

16.1 Release Notes

This document includes release notes for SNMP Research Agent, Manager, and End-user products which include the following products:

Agent Products

- Critical Applications Subagent
- Critical Applications MIB Subagent
- DR-Web™ EMANATE®/Lite Agent
- EMANATE® Agent Development Toolkit
- DR-Web™ EMANATE® Agent
- DR-Web™ EMANATE® Binary Tools
- EMANATE® Master Agent
- EMANATE® with Proxy Option
- EMANATE® Protocol Interface Component Subsystem (EPIC)
- EMANATE®/Lite
- EMANATE®/Lite with Proxy Option
- Event MIB Subagent
- File System Monitor MIB Subagent
- Host Resources MIB Subagent
- Host Resources MIB Subagent
- Logfile Monitor MIB Subagent
- Native Agent Adapter
- Native Subagent Adapter
- Packaged Agent System
- DR-Web™ Packaged Agent System
- RMON MIB Subagent

Schedule MIB Subagent
Service Monitoring MIB Subagent
System-Level Applications MIB Subagent

End-User Products

CIAgent®
EnterPol®

Manager Products

SNMP Asynchronous Request Library
BRASS™ Management Application Development Toolkit
DR-Web™ Manager
Mid-Level Manager
SNMP Security Pack

Utilities

Cross-Development Tools
SNMP Libraries, Utilities, and MIB Tools

Upgrades for all Products

Documentation

All manuals in pdf format have been improved by the addition of bookmarks which help to make our pdf manuals a quick and easy reference tool. The manuals are included in pdf format on the product CD. All SNMP Research products are shipped with a Quick Start guide instead of the printed manuals. The printed manuals are available upon request, but are no longer included in the with the product.

Internet Standards

Since the release of version 15.4, several Internet standard documents have been published or updated by the Internet Engineering Task Force (IETF). SNMP Research has upgraded its products to meet these latest versions of the Internet standard documents (Requests For Comments, or RFCs). The latest additions and updates to IETF RFCs are as follows:

- RFC3410 (Informational): Introduction and Applicability Statements for Internet Standard Management Framework (December 2002)
- RFC3411: An Architecture for Describing SNMP Management Frameworks (December 2002)
- RFC3412: Message Processing and Dispatching (December 2002)
- RFC3413: SNMP Applications (December 2002)
- RFC3414: User-based Security Model (December 2002)
- RFC3415: View-based Access Control Model (December 2002)
- RFC3416: Version 2 of SNMP Protocol Operations (December 2002)
- RFC3417: Transport Mappings (December 2002)
- RFC3418: Management Information Base (MIB) for the Simple Network Management Protocol (SNMP) (December 2002).

Operating Systems

SNMP Research's products have been updated to support newer versions of existing supported platforms. For a current list of supported platforms, please visit our Web site at <http://www.snmp.com>.

Some of the most popular platforms include:

- Solaris 8 and Solaris 9
- Microsoft Windows NT 4.X, Windows 2000, Windows XP, and Windows Server 2003
- HP-UX 11iv1 on PA-RISC processors, 11iv2 on Intel Itanium processors
- Red Hat Linux 7.2 and 9.0
- VxWorks
- OSE

SNMP Research has added the following platforms our list of supported platforms for some of our products:

- IBM AIX 5.2
DR-Web™ Agent
EMANATE® Source
EMANATE® Master Agent binary
EMANATE® Subagent Development Toolkit
EPIC
Asynchronous Request Library
BRASS™
BRASS™ Management Application Development Kit
- INTEGRITY
EMANATE® Master Agent binary
EMANATE®/Lite
- VOS
BRASS™

Upgrades for Agent Products

SNMP Research provides agent products for End-users and Original Equipment Manufacturers (OEMs). Agent products are available for both open and embedded systems. SNMP Research's agent products are built upon the world's leading EMANATE® agent technology. We offer options such as Web accessibility to agent information, run-time extensibility, and compile-time extensibility. The following list of items reflects a brief overview of some of the more significant updates that affect SNMP Research's Agent products.

