Symantec™ Endpoint Protection Manager Database Schema Reference

For Symantec Endpoint Protection and Symantec Network Access Control

Symantec

Symantec™ Endpoint Protection Manager Database Schema Reference

Documentation version 12.01.00.00

Legal Notice

Symantec Corporation
350 Ellis Street
Mountain View, CA 94043

http://www.symantec.com

Technical Support

Symantec Technical Support maintains support centers globally. Technical Support's primary role is to respond to specific queries about product features and functionality. The Technical Support group also creates content for our online Knowledge Base. The Technical Support group works collaboratively with the other functional areas within Symantec to answer your questions in a timely fashion. For example, the Technical Support group works with Product Engineering and Symantec Security Response to provide alerting services and virus definition updates.

Symantec's support offerings include the following:

- A range of support options that give you the flexibility to select the right amount of service for any size organization
- Telephone and/or Web-based support that provides rapid response and up-to-the-minute information
- Upgrade assurance that delivers software upgrades
- Global support purchased on a regional business hours or 24 hours a day, 7 days a week basis
- Premium service offerings that include Account Management Services

For information about Symantec's support offerings, you can visit our Web site at the following URL:

www.symantec.com/business/support/

All support services will be delivered in accordance with your support agreement and the then-current enterprise technical support policy.

Contacting Technical Support

Customers with a current support agreement may access Technical Support information at the following URL:

www.symantec.com/business/support/

Before contacting Technical Support, make sure you have satisfied the system requirements that are listed in your product documentation. Also, you should be at the computer on which the problem occurred, in case it is necessary to replicate the problem.

When you contact Technical Support, please have the following information available:

- Product release level

- Hardware information
- Available memory, disk space, and NIC information
- Operating system
- Version and patch level
- Network topology
- Router, gateway, and IP address information
- Problem description:
  - Error messages and log files
  - Troubleshooting that was performed before contacting Symantec
  - Recent software configuration changes and network changes

Licensing and registration

If your Symantec product requires registration or a license key, access our technical support Web page at the following URL:

www.symantec.com/business/support/

Customer service

Customer service information is available at the following URL:

www.symantec.com/business/support/

Customer Service is available to assist with non-technical questions, such as the following types of issues:

- Questions regarding product licensing or serialization
- Product registration updates, such as address or name changes
- General product information (features, language availability, local dealers)
- Latest information about product updates and upgrades
- Information about upgrade assurance and support contracts
- Information about the Symantec Buying Programs
- Advice about Symantec's technical support options
- Nontechnical presales questions
- Issues that are related to CD-ROMs, DVDs, or manuals

Support agreement resources

If you want to contact Symantec regarding an existing support agreement, please contact the support agreement administration team for your region as follows:

Asia-Pacific and Japan	customercare_apac@symantec.com
Europe, Middle-East, and Africa	semea@symantec.com
North America and Latin America	supportsolutions@symantec.com

Additional enterprise services

Symantec offers a comprehensive set of services that allow you to maximize your investment in Symantec products and to develop your knowledge, expertise, and global insight, which enable you to manage your business risks proactively.

Enterprise services that are available include the following:

Managed Services	Managed Services remove the burden of managing and monitoring security devices and events, ensuring rapid response to real threats.
Consulting Services	Symantec Consulting Services provide on-site technical expertise from Symantec and its trusted partners. Symantec Consulting Services offer a variety of prepackaged and customizable options that include assessment, design, implementation, monitoring, and management capabilities. Each is focused on establishing and maintaining the integrity and availability of your IT resources.
Education Services	Education Services provide a full array of technical training, security education, security certification, and awareness communication programs.

To access more information about enterprise services, please visit our Web site at the following URL:

www.symantec.com/business/services/

Select your country or language from the site index.

Schema tables

This document includes the following topics:

About the Symantec Endpoint Protection Manager database schema

The Symantec Endpoint Protection Manager database stores all the information that concerns the Symantec software and associated security information. The information is stored in a series of tables, the database schema.

You can use the database schemas to create custom reports to find information about a large number of clients.

Data types represent the physical make up of the data.

The following data types are used in the database:

bigint	char
int	varchar
tinyint	nvarchar
datetime	varbinary

Some data types include the physical length of the field in parentheses. For example, char(24) indicates a character field with a length of 24 characters.

An asterisk (*) beside a field name indicates that the field acts as a Primary Key in the tables. The Primary Key is a column or a set of columns that uniquely identify all the rows in a table. Primary Keys may not contain null values. No two rows can have the same Primary Key value; therefore, a Primary Key value always uniquely identifies a single row. More than one key can uniquely identify rows in a table. Each of these keys is called a Candidate Key. Only one candidate can be chosen as the Primary Key of a table; all other Candidate Keys are known as Alternate Keys.

