SNMP Version 3

More about VACM and USM

Nick Urbanik <nicku(at)nicku.org> © 2003, 2005
Copyright Conditions: Open Publication License
(see http://www.opencontent.org/openpub/)

Goals of SNMPv3 (RFC 3411)
- Avoid reinventing the wheel—use existing work
- Support secure net operation
- Support forward and backward compatibility
- Support remote configuration
  - USM and VACM configuration is through SNMP tables and variables
- Security protection against:
  - modification of information by unauthorised parties
  - an unauthorised person masquerading as an authorised person
  - message stream modification by reordering, delaying or replaying exchanges
  - disclosure (eavesdropping)

VACM

- The View-based Access Control Model (VACM)
- VACM has five main components, as we mentioned earlier:
  - groups of users
  - security level, i.e., v1, v2c, usm
  - contexts — see slide 8
  - MIB views, view families — see slide 8
  - access policy, i.e., read only, read-write, notify, no access.
- How do we set up SNMPv3 users on agents and network management software?
- How do we control access to a subset of the variables on an agent?

Context Example from RFC 3411

iisAccessAllowed from RFC 3415

VACM on Net-SNMP

Net-SNMP VACM

- Net-SNMP uses four keywords to set up VACM in /etc/snmp/snmpd.conf:
  - com2sec
  - group
  - view
  - access
- These set up access control to variables on the agent.
  - access and view determine what access is being controlled to.
  - group and com2sec determine who has this access.
The access Keyword

- Specifies which group has access to which parts of the MIB tree.
- Has 8 parameters. Syntax (all on one line):
  access (group) (context) (securityModel) (prefix) (readview) (writeview) (notifyview) (context)
- Last three parameters (readview), (writeview), (notifyview) are views, defined by view statements.
- Indicate which part of the MIB tree has read access, which part of tree has write access, and which part has permission for access to send notifications (i.e., traps or informs requests).
- The (group) parameter is defined by a group statement.
- Represents a group of users.
- Default (context) is the empty string **"**. See slide 2.

access: The (prefix) Parameter

- The (prefix) parameter to access can be either exact or prefix.
- Indicates whether context name needs to match exactly or whether only the first part of the context name needs to match.
- The default value is exact.

access with SNMPv1, v2c

- For SNMPv1 and SNMPv2c clients.
- Security Level will be noauth, and context will be empty (the empty string).

The com2sec keyword

- Maps a community string and a source IP or network address to a security name (user name).
- Syntax:
  com2sec (securityName) (source)
- The security name is used by the group keyword — see group.
- Source can be a hostname, a subnet or the word "default".
- A subnet can be written as IP/mask or IP/BITS, e.g., our lab subnet can be written as 172.19.64.0/255.255.192.0 or 172.19.64.0/18.
- Only needed for access control with SNMPv1 and v2c.
- Not used with SNMPv3.

The group Keyword

- maps pairs of Security Model and Security Name to a group name.
- Syntax:
  group (groupName) (securityModel) (securityName)
- A Security Model is one of: acl, v1, v2c or usm.
- The Security Name is the username.
- All members of one group have the same access rights.
- A user cannot belong to more than one group for each of the three security models.

Views and the view Keyword

- The view determines what part of the MIB access is controlled to.
- Uses concept of a subtree.
  - A subtree is a node in the MIB tree and all the elements under that node.
  - In other words, all the MIB elements in a subtree have the same common prefix.
- Syntax:
  view (viewName) (incl/excl) (subtree) (mask(optional))
The optional view mask allows the access control to select individual rows in a table.

RFC 3415 calls this a family of subtrees, since a row of n elements can be also represented by n subtrees

RFC 3415 calls the mask the family mask

Under mib-2 is the important ifTable
- Provides statistics on each network interface
- Includes such things as network traffic, errors...
- One row in the table for each network interface

Walking ifTable — 1

Walking ifTable — 2

Views

Examples

VACM

VACM on Net-SNMP

VACM

User-based Security Model

Walking ifTable — 4

Walking ifTable — 3

The View Mask — 3

The View Mask — 2

The View Mask — 1

The Network Interface Table, ifTable

In Numbers — 1

In Numbers — 2

In Numbers — 1

In Numbers — 2

ifTable in Numbers — 1

ifTable in Numbers — 2

Instance Number

Notice that the index is the number at the end of the OID
- Called an instance number. Index starts from 1
- Suppose we are an ISP, want to allow customer A to view
  their own network interface, but not that of customer B, their competitor.
- Note that as we go along a row, the OID element just
  before the instance number changes
- Suppose customer A has a network interface with the
  index 5.
- So want to allow access for customer A to
  .1.3.6.1.2.1.2.2.1.5

So we can provide a view mask to specify this:

1 1 1 1 1 1 1 1 0 0 0 0

A zero in the bit mask is like a wildcard or “don’t care” specifier

A mask of all 1’s is the same as a single view subtree

Specify this:

1 1 1 1 1 1 1 1 1 1 1 1

For Net-SNMP, the mask is specified as a list of
hexadecimal bytes separated with ‘.’ or ‘;’.
The View Mask — 3