New Features and Enhancements

- System-Independent Method Routines are generated much smaller by default.
The **postmosy** tool that generates the C code for agent extensions (including Subagents) produces `v_` routines that are much smaller by default. These changes reduce the code size of the EMANATE® Master Agent and Subagents and the EMANATE®/Lite Agent and SNMP Agent Extensions. The system-independent method routines can be generated in the old style using the new command-line argument `-classic.v`.
- Example configuration files have been improved.
The default `snmpd.cnf` configuration file has been improved to better map SNMPv3 configurations with real world customer environments.

- The size of the `snmpd.jnk` file has been limited.
- Global static objects have been moved into a separate module.
A single API is provided with type, pointer to storage, and read/write arguments to allow global variables to be modified by subroutine calls.
- A utility has been documented which creates a configuration file via command-line interface.
The `snmpdcfg` program is a utility that is invoked from the command line and provides users a simple way to create an initial agent configuration file. The user chooses options and answers questions, and upon exit, the configuration file is generated and saved.
- The SNMPv3 Configuration Wizard is now provided.
This is a Java-based tool to configure users, passwords, MIB views, communities, groups, and many other aspects of SNMPv3 Configuration.
- The `snmpEngineID` for the local agent is now stored in the configuration file.
This implementation allows a method of generating an `engineID` independently from the entity's IP address. It is a benefit for users of DHCP environments.
- MIB-2: New IANA type support.
Support for the newer IANA interfaces has been added and include default speeds. Interfaces table now recognizes the new objects, `ifSpeed` of `elxl` and `iprb` interfaces which are installed on a Solaris/Intel machine.
Support is added for the `eri` interface. Minor updates have been made to the `qfe` interface support. The `qfe` interface has been changed to `kstat` to get interface speed (which is mandatory for `eri` interface - streams `ioctl` no longer works). Minor fixes are included so the product will compile on Solaris 5.5.
- Undetermined loopback interface speed has been reduced from a warning message to a trace message on Solaris machines to prevent spurious "could not locate `ifspeed` for `lo0`" messages from being issued.
- The agent can bind to a specified trap sending port rather than using a transient port.
The option `-send_trap_port <PORT>` has been added to EMANATE[®] and EMANATE[®]/Lite to allow the SNMP Agent to bind to a specific port for sending traps.
- Private MIB objects have been added to assist remote support.
Agent version, and operating system on which the agent was built is now supported in a Private(99) MIB object.
- A "ready" phase in **Set** processing that occurs once per row makes it possible to do full-row based ready checks.

- Agent APIs for USM and VACM management have been added.

The `confapi.c` routines have been added to allow management of the USM and VACM directly within the agent code. Available only to recipients of EMANATE[®]/Lite or the EMANATE[®] Subagent Development Toolkit in source code form.

Bug Fixes

- Agents now disregard inactive rows.
- **DPRINTF()** has been modified to protect against recursive calls and the possibility of buffer overflow problems.
- The EMANATE[®] Master Agent has been modified to protect against incorrectly ordered OIDs from Subagent.

To avoid an infinite loop in the EMANATE[®] Master Agent, if a Subagent returns an OID less than or equal to the requested OID, the Master Agent ignores that result from that Subagent (and generates an **APWARN** message).

- Critical Application Subagent no longer sends traps while **critagt** process is starting.
Previous Behavior: The **critagt** sent traps for all monitored processes when the **critagt** process started, even if they were already running or incorrectly showing a change in process status (from down to up).

Current Behavior: By default, traps are disabled while the **critagt** process is starting. **Traps** are enabled when the **critagt** process completes initialization.

To simulate the previous behavior, the new command-line option `-traps_on_start` enables traps to be sent while the **critagt** process is starting.

- EMANATE[®] thread blocks have been coded as condition variable blocks.

Some operating systems show odd behavior with thread scheduling. This has been addressed by replacing the current blocking structure (which is to simply yield control to another thread) with a condition variable block, in which another thread (presumably the Master Agent thread) can explicitly determine which thread will execute next.

- Previously built `.my` files are not rebuilt on every compilation.

Dependency checking has been addressed on some operating systems so that the `.my` files in `mibs/common` and the `snmpinfo.dat` file are not rebuilt on every make.

- Root pointers race condition.

There is a potential race condition when a Subagent registers while processing a request with multiple variable bindings. This has been addressed by queuing subagent registration/de-registration requests until all outstanding **GetNext** or **GetBulk** requests have been completed.

- Incomplete **snmpTargetAddrTable** and **snmpTargetParamsTable** ready checks.
Proper checks are performed for the **snmpTargetAddrTable** and **snmpTargetParamsTable** to ensure that all of the necessary objects are being **Set** in the row before the row's status can be made active.
- Code for **vacmSecurityToGroupStatus** results in **notReady** error.
vacmSecurityToGroupStatus results in an error of **notReady** instead of **notInService** if **vacmGroupName** has not yet been set (RFC3415).
- **StorageType** implementation has been updated to match V2-TC.
The V2-TC MIB document is updated from RFC1903 to RFC2579 so that a **Set** request on rows with **StorageType** of **permanent** will return an error of **wrongValue** rather than **inconsistentValue**.
- **usmUserEntries** with **permanent** type cannot be set.
SNMP Research does not permit **Set** requests to complete on **usmUserEntries** that have a storage type of **permanent** per RFC2574.
If the **vacmAccessStatus** object of a row in the **vacmAccessTable** does not have the value **active(1)**, the row will not be used by the agent to authorize **Get** requests, **Set** requests, or the sending of **Traps**. Previously, the agent ignored the value of **vacmAccessStatus**.
- Calling **SearchTable()** for a next request returns valid bit of column object.
When the **SearchTable()** routine is called to do a search for the "next" object, **SearchTable()** routine returns the valid bit of the columnar object being searched. In the previous release, the "next" search request returns NULL when the valid bit is turned off. In the current release, when the valid bit is turned off, a "next" search request returns the next valid columnar object.
- EMANATE[®] Master Agent connect attempts should happen at a controlled rate.
The Subagent only tries to connect once every few seconds (with exponential backoff) for better performance because some operating systems hang when waiting longer than one second for **connect()** to fail.
- On UNIX, loading an EMANATE[®] Tightly Coupled Subagent with the load command no longer causes a bogus error message from the load program.
- In the RFC2790 Host Resources MIB implementation for Linux, earlier releases of the **hostagt** stopped listing processes in the **hrSWRunTable** when it encounters a TUX process that contains a " " character. The problem has been corrected.
- The implementation of **snmpTargetAddrTMask** was changed to comply with RFC2576.

Previous behavior: the `snmpTargetAddrTMask` object is set to “255.255.255.255” in the default configuration file. If one modifies this to “-”, it will not be accepted. If one attempts to set the `snmpTargetAddrTMask` object to “”, the configuration file entry is set to “0.0.0.0:0.” As a result, a request from any host will be accepted.

Current behavior: If the length of the mask is 0, the mask is treated as if all its bits were 1 (i.e., 255.255.255.255) and its length is equal to the length of the corresponding value of `snmpTargetAddrTable`.

- The system independent test method routine has been modified for efficiency.

During the processing of a **Set** request, the system dependent get method routines needs to be called to determine if the instance exists.

Previous behavior: The get method routine could be called numerous times per instance if the **Set** request is to change multiple objects in the same family.

Current behavior: The get method routine is called only once per instance when the **Set** request is to change multiple objects in the same family.

- There are two new functions available to customers of full source code agent products EMANATE and EMANATE/Lite. These are:

```
SR_UINT32 GetDefaultV1TrapAgentAddr(void);
```

This function allows the caller to see the value of `agent-addr` that will be inserted into the message header of an outgoing SNMPv1 trap message. The value is an IPv4 address expressed as four bytes in network byte order.

```
SR_UINT32 SetDefaultV1TrapAgentAddr(SR_UINT32 ipAddr);
```

This function allows the caller to change the value of `agent-addr` that will be inserted into the message header of an outgoing SNMPv1 trap message. The function returns the previous value. The value is an IPv4 address expressed as four bytes in network byte order.

- When username and password are supplied at the browser and access to DR-Web™ is denied, the reason for the denial can be seen the agent’s debug output if the `-apconfig` command-line was supplied at agent startup.
- DR-Web™ made the assumption that an octet string with a length restriction of [0..255] must be a display string. This prevented a user from entering values for `snmpTargetAddrTMask` and other similar objects in a **Set** page in the browser. This behavior has been corrected.
- The `hrStorageNetworkDisk` was implemented on Solaris and Linux for RFC2790 Host Resources MIB.
- Sometimes the load program for EMANATE® Tightly Coupled Subagents would print an erroneous message saying, “I killed the master agent.” This message has been removed.

- A memory leak was fixed in **mib2agt** for Unixware 7.1.
- Tightly Coupled Subagents are now supported in the QNX version of EMANATE.
- **GetSubagentEvent()** was modified to handle **select()** errors caused by a bad file descriptor.
- A bug was fixed in the unload utility for EMANATE[®] that would sometimes cause a Tightly Coupled (shared-library) Subagent to not really be unloaded.
- If the *-light* option is used with **postmosy**, a field in the `<base>oid.c` array initializer is set to NULL. The agent would sometimes refer to this field even when the code was compiled with **-DLIGHT**. This was been corrected by adding more **#ifndef LIGHT** ifdefs around the relevant sections of code in `objectdb.c`.
- In EMANATE[®]/Lite, the default context entry is automatically added to the **vacmContextTable** only if it is not already configured. This eliminates a warning message during agent startup.

Upgrades for Management Tools

New Features and Enhancements

- ARL table code now handles incomplete responses from an agent.
- The **engineID** discovery process has been improved.

When performing **engineID** discovery, we are now sending an empty VarBind list, and a populated list immediately thereafter, instead of filling in the VarBind list on the initial request(s). The variable **snmpEngineID** is a local SNMP entity's **snmpEngineID** and is either automatically generated upon startup or manually configured.

Note: If the value of **snmpEngineID** is changed after the plain text passwords are converted to localized keys, then the SNMPv3 users will not be able to access the agents using the passwords.

- Improved handling of **engineID**.

BRASS[™] is enhanced to recover when an agent changes its **engineID** (which can happen on a redundant agent startup). SNMP Security Pack will still drop one packet when this occurs, but will recover after that packet. If the **engineID** for this host has changed, then the localized authentication and privacy passwords are no longer useful. The authentication and privacy passwords are cleared and a function is forced to try to find new ones. This update fixes the error caused by the "wrong engine ID" code. If this case involved an agent restart, chances are that the right **engineID** has been put in the cache.

The value of `SNMP_ENGINE_ID_SRC` determines the method for generating a local SNMP entity's `snmpEngineID`. The `snmpEngineID` is either automatically generated based on an algorithm, or the manager or agent always uses the manually configured `snmpEngineID` that is stored in the configuration file.

- ARL allows multiple session arguments.

The following define statement allows the user to specify the number of different `SESSION_ARGS`.

```
#define ARL_SA_SESSION_ARG_COUNT    16
```

This feature may be used, for example, to **malloc** an array of `SESSION_ARGS` with enough array elements to contain all `SESSION_ARGS`. Note that if a `SESSION_ARG` is added to the list, this counter should be increased.

- BRASSSM on Solaris operating systems: SIGPIPE handling changed between versions 2.5 and 2.6.

BRASSSM Clients now explicitly ignore SIGPIPE due to the difference in SIGPIPE signal behavior between Solaris versions 2.5 and 2.6 through 2.8.

- **brassagt** has been renamed to **brassd**.
- The BRASS Server assumes *-nosubagent*, and only tries to connect to an EMANATE[®] Master Agent when *-subagent* is specified.

With Release 15 and earlier BRASSSM Subagent, the default behavior is to try to connect to the EMANATE[®] Master Agent unless the *-nosubagent* command-line option is specified. The default behavior has been changed so that the BRASS Server will **not** try to connect to the EMANATE[®] Master Agent unless the *-subagent* command-line option is specified. This change prevents unnecessary delay times when `brassd` is hanging on Master Agent connect failures.

- The instance portion of the `OID` parameter is no longer required in some functions.

Previously, the following functions would fail if the argument was a fully-qualified `OBJECT IDENTIFIER` containing instance information. However, SNMP Research has changed an invocation of `GetMIBNodeFromOID()` to `LP_GetMIBNodeFromOID()` in the following functions which makes the functions easier to use, because it is no longer required to strip off the instance portion of the `OID` parameter. The following functions are affected:

GetEnumerationsFromOID()

GetIndexEntriesFromOID()

GetSizeSpecFromOID()

GetRangeSpecFromOID()

GetTableColumns()

- **InstToVarBind()** function has been added.

InstToVarbind() function parses a table **OBJECT IDENTIFIER** and builds a **VarBind** list containing the objects that are the index columns for that row, what some SNMP books call the “Grant’s Tomb” objects.

This works even if the index columns are not-accessible. An algorithm parameter allows the user to choose to convert all indices, the **th** index, a particular index whose **OBJECT IDENTIFIER** specified with a search key, the accessible indices, the not-accessible indices, the indices that are in the same table, or by indices in another table (an **AUGMENT**-ed table).

- Object Identifiers can now be processed in a human-friendly format using new routines.
- Asynchronous Request Library and **BRASSTM** utilities now support **BITS** values
- TCP client connections are only allowed from the localhost.

Security has been patched so that the **BRASS** Server will, by default, only accept TCP client connections from the localhost. (The old default allowed TCP client connections from any host.) An environment variable may be set to change the default behavior.

- Asynchronous Request Library and **BRASSTM** now include configuration APIs.

The Application Programming Interfaces contained in **confapi.c** are now included with Asynchronous Request Library and **BRASSTM** products. These APIs allow developers to configure the Loosely Coupled, **mgr.cnf**. In addition, the SNMP Configuration chapters have been included in the developer documentation. Available only to recipients of **BRASSTM** Management Application Development Toolkit in source code form.

- Thread-Safe **BRASSTM**

A new version of **BRASSTM** includes Application Programming Interfaces that support the use of re-entrant functions.

- Notification Filter Message Handler Functions

SNMP Research has added the **ArISetAutoNotificationHandler()** routine which allows greater flexibility to set up notification filters. Other related functions have been added. However the user only needs to use **ArISetAutoNotificationHandler()** in order to quickly and easily set up a notification filter.

Bug Fixes

- **mgrtool** now correctly reports indices for augmented tables in the **snmpinfo.dat** file.
- **BRASSTM** memory leak on application shutdown.

A memory leak existed in **BRASSTM** code when an **ARL** application is shut down and then re-initialized without explicitly closing the log (using **CloseLog()**).

- **brassd** failed upon startup when `mgr.cnf` was missing.
If `mgr.cnf` does not exist, the BRASS Server now creates a default one.
- **snmpEngineID** generation in ARL and BRASSSM
The **snmpEngineID** currently computed by ARL and BRASSSM now uses the **snmpTrapPort** rather than the **snmpPort** in byte 8. The **snmpEngineID** is also now stored in `mgr.cnf`.
- **GenerateManagerDefaults()** (`arl/gen_mgr.c`)
The appended newlines and end of files for **usmUserNames** and passwords entered using a command line entry are now correctly written to `mgr.cnf`.
- The function **InitializeTrapRcvIO()** now returns an error if call to **bind()** fails.
- The **SimpleSleuth** discovered two problems in the BRASS Server: a potential memory leak, and an error that would result in **brassd** crashing. Both problems have been fixed.
- Changes were made to the ARL libraries so fix a bug in the creation of outgoing **Traps**. **Traps** with time ticks were sent as `INTEGER_TYPE` rather than `TIME_TICKS_TYPE`. Affects Asynchronous Request Library and BRASSSM.
- A bug was fixed in **ArlSendRequest()** when the fields `and` (used in **GetBulk** requests) are overloaded for `error-status` and `error-index` values (used in non-**GetBulk** requests).

Upgrades for Core Libraries

Core Libraries are source code tools that provide the foundational routines used as the building blocks for other products. Core Libraries are written to support SNMPv1, SNMPv2c, SNMPv3, and all MIB variables that conform to the Internet Standard Structure of Management Information (SMI). The following list of items reflects a brief overview of some of the more significant updates that affect SNMP Research's core libraries.

New Features and Enhancements

- Directory and code layout analysis and changes.
Our code base has been split to reduce unnecessary files being linked into products that do not need those particular features. This has reduced the executable code space. Specifically, the functions inside `libsnmp.a` and `liblogbook.a` were reordered into two new directories, `engine` and `devkit`.
- Authentication and encryption code have been separated.

Encryption modules have been moved so that they are in a separate library to prevent export issues with the binary versions of the EMANATE[®] Master Agent Development Toolkit and the BRASS[™] Management Application Development Toolkit.

- MIB directories have been updated.

Updated the `mibs` directory and subdirectories, `mibs/rfc`s and `mibs/common`, to remove obsolete, unimplemented, and ancient enterprise documents.

- SNMPv2* has been removed from code.

This is a housekeeping and code maintenance change.

- Maximum Octet String Size is redefined.

`MAX_OCTET_STRING_SIZE` has been redefined to be the “maximum packet size minus two” in order to prevent buffer overflow problems.

Bug Fixes

- **ConvertToDisplayString()** now handles zero-length OctetStrings.

ConvertToDisplayString() now allows a zero-length OctetString on systems that implement **malloc** by returning NULL when 0 is passed as the size to **malloc**.

- SNMP Research’s `md5.h` conflicts with FreeBSD’s.

The `md5.h` header file has been renamed to `sr_md5.h` to prevent type conflicts when SNMP Research’s source code product is built on FreeBSD 4.3.

- Missing services entry message has been reduced from **error** to **warning**.

The warning generated when there is no `snmp` entry in `/etc/services` has been reduced to a lower log level from **error** to **warning**.

- In general, the clean targets in Makefiles do a better job of removing files. This was only a problem with a few products and only on a few platforms.

Upgrades for MIB Tools

- When merging MIB trees, only parts of the **OTE_TREE_ELE** structure was being copied. The entire contents of the structure is now copied.

Agent MIB Tool Enhancements

postmosy

- **postmosy** now creates prototypes for routines that are generated with the *-traps* option.
- **CloneOctetString()** for **Augmented** objects are now being added correctly to the **Clone_Entry()** routine for **Augmented** tables.
- Enumerated labels that contain dashes are now converted into underscores.
- Negative default values now generate the correct code.

The call to **isdigit()** for a string **-1** if the default value is **-1** previously caused **postmosy** to recognize the default value as a string.

- Augmented tables with **row-status** are now generated correctly.

When a table is augmented by another table which has the **row-status** object, the **v_<base>.test()** routine previously did a NULL comparison on the augmented table entry, however the augmented table entry is not defined.

- **Postmosy** now creates the **<family>_set()** routine in addition to the **<family>_DeleteCallback()** routine when **row-status** is set to **readOnly**.

Previously, when a **row-status** object was set to **ReadOnly**, the **postmosy** generated code created the **<family>_DeleteCallback()** routine but did not create the **<family>_set()** routine.

- **postmosy** also includes the functionality of the *-merge* command line option, which is the same as the **mgrtool -merge** command line option.
- The MIB tool **postmosy** provides better error messages on enumerations when reading an input file. For example, now the error message identifies the MIB object where the error occurred.

Previous behavior:

```
get_snmpinfo_enum: error in input file format
```

Current behavior:

```
get_snmpinfo_enum (marsMcGrpRowStatus)
```

Manager MIB Tool Enhancement

mgrtool

- The MIB tool **mgrtool** provides better error messages on enumerations when reading an input file. For example, now the error message identifies the MIB object where the error occurred.

Previous behavior:

```
get_snmpinfo_enum: error in input file format
```

Current behavior:

```
get_snmpinfo_enum (marsMcGrpRowStatus)
```

- **mgrtool** correctly places curly braces.

If a table entry name has a dash in it, then previously, **mgrtool** would not put into the `snmpinfo.dat` file the index information with surrounding curly braces, which is supposed to look like this:

```
systemM12_LockEntry    1.3.6.1.4.1.290.3.1.96.1.15.1  Aggregate not-accessible
{
    systemM12_LockNum
}
```

The dashes were being converted to underscores in one **mgrtool** data structure but not another.

The MIB tool called **mgrtool** has two new command line options and new exit codes which are as follows:

- *-merge_report <filename>*

The command-line option *-merge_report <filename>* will cause **mgrtool** to print the merge errors and warnings to a file in addition to printing them to the screen.

- *-merge*

-merge behaves in the following manner when there is a conflict in the MIB document being merged.