In a normalized table, all of a row's data values depend on the Primary Key. For example, in a normalized employee table with EmployeeID as the Primary Key, all columns contain data that is related to a specific employee. The table does not have a DepartmentName column, because the name of the department depends on a Department ID, not on an Employee ID.

In addition to the data tables, the Symantec Endpoint Protection Manager database contains views to enable you to look at the tables in different ways. A number of the views include human-readable IP address information.

See “About database views” on page 11.

About database views

The Symantec Endpoint Protection Manager database contains views to enable you to look at the data tables in different ways. The view names begin with the letter V to distinguish them from the tables. The following table lists these views and the purpose of each.

Views that are marked with an asterisk (*) provide human-readable IP address information. These views contain human-readable columns that are named

xxx_TEXT. The columns correspond to the non-human-readable field. For example, DNS_SERVER1_TEXT corresponds to the original DNS_SERVER1 non-human-readable field in the view V_SEM_COMPUTER.

Table 1-1 Purposes of database views

View	Purpose
V_AGENT_BEHAVIOR_LOG	Query client activities for clients.
V_AGENT_PACKET_LOG*	Query packet traffic events for clients.
V_AGENT_SECURITY_LOG*	Query security events for clients.
V_AGENT_SYSTEM_LOG	Query system events for clients.
V_AGENT_TRAFFIC_LOG*	Query traffic events for clients.
V_ALERTS*	Query risk and TruScan events with human-readable IP address information.
V_ENFORCER_CLIENT_LOG	Query client activities for Enforcers.
V_ENFORCER_SYSTEM_LOG	Query system activities for Enforcers.
V_ENFORCER_TRAFFIC_LOG*	Query traffic activities for Enforcers.
V_LAN_DEVICE_DETECTED*	Query detected devices with human-readable IP address information.
V_LAN_DEVICE_EXCLUDED*	Query known devices with human-readable IP address information.
V_SECURITY_VIEW	Query cross-technology security events.
V_SEM_COMPUTER*	Query computer information with human-readable IP address information.
V_SERVER_ADMIN_LOG	Query administrator activities for servers.
V_SERVER_CLIENT_LOG	Query client activities for servers.
V_SERVER_ENFORCER_LOG	Query Enforcer activities for servers.
V_SERVER_POLICY_LOG	Query policy change activities for servers.
V_SERVER_SYSTEM_LOG	Query system activities for servers.

Actual Action schema (ACTUALACTION table)

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_ACTUALACTION.

Table 1-2 Actual Action schema

Database Field Name	Comment	Data Type
ACTUALACTION_IDX*	Primary Key (one of 1...500 as shown)	int, not null

Table 1-2 Actual Action schema *(continued)*

Database Field Name	Comment	Data Type
ACTUALACTION		varchar(255), not null

Table 1-2 Actual Action schema *(continued)*

Database Field Name	Comment	Data Type
	A hard-coded English string that was used for the following lookups: -1 = Action invalid 1 = Quarantined 2 = Renamed 3 = Deleted 4 = Left alone 5 = Cleaned 6 = Cleaned or macros deleted 7 = Saved 9 = Moved back 10 = Renamed back 11 = Undone 12 = Bad 13 = Backed up 14 = Pending repair 15 = Partially repaired 16 = Process termination pending restart 17 = Excluded 18 = Restart processing 19 = Cleaned by deletion 20 = Access denied 21 = Process terminated 22 = No repair available 23 = All actions failed 98 = Suspicious 99 = Details pending 110 = Detected by using the commercial application list 111 = Forced detection by using the	

Table 1-2 Actual Action schema *(continued)*

Database Field Name	Comment	Data Type
	file name 1000 = Forced detection by using the file hash 500 = Not applicable	

Admin User schema (ADMINUSER table)

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_ADMINUSER.

Table 1-3 Admin User schema

Database Field Name	Comment	Data Type
USER_ID*	Primary Key, Logon user ID.	char(32), not null
USER_NAME	The user name of the admin.	nvarchar(255), varchar(255), not null
DOMAIN_ID	The GUID for the currently logged in domain.	char(32), not null
AUTOREFRESH	The user-defined auto refresh value for all logs (computer status, notifications, scan, and so on).	int, not null
LASTCHANGE	The last time that the user accessed the console.	int, not null
LASTSPMTIME	The last time of a successful keep alive to application server.	int, not null

Agent Behavior Logs schema (AGENT_BEHAVIOR_LOG_1 and AGENT_BEHAVIOR_LOG_2 tables)

The Agent Behavior Logs data table is not used in Symantec Network Access Control.

Table 1-4 describes the database schema for the client Behavior logs.

This schema has two tables. When logs are stored, the Symantec Endpoint Protection Manager uses the first table until it is full. The management server then uses the second table. The data in the first table is kept intact until the second table fills. Then the management server starts to fill the first table again. This cycle is continuous.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

The key is either I_AGENT_BEHAVIOR_LOG_1_LOG_IDX or I_AGENT_BEHAVIOR_LOG_2_LOG_IDX. The LOG_IDX field serves as the table's unique identifier, but it is not formally classified as the table's primary key. This field has an index on it, but it is not the primary key index. This table has no primary key.