- Note that in creating a view mask, we start from the left, writing hexadecimal digits.
- We don’t care about the bits representing non-existent elements after the end of the subtree parent.
  - I mean the bits to the right of the vertical line in slide §24
- These bits could be one or zero; I chose zero, since zero means “don’t care; you can use any value here”
- We can specify this family of view subtrees like this:
  - View name included interfaces ifTable ifEntry ifIndex 5 ff a0
  - This view can then be used in an access statement
  - see the example in slide §24

The View Mask — 4

- One bit in the view mask determines access to one element in the OID
  - It doesn’t matter how big or small the numerical component of the OID is
  - one bit controls whether different values for that component are included in the family of view subtrees or not
- RFC 3415 says that any bit mask is extended with 1’s to the same length in bits as the number of identifiers in the OID if it is shorter.
  - As a consequence, a family mask of zero length corresponds to a single view subtree.

Net-SNMP VACM Example 1

```bash
# sec.name source community
cisco local  localnet public

group MyGroup v3 local
group MyGroup v3 ictnetwork
# viewname incl/excl subtree
view all included .1
# group.name context sec.model sec.name
group MyGroup v3 any read write notif
access MyGroup "+" any noauth exact all none
access MyGroup "+" any noauth exact all none
```

Cisco VACM Configuration

- Cisco IOS specifies a view with the following syntax:
  - snmp-server view viewA ifEntry.*.5 included
  - Can specify a group with:
  - snmp-server group groupA v3 auth read viewA
  - Cisco uses the snmp-server user command to specify users and group membership
  - See also pages 284–285 of Essential SNMP.

Net-SNMP VACM Example 2

```bash
group companyA uses companyAManager

group companyA uses companyAManager
view viewA included IPv4-MIB::ifIndex.5 ff.a0
view viewB included IPv4-MIB::ifIndex.2 ff.a0
access companyA ** usm priv exact viewA none none
access companyA ** usm priv exact viewB none none
companyAManager is a USM user that has read-only access to the iTable row that corresponds to the company A’s own network interface, and no other access.
companyBManger is a USM user that has read-only access to the iTable row that corresponds to the company B’s own network interface, and no other access.
```

User-based Security Model

- USM allows remote configuration of users
- Securely supports strong authentication using MD5 or SHA1 and encryption using DES
- Remotely create new users by cloning existing users
- Can only clone a user once
- Each user must be given access using VACM or that user account cannot be used
  - Add the user to a group
  - provide access to that group through views

Configuring USM Users — 1

- USM users can be created with the net-snmp-config program:
  - Stop the agent first, then create the initial user:
  - sudo service snmpd stop
  - $ sudo net-snmp-config --create-snmpv3-user \
    -a "my_password" myuser
  - SNMPv3 pass phrases must be at least 8 characters long.
  - We have created a user "myuser" with a password of "my_password" and using md5 for authentication and DES for encryption.
  - Very simple access control has been added to
  - /usr/share/snmp/snmpd.conf allowing the user write access to entire tree
Now start the agent, and test the user. First we test without encryption, then with encryption:

```
$ sudo service snmpd start
```

```
$ snmpget -v 3 -u myuser -l authNoPriv -a MD5 -A my_password localhost sysUpTime.0
```

```
$ snmpget -v 3 -u myuser -l authPriv -a MD5 -A my_password -x DES -X my_password localhost sysUpTime.0
```

Can create as many users as you like in this way.

Better to improve access control using VACM over the default of write access everywhere.

The standards were updated in December 2002

Most (all?) text books are out of date

**RFC 1155** SNMPv1

**RFC 1157** SMIv1

**RFC 1212** Concise MIB definitions

**RFC 1215** SNMPv1 traps

**RFC 1301** SNMPv2c

**RFC 1350** OLD SNMPv3 overview

**RFC 1378** SMIv2

**RFC 1379** SMIv2 textual conventions

**RFC 1584** SNMP coexistence v1 v2 best practice

**RFC 3411** SNMPv3 architecture

**RFC 3412** SNMPv3 message processing

**RFC 3413** SNMPv3 applications

**RFC 3414** SNMPv3 USM

**RFC 3415** SNMPv3 VACM

**RFC 3416** SNMPv2 protocol operations

**RFC 3417** SNMPv2 transport mappings

**RFC 3418** SNMPv2 MIB

**RFC 3422** SNMPv3 overview

**RFC 3210** SNMPv3 applications

** RFC 3211** SNMPv3 USM

** RFC 3212** SNMPv3 VACM

** RFC 3512** SNMP configuring networks into view

** RFC 3514** SNMPv2 operations

** RFC 3515** SNMPv2 MIB

** RFC 3584** SNMP coexistence v1 v2 best practice

** RFC 3411–3415** Available from many sites, including

http://www.rfc-editor.org

See the Net SNMP FAQ, in /usr/share/doc/net-snmp-5.2.1/FAQ. Also see /usr/share/doc/net-snmp-5.2.1/README.snmpv3.


Pages 526, 527 explain the context example from rfc 2271 well. Actually, the example is changed slightly in rfc 3411.


James Boney, Cisco IOS In a Nutshell, O’Reilly, January 2002, 1-56592-942-X.