- **Enumeration Conflicts:** If two MIB modules being merged contain the same object with a different set of enumerations, the following rules apply:

1. If the enumerations from one MIB module are a superset of the other, the superset will be used.
 2. If the enumeration names change, but the numbers are the same, the list from one of the modules will be used.
 3. Other conflicts will cause an error message to be printed and the merge will fail.
- **Syntax Conflicts:** If two MIB modules being merged contain the same object with a different SYNTAX, an error will be printed and the merge will fail. Note that the following pairs are considered the same and will not cause an error:
- Counter and Counter32
 - Gauge and Gauge32
 - INTEGER and Integer32 (as long as the INTEGER has no enumerations)
- **Size and Range Conflicts:** For OCTET STRING types with a different size or INTEGER types with a different range, the widest range will be used such that the result range will be the lowest lower bound and the highest upper bound, even if boundaries come from different object definitions.
- **Index Conflicts:** Any change in the index clause is an error condition.
- **Access Conflicts:** If two MIB modules being merged contain the same object with a different ACCESS or MAX-ACCESS, the most open access will be used.
- **OID Conflicts:** In the case where two objects have a different name, but the same OID, both names will be output to `snmpinfo.dat` with the same OID and the same attributes. The attributes will be determined by the rules above.
- **Exit Codes** The exit codes from `mgrtool` are as follows:
- 0: No Errors or Warnings
 - 1: No Errors, but at least one Warning, processing completed successfully
 - 2: Merge Error, processing did not complete
 - 1: Error

The mibgc MIB Compiler

The **mibgc** is a new mosy-like MIB compiler that parses MIB documents using a well-defined grammar. While providing functionality similar to mosy, **mibgc** supports all MIB documents that adhere to the current SMI standards. It provides clear error descriptions for MIB syntax and formatting, and even attempts to fix errors so that compilation does not stop. **mibgc** provides these additional features:

- Understands SMIV1 textual conventions.
- Supports SMIV2 to SMIV1 and future SMI conversions.

- Generates useful warnings for MIB formatting error and attempts to proceed with compilation unless the error is irrecoverable.
- Parses, but generates no output for MIB **AGENT-CAPABILITIES** statements.
- Processes multiple MIB documents contained within one input file.
- Provides options to ignore non-compliant MIB formatting with more flexibility than mosy currently allows.
- Utilizes a well-defined grammar: The parsing routines are based upon clearly understood parameters.

The mibgp MIB pre-processor

The **mibgp** MIB pre-processor is included with the SNMP Research product in addition to the **premosy** processor. It includes the following features:

- Overwrite protection: **mibgp** creates different versions of the generated MIB file.
- Revision History: **mibgp** saves the location of the source file and the time it was processed to the initial lines of each generated MIB file.
- One to Many MIB Generation: If a single text document contains more than one MIB definition, mibgp processes each definition into separate output files.
- Many to Many MIB Generation: A command-line option allows **mibgp** to process all the text documents in a named location into the same location or specified location.
- Eliminates many common problems with MIB file generation, such as multiple newlines, headers and/or footers, **MACRO** definitions, and quoted strings.

Upgrades for Utilities

New Features and Enhancements

- Management Tools and Command Line Utilities have improved **EngineID** discovery. When performing **EngineID** discovery, we are now sending an empty VarBind list, and a populated list immediately thereafter, instead of filling in the VarBind list on the initial request(s).

- New type: -T for DateAndTime format.

The utilities **inform**, **setany**, and **trapsend** now support the format -T for DateAndTime. The date and time is specified in the following format, 2004-Jan-10,12:30:00.0.

- When retries was specified in the command line arguments to the Asynchronous Request Library/BRASS™ utilities, both the command line utility code and the underlying Asynchronous Request Library code was executing the retries, causing (retries * retries) retry attempts. This problem has been fixed so that there is only (retries) number of retry attempts.

Upgrades for Management Stations and Applications

SNMP Research provides management stations and applications for End-users and Original Equipment Manufacturers (OEMs). Management stations and applications enable administrators to monitor and control networks, systems, and applications. Options, such as Web and java-based interfaces and policy-based management, are also available. The following items reflect a brief overview of some of the more significant updates that affect SNMP Research's management stations and applications.

All user products have a new installation/uninstallation process that is much easier to use.

SNMP Security Pack

SNMP Security Pack adds support for SNMPv3, the standard for secure administration.

Service Monitor

Service Monitor provides a new extension to CIAgent® that monitors network services across multiple networks to allow administrators to ensure response times are within acceptable parameters.

Apache Server Monitor

Apache Server Monitor provides a new feature to CIAgent® that allows the user to set-up custom pages and custom settings for easy monitoring of Web servers, FTP servers, and much more.