Table 1-4 Agent Behavior Logs 1 and 2 schema

Database Field Name	Comment	Data Type
USN	A USN-based serial number; this ID is not unique.	bigint, not null
DOMAIN_ID	The GUID of the domain to which the log belongs.	char(32), not null
SITE_ID	The GUID of the site to which the log belongs.	char(32), not null
SERVER_ID	The GUID of the server to which the log belongs.	char(32), not null
GROUP_ID	The GUID of the group to which the log belongs.	char(32), not null
COMPUTER_ID	The GUID of the client computer to which the client log belongs.	char(32), not null

Table 1-4 Agent Behavior Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
EVENT_ID	An event ID from the Symantec Endpoint Protection client. Possible values are as follows: 501 = Application Control Driver 502 = Application Control Rules 999 = Tamper Protection	int, not null
EVENT_TIME	The event-generated time (in GMT).	bigint, not null
SEVERITY	The seriousness of the event. 0 is most serious.	int, not null
AGENT_ID	The GUID of the client.	char(32), null
HARDWARE_KEY	The hash of the computer hardware information.	char(32), null
HOST_NAME	The host name of client computer.	nvarchar(256), varchar(256), null
ACTION	Possible values include the following: 0 = allow 1 = block 2 = ask 3 = continue 4 = terminate	int, null
TEST_MODE	Was this rule run in test mode? 0 = No, Else = Yes	int, null
DESCRIPTION	The behavior that was blocked.	nvarchar(256), varchar(256), null
VAPI_NAME	The API that was blocked.	nvarchar(256), varchar(256), null
ENCODED_API_NAME		nvarchar(256), varchar(256), null
BEGIN_TIME	The start time of the security issue.	bigint, null

Table 1-4 Agent Behavior Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
END_TIME	The end time of the security issue. End time is an optional field because Symantec may fail to detect the exact end time of traffic, like UDP. In those cases, the end time is equal to start time.	bigint, null
RULE_ID	The ID of the rule that the event triggered. It is always 0 if the rule ID is not specified in the security rule. The field is helpful to security rule troubleshooting. If multiple rules match, RULE_ID logs the rule that has final decision on PacketProc (pass/block/drop).	char(32), null
RULE_NAME	The name of the rule that the event triggered. It is always an empty string if the rule name is not specified in the security rule. It is also used for troubleshooting. In theory, the IT admin can know the rule by ID. However, the name gives the user a direct view of the rule that can be used.	nvarchar(256), varchar(256), null
CALLER_PROCESS_ID	The ID of the process that triggers the logging.	bigint, null
CALLER_PROCESS_NAME	The full path name of the application that is involved. It may be empty if the application is unknown, if the operating system is involved, or if no application is involved. Also, it may be empty if the profile says “don’t log the application name in the raw traffic log.”	nvarchar(256), varchar(256), null
CALLER_RETURN_ADDRESS	The return address of the caller. This field allows the software to detect the calling module that makes the API call.	bigint, null

Table 1-4 Agent Behavior Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
CALLER_RETURN_MODULE_NAME	The module name of the caller. See the “CallerReturnAddress” field for more information.	nvarchar(256), varchar(256), null
PARAMETER	The parameters that were used in the API call. Each parameter was converted to STRING format and separated by one space character. Double quotation characters within the string are escaped by a backslash (\) character.	nvarchar(256), varchar(256), null
ALERT	ALERT indicates whether this event is counted during alert notification processing at the server. ALERT is true if Tamper Protection logs the event. It is false otherwise. Possible values are as follows: True = 1 False = 0	int, null
SEND_SNMP_TRAP	SEND_SNMP_TRAP reflects the send SNMP trap action. SEND_SNMP_TRAP is true if send is true.	tinyint, null
USER_NAME	The logon user name.	nvarchar(256), varchar(256), null
DOMAIN_NAME	The logon (Windows) domain name.	nvarchar(256), varchar(256), null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null

Table 1-4 Agent Behavior Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
REPETITION	Event repetition due to aggregation (damper).	int, not null
LOG_IDX*	The log index unique ID.	char(32), null
IP_ADDR	The IP address of the computer that is associated with the application control violation.	bigint, null
FILE_SIZE	The size of the file that is associated with the application control violation, in megabytes.	bigint, null
ACTION_TYPE	The violation type that triggered the SymProtect event.	smallint, null
PARAM_DEVICE_ID	The GUID that corresponds to a device, such as floppy or DVD.	varchar(256), null

Agent Packet Logs schema (AGENT_PACKET_LOG_1 and AGENT_PACKET_LOG_2 tables)

The Agent Packet Logs data table is not used in Symantec Network Access Control.

Table 1-5 describes the database schema for the client Packet logs.

This schema has two tables. When logