NET311
Computer Network Management

Dr. Mostafa H. Dahshan
Department of Computer Engineering
College of Computer and Information Sciences
King Saud University
mdahshan@ksu.edu.sa

Chapter 7

SNMP Management: SNMPv3

Objectives

- SNMPv3 features
  - Documentation architecture
  - Formalized SNMP architecture
  - Security
- SNMP engine ID and name for network entity
- SNMP architecture
  - Integrates the three SNMP versions
- User security model, USM
  - Derived from user ID and password
  - Authentication
  - Privacy
  - Message timeliness
- View-based access control model, VACM
  - Configure set of MIB views for agent with contexts
  - Family of subtrees in MIB views
  - VACM process

Key Features

- Modularization of document
- Modularization of architecture
- SNMP engine
- Security feature
    - Secure information
    - Access control

Notes

Architecture

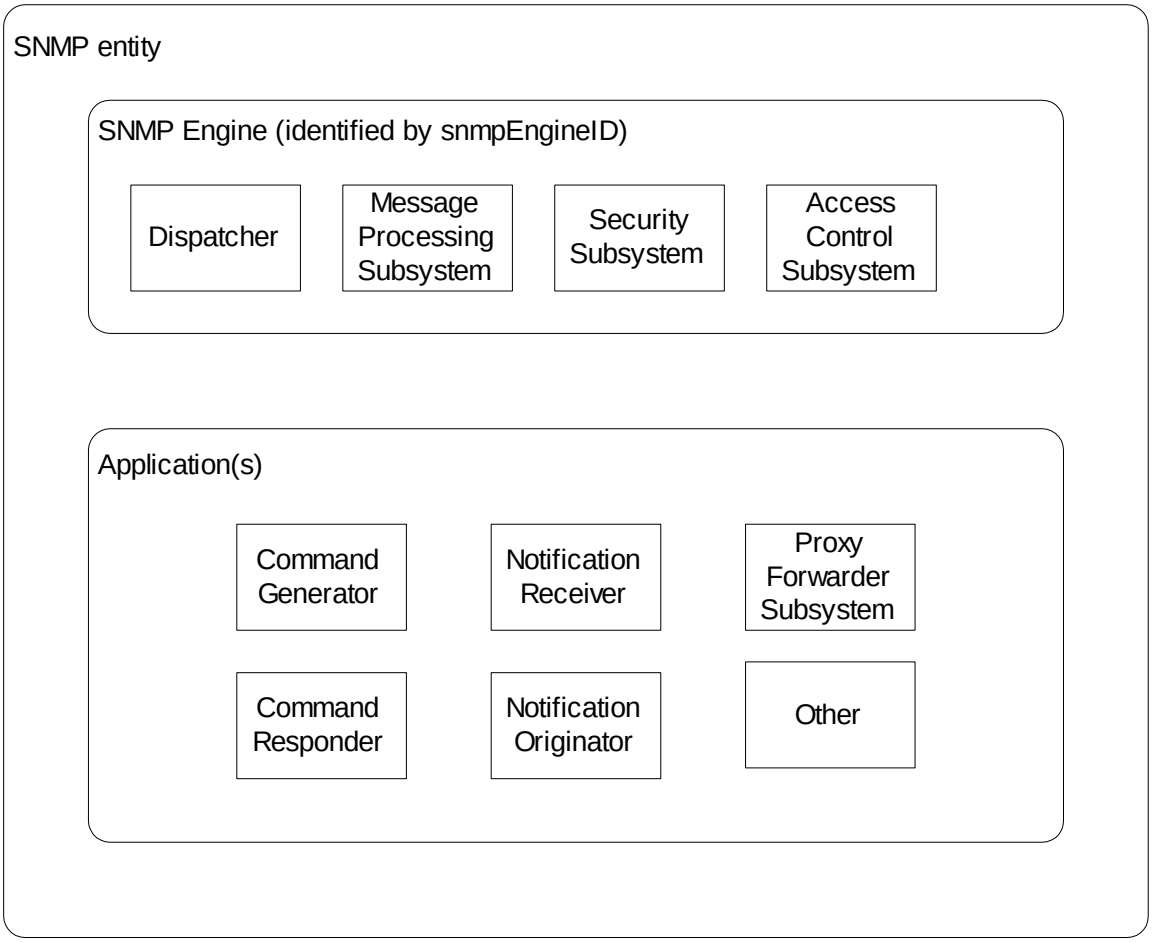

Figure 7.2 SNMPv3 Architecture

Notes

- SNMP entity is a node with an SNMP management element - either an agent or manager or both
- Three names associated with an entity
  - Entities: SNMP engine
  - Identities: Principal and security name
  - Management Information: Context engine