EnterPol

- EnterPol Autodiscovery Engine
- The following improvements were made to **EnterPol**'s autodiscover process:
 - Greatly improved automatic seeding;
 - Several byte order fixes in various modules;
 - Fixes to autodiscover “(un)managed” subnet scoping;
 - Re-analysis of previously-discovered nodes when autodiscover is stopped then restarted.
 - Autodiscover parses and processes authentication and privacy keys containing prefixes which allow the user to specify the desired authentication and privacy protocols.
- A program called **displaydb**, included with **EnterPol**, was extended to add support for rendering the **EnterPol** event database.
- In previous releases, the agent can crash in the **Inform** re-send code if:
 1. The **snmpTargetParamsTable** indicates that the **Inform** should be sent as **authPriv**, and
 2. If the **usmUserTable** defines the user as **authNoPriv**.

This bug has been corrected. Instead, no **Inform** will be sent.

- Log files for EnterPol are now stored in a configurable directory specified in `enterpol.cnf`. The defaults are:
 - On UNIX: `/opt/Snmpri/EnterPol/logfiles`.
 - On Microsoft Windows: `C:\Program Files\Snmpri\EnterPol\logfiles`.
- The **EnterPol** and SNMP Security Pack product now include command-line utilities for performing simple SNMP **Get**, **GetNext**, **GetBulk**, and **Set** operations as well as for doing a MIB walk. The names of the utilities, respectively, are **getone**, **getnext**, **getbulk**, **setany**, and **getmany**.

SNMPv3 Configuration Wizard

The SNMPv3 Configuration Wizard is now included with SNMP Research Agent products for the developer's internal use only and is not intended for distribution to their end-user customers. A resale version may be licensed separately for distribution with an OEM customer's products.

The SNMPv3 Configuration Wizard now allows use of a non-default contextName. A customer may use a non-default contextName from the SNMPv3 Configuration Wizard when performing configuration sets to a device. SNMP Research products feature a new command-line option that

“turns on” the fields that allow the `contextName` to be entered using the wizard. By default, the context information feature is turned off.

CIAgent®

- New command line options have been added for starting CIAgent® processes on UNIX systems.
The `ciagent` command-line options `-fullstart` and `-fullstop` allow the user to start or stop all CIAgent® processes, including the Master Agent. The `ciagent -start` and `stop` command-line options allow the user to start or stop all CIAgent® subagent processes, such as the **Log File Monitor**, independently from the Master Agent.
- A problem was fixed in CIAgent® that caused startup scripts and documentation files to be copied to the wrong directories on HP/UX systems.
- The `ciinstall.log` file is created in the `TEMP` directory on Windows instead of in the `C:\Program Files\Snmpr` directory.
- The System Applications Subagent (**sappagt**) no longer starts by default when CIAgent® is started.
- The scripts that are used to stop the Master Agent and Subagents have been moved from `rc2.d` to `rc0.d` on UNIX systems.
- The Subagents that use a configuration file have been improved to back up the configuration file at startup, and also to use the `.jnk` file to retain any incorrect configuration entries. For example, **File System Monitor** (`fsagt`) uses the configuration file `fsagt.cnf`, the backup file `fsagt.cnf~`, and the junk file `fsagt.jnk`.
- User comments are retained in the configuration files. Previously, any comments the user added were stripped out of the file. Comments must begin with comment marks (`'##'`) at the beginning of the line in order to be retained. For example, the following entry is retained:

```
## This is a user comment  
## which will be written back out to the file.
```
- All Subagents write consistent error and warning output to a log file.
- Support has been added for the `-log_closefile` option which causes the logfile to be opened and then closed each time an log message is written. This allows the file to be accessible for backup without requiring the Subagent to be stopped. Previously, the log file would remain open through Subagent operation, which meant that the Subagent had to be stopped before the file could be moved.

- Erroneous configuration file entries that are placed in the .jnk file are preceded by the text “PARSING_ERROR:” at the beginning of the file. This allows the **Log File Monitor** to check this file for new errors.
- The configuration files for the **Logfile Monitor** (logagt), **File System Monitor** (fsagt), and **Critical Applications Monitor** (critagt) have the MIB variable names added so that it is easier to identify the objects in the configuration entries.

For More Information

For full descriptions of all of our products and their features, please visit our Web site at <http://www.snmp.com>.