] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Textual Conventions for SMIv2", STD 58, RFC 2579, April 1999.

[RFC2580] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Conformance Statements for SMIv2", STD 58, RFC 2580, April 1999.

[RFC3289] Baker, F., Chan, K., and A. Smith, "Management Information Base for the Differentiated Services Architecture", RFC 3289, May 2002.

12.2. Informative References

[RFC4058] Yegin, A., Ohba, Y., Penno, R., Tsirtsis, G., and C. Wang, "Protocol for Carrying Authentication for Network Access (PANA) Requirements", RFC 4058, May 2005.

[RFC4016] Parthasarathy, M., "Protocol for Carrying Authentication and Network Access (PANA) Threat Analysis and Security Requirements", RFC 4016, March 2005.

[I-D.ietf-pana-framework]
Jayaraman, P., "PANA Framework", draft-ietf-pana-framework-06 (work in progress), March 2006.

[RFC3414] Blumenthal, U. and B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", STD 62, RFC 3414, December 2002.

[RFC3410] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", RFC 3410, December 2002.

[RFC3411] Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks", STD 62, RFC 3411, December 2002.

[RFC2401] Kent, S. and R. Atkinson, "Security Architecture for the Internet Protocol", RFC 2401, November 1998.

[RFC2409] Harkins, D. and D. Carrel, "The Internet Key Exchange (IKE)", RFC 2409, November 1998.

[RFC3588] Calhoun, P., Loughney, J., Guttman, E., Zorn, G., and J. Arkko, "Diameter Base Protocol", RFC 3588, September 2003.

[I-D.ietf-aaa-diameter-cc]
Mattila, L., Koskinen, J., Stura, M., Loughney, J., and H. Hakala, "Diameter Credit-control Application", draft-ietf-aaa-diameter-cc-06 (work in progress), August 2004.

[I-D.ietf-eap-keying]
Aboba, B., "Extensible Authentication Protocol (EAP) Key Management Framework", draft-ietf-eap-keying-13 (work in progress), May 2006.

[802.11i] IEEE P802, "Wireless MAC and PHY specifications, MAC security enhancements", IEEE 802.11i, July 2004.

[RFC2464] Crawford, M., "Transmission of IPv6 Packets over Ethernet Networks", RFC 2464, December 1998.

[RFC2720] Brownlee, N., "Traffic Flow Measurement: Meter MIB", RFC 2720, October 1999.

[RFC2722] Brownlee, N., Mills, C., and G. Ruth, "Traffic Flow Measurement: Architecture", RFC 2722, October 1999.

Authors' Addresses

Yacine El Mghazli (Editor)
Alcatel
Route de Nozay
Marcoussis 91460
France

Email: yacine.el_mghazli@alcatel.fr

Yoshihiro Ohba
Toshiba America Research, Inc.
1, Telcordia Drive
Piscataway, NJ 08854
USA

Email: yohba@tari.toshiba.com

Julien Bournelle
GET/INT
9, rue Charles Fourier
Evry 91011
France

Email: julien.bournelle@int-evry.fr

Intellectual Property Statement

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.