SNMP Engine ID

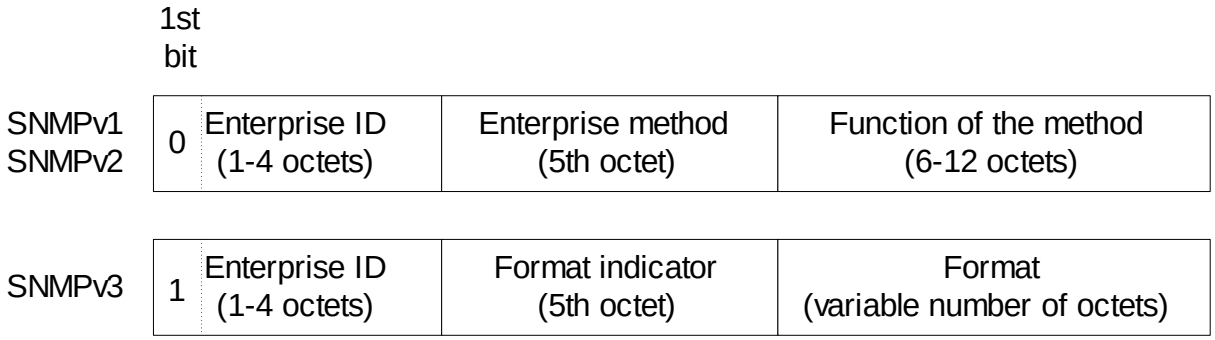

Figure 7.3 SNMP Engine ID

Notes

- Each SNMP engine has a unique ID: *snmpEngineID*
- Acme Networks {enterprises 696}
- SNMPv1 snmpEngineID '000002b8'H
- SNMPv3 snmpEngineID '800002b8'H
  (the 1st octet is 1000 0000)

•Engine ID is used wish hash function to generate keys for authentication and encryption.

SNMPv3 Engine ID Format 5th Octet

Table 7.2 SNMPv3 Engine ID Format (5th octet)

0	Reserved, unused
1	IPv4 address (4 octets)
2	IPv6 (16 octets) Lowest non-special IP address
3	MAC address (6 octets) Lowest IEEE MAC address, canonical order
4	Text, administratively assigned Maximum remaining length 27
5	Octets, administratively assigned Maximum remaining length 27
6-127	Reserved, unused
128-255	As defined by the enterprises Maximum remaining length 27

Notes

- For SNMPv1 and SNMPv2:
  - Octet 5 is the method
  - Octet 6-12 is IP address
- Examples: IBM host IP address 10.10.10.10

  SNMPv1: 00 00 00 02 01 0A 0A 0A 0A 00 00 00

  SNMPv3: 10 00 00 02 02 00 00 ... 00 00 00 0A 0A 0A 0A

SNMPv2 MIB

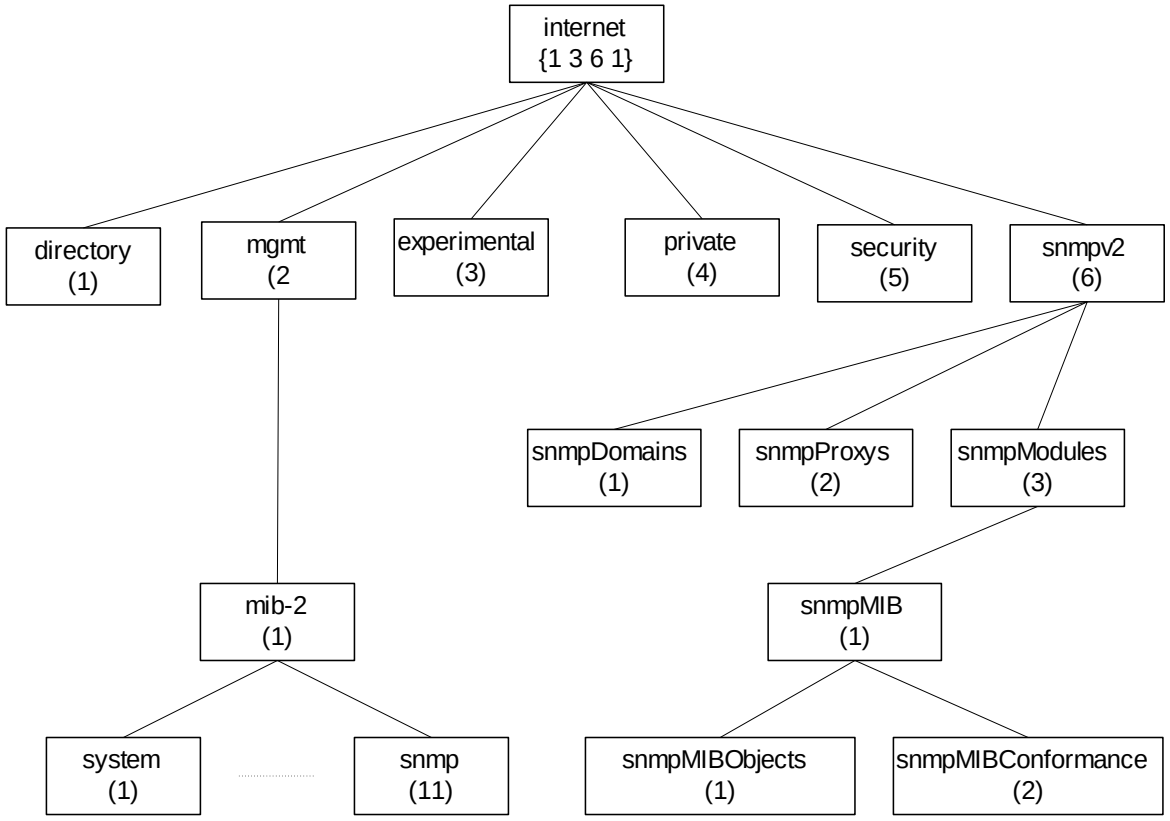

Figure 6.31 SNMPv2 Internet Group

Notes

- SNMPv3 MIB developed under snmpModules
- Security placeholder not used

SNMPv3 MIB

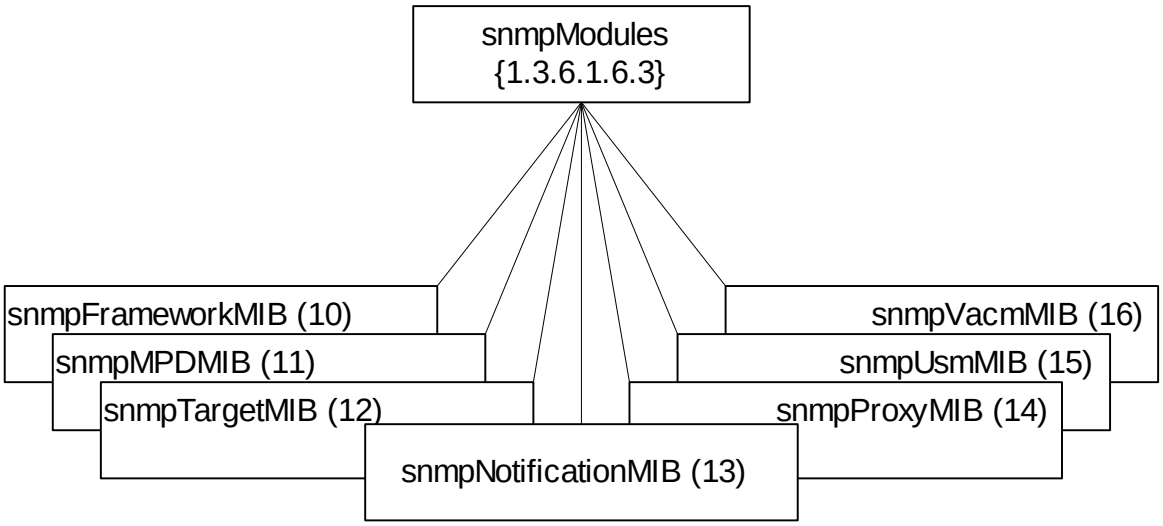

Figure 7.7 SNMPv3 MIB

Notes

- snmpFrameworkMIB describes SNMP management architecture
- snmpMPDMIB identifies objects in the message processing and dispatch subsystems
- snmpTargetMIB and snmpNotificationMIB used for notification generation
- snmpProxyMIB defines translation table for proxy forwarding
- snmpUsm MIB defines user-based security model objects
- snmpVacmMIB defines objects for view-based access control

SNMPv3 Message Format

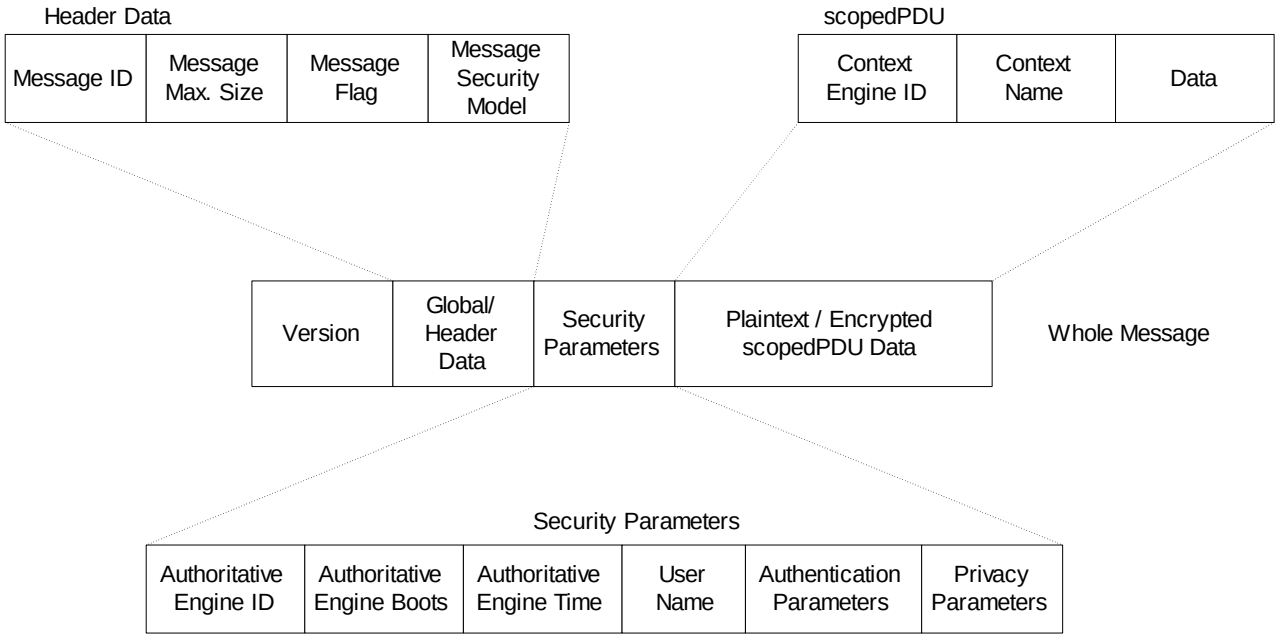

Figure 7.12 SNMPv3 Message Format

Notes

SNMPv3 Message Format

Table 7.7 SNMPv3 Message Format

Field	Object name	Description
Version	msgVersion	SNMP version number of the message format
Message ID	msgID	Administrative ID associated with the message
Message Max. Size	msgMaxSize	Maximum size supported by the sender
Message flags	msgFlags	Bit fields identifying report, authentication, and privacy of the message
Message Security Model	msgSecurityModel	Security model used for the message; concurrent multiple models allowed
Security Parameters (See Table 7.8)	msgSecurityParameters	Security parameters used for communication between sending and receiving security modules
Plaintext/Encrypted scopedPDU Data	scopedPduData	Choice of plaintext or encrypted scopedPDU; scopedPDU uniquely identifies context and PDU
Context Engine ID	contextEngineID	Unique ID of a context (managed entity) with a context name realized by an SNMP entity
Context Name	contextName	Name of the context (managed entity)
PDU	data	Contains unencrypted PDU

Security Threats

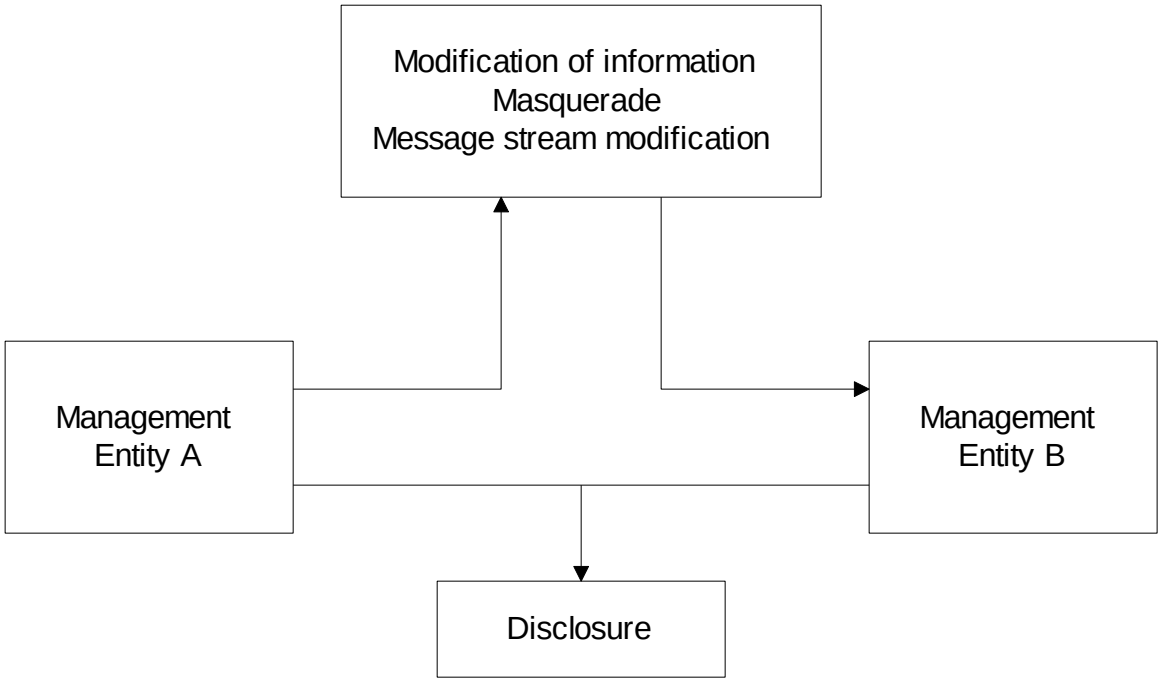

Figure 7.10 Security Threats to Management Information

Notes

- Modification of information: Contents modified by unauthorized user, does not include address change
- Masquerade: change of originating address by unauthorized user
- Fragments of message altered by an unauthorized user to modify the meaning of the message
- Disclosure is eavesdropping
- Disclosure does not require interception of message
- Denial of service and traffic analysis are not considered as threats

Security Services

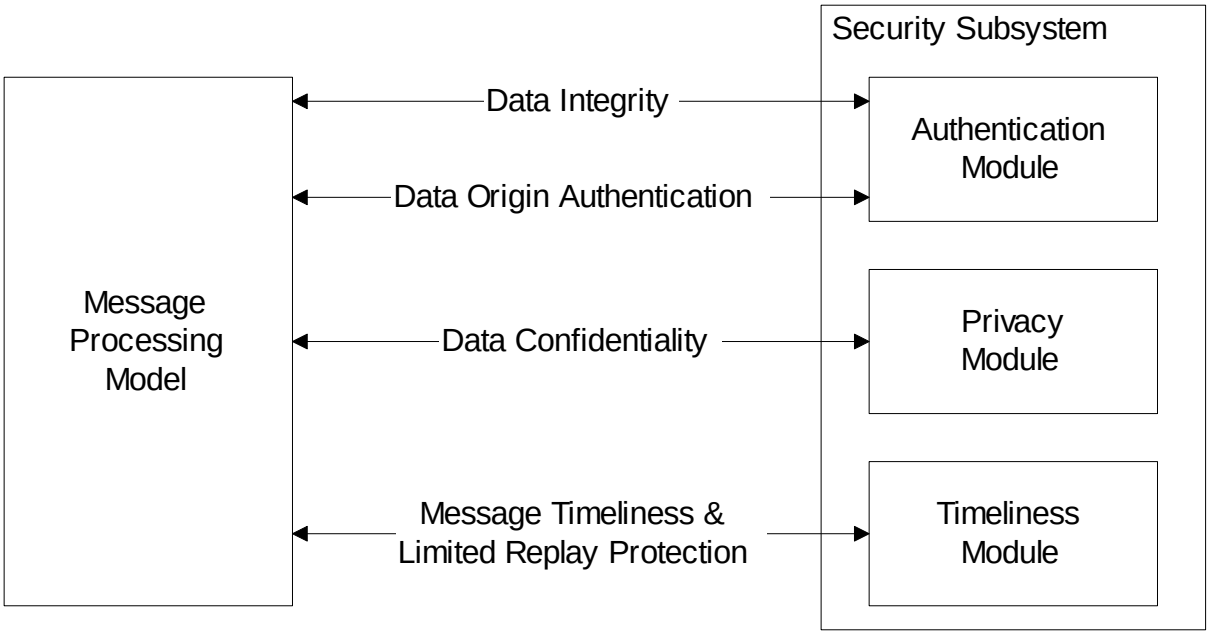

Figure 7.11 Security Services

Notes

- Authentication
  - Data integrity:
    - HMAC-MD5-96 / HMAC-SHA-96
  - Data origin authentication
    - Append to the message a unique Identifier associated with authoritative SNMP engine
- Privacy / confidentiality:
  - Encryption
- Timeliness:
  - Authoritative Engine ID, no. of engine boots and time in seconds

User-based Security Model

- Based on traditional user name concept
- USM primitives across abstract service interfaces
    - Authentication service primitives
        - authenticateOutgoingMsg
        - authenticateIncomingMsg
    - Privacy Services
        - encryptData
        - decryptData

Notes

Secure Outgoing Message

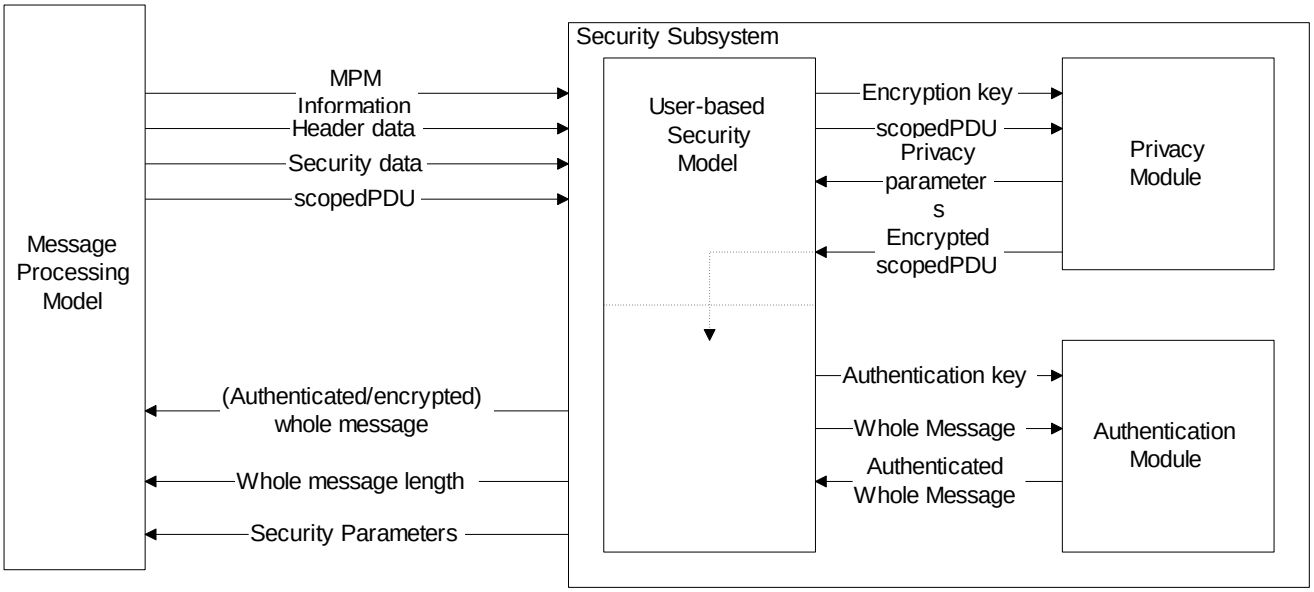

Figure 7.13 Privacy and Authentication Service for Outgoing Message

Notes

- USM invokes privacy module w/ encryption key and scopedPDU
- Privacy module returns privacy parameters and encrypted scopedPDU
- USM then invokes the authentication module with authentication key and whole message and receives authenticated whole message

Secure Incoming Message

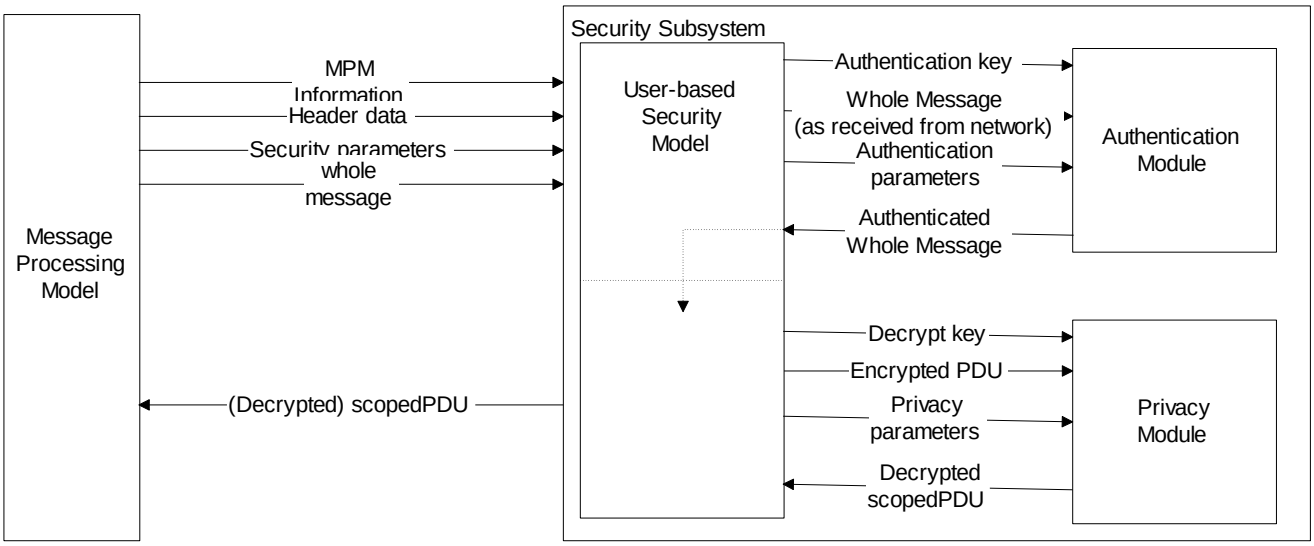

Figure 7.14 Privacy and Authentication Service for Incoming Message

Notes

- Processing secure incoming message reverse of secure outgoing message
- Authentication validation done first by the authentication module
- Decryption of the message then done by the privacy module

Security Parameters

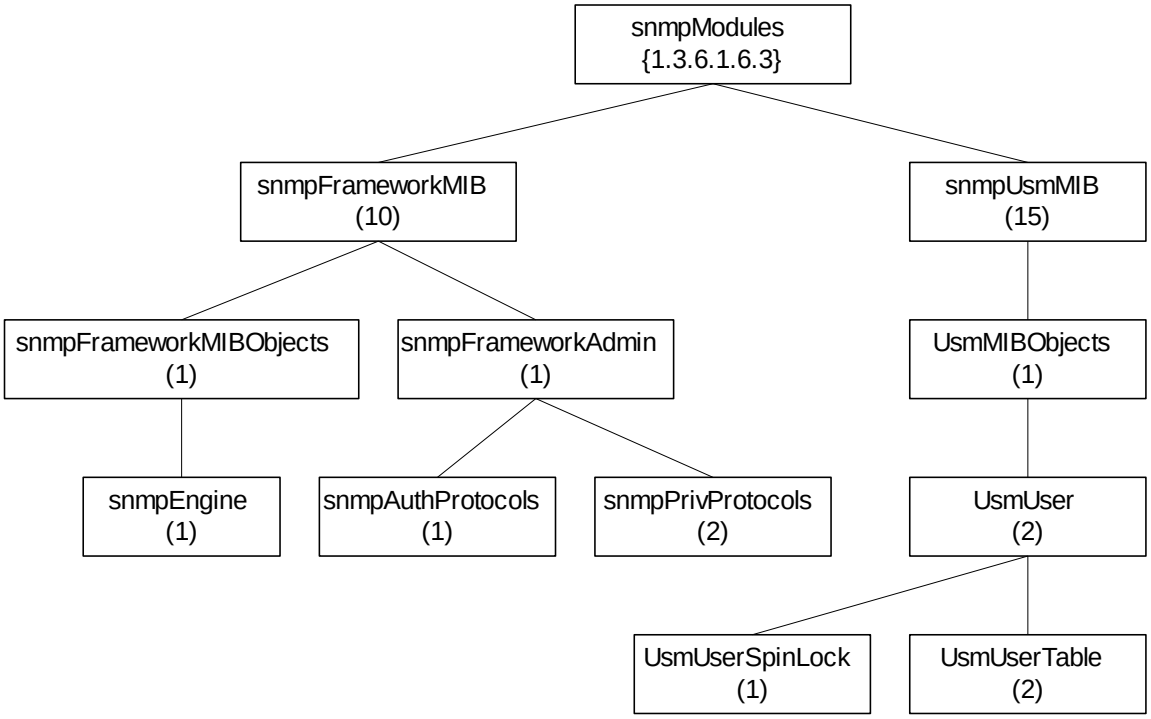

Figure 7.15 SNMPv3 MIB Objects for Security Parameters

Notes

Table 7.8 Security Parameters and Corresponding MIB Objects

Security Parameters	USM User Group Objects
msgAuthoritativeEngineID	snmpEngineID (under snmpEngine Group)
msgAuthoritativeEngineBoots	snmpEngineBoots (under snmpEngine Group)
msgAuthoritativeEngineTime	snmpEngineTime (under snmpEngine Group)
msgUserName	usmUserName (in usmUserTable)
msgAuthenticationParameters	usmUserAuthProtocol (in usmUserTable)
msgPrivacyParameters	usmUserPrivProtocol (in usmUserTable)

Privacy Module

- Encryption and decryption of scoped PDU (context engine ID, context name, and PDU)
- CBC - DES (Cipher Block Chaining - Data Encryption Standard) symmetric protocol
- Encryption key (and initialization vector) made up of secret key (user password), and timeliness value
- Privacy parameter is *salt* value (unique for each packet) in CBC-DES

Notes

Authentication Key

- Secret key for authentication
- Derived from user (NMS) ID and password
- MD5 or SHA-1 algorithm used
- Authentication key is *digest2*

Notes

Authentication Parameters

- Authentication parameter is Hashed Message Access Code (HMAC)
- HMAC is 96-bit long (12 octets)
- Derived from authentication key (*authKey*)

Notes

Encryption Protocol

- Cipher Block Chaining mode of Data Encryption Standard (CBC-DES) protocol
- 16-octet *privKey* is secret key
- First 8-octet of *privKey* used as 56-bit DES key; (Only 7 high-order bits of each octet used)
- Last 8-octet of *privKey* used as pre-initialization vector

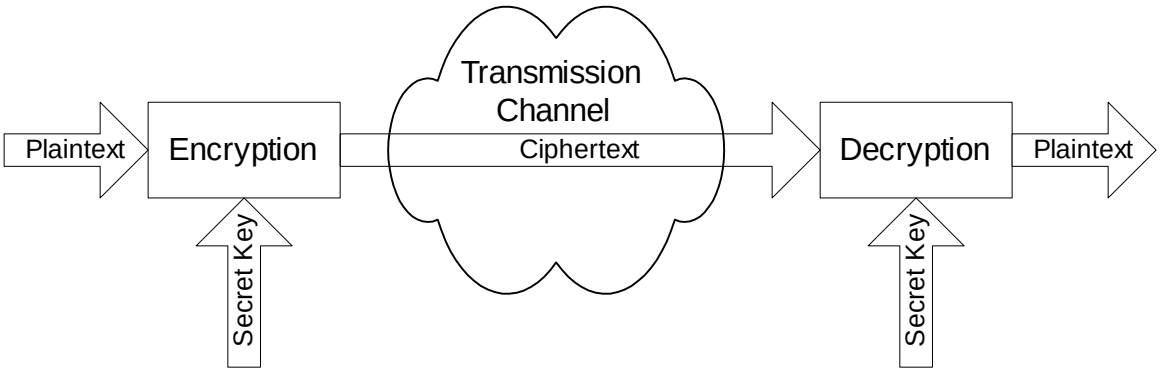

Figure 11.33 Basic Cryptographic Communication

Notes

Access Control

- View-based Access Control Model
  - Groups: Name of the group comprising security model and security name: In SNMPv1, is community name
  - Security Level
    - no authentication - no privacy
    - authentication - no privacy
    - authentication - privacy
  - Contexts: Names of the context
  - MIB Views and View Families
    - MIB view is a combination of view subtrees
  - Access Policy
    - read-view
    - write-view
    - notify-view
    - not-accessible

Notes

VCAM Process

Answers 6 questions:

1. Who are you (group)?
2. Where do you want to go (context)?
3. How secured are you to access the information (security model and security level)?
4. Why do you want to access the information (read, write, or send notification)?
5. What object (object type) do you want to access?
6. Which object (object instance) do you want to access?

Notes

VCAM Process

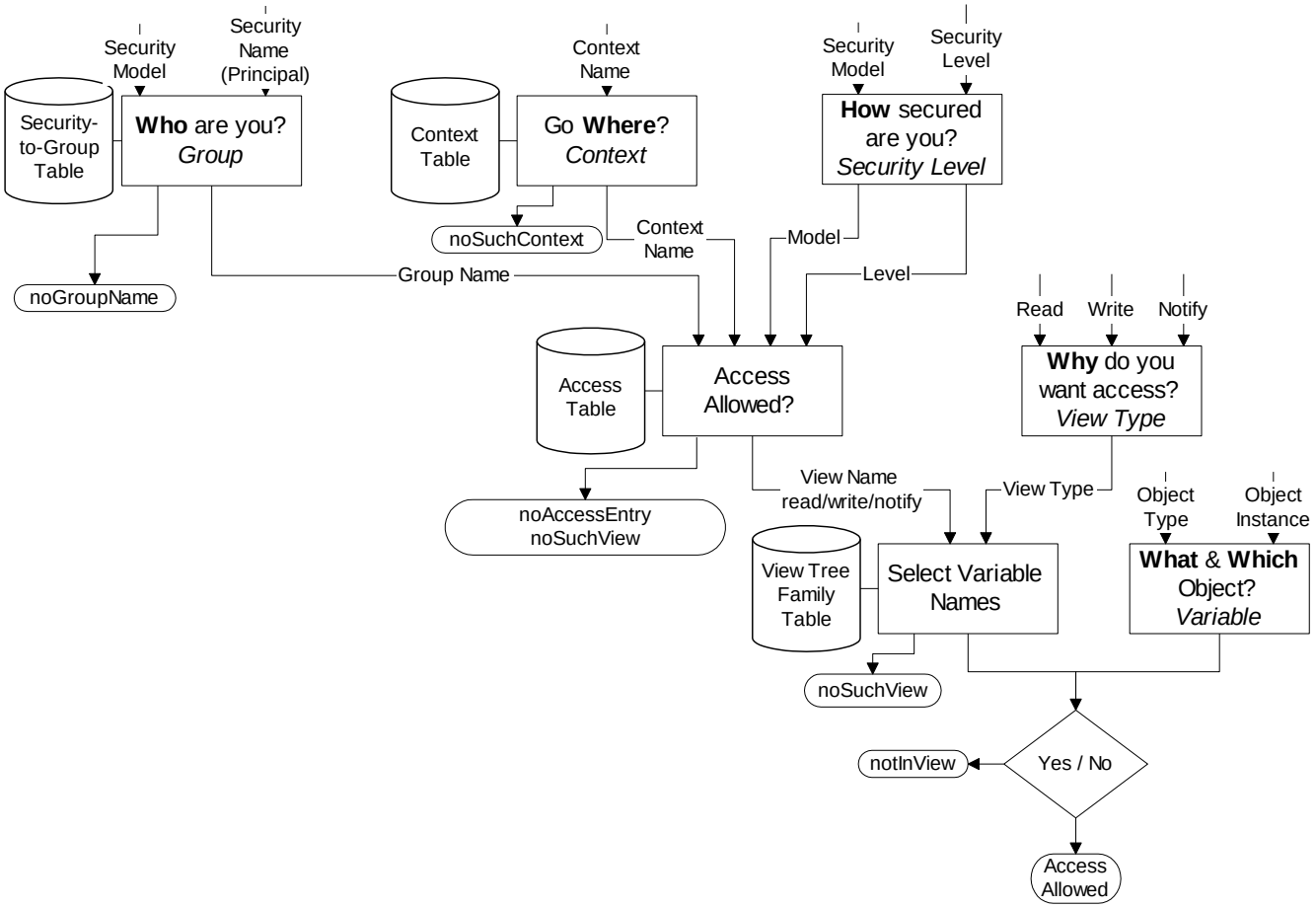

Figure 7.16 VACM Process

Notes

VACM MIB

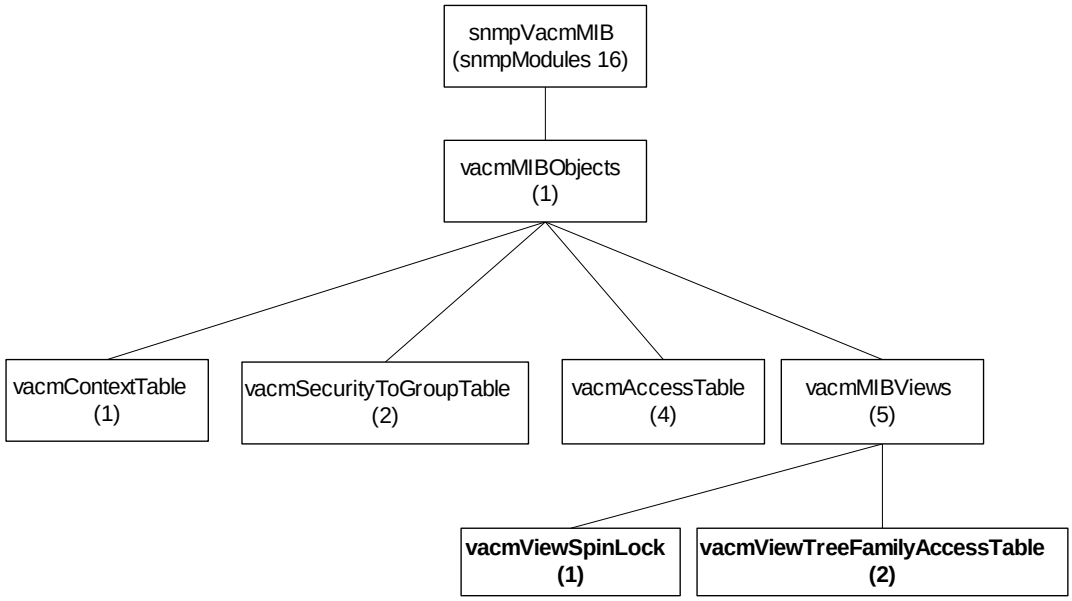

Figure 7.17 VACM MIB

Notes

- Four tables used to achieve access control:
  - Group defined by security-to-group table
  - Context defined by context table
  - Access determines access allowed and the view name
  - View tree family table determines the MIB view, which is very flexible

MIB Views

- Simple view:
  - *system* 1.3.6.1.2.1.1
- Complex view:
  - All information relevant to a particular interface – *system* and *interfaces* groups
- Family view subtrees
  - View with all columnar objects in a row appear as separate subtree.
  - OBJECT IDENTIFIER (family name) paired with bit-string value (family mask) to select or suppress columnar objects

Notes

VACM MIB View

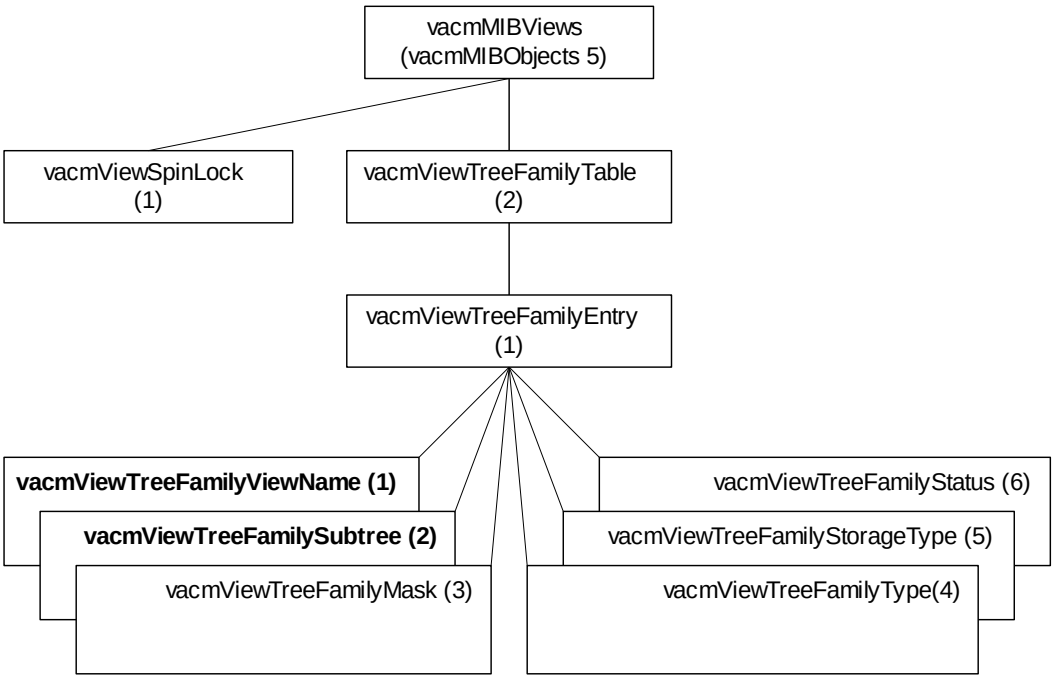

Figure 7.19 VACM MIB Views

Notes

Example:

Family view name = “system”
Family subtree = 1.3.6.1.2.1.1
Family mask = “” (implies all 1s by convention)
Family type = 1 (implies value to be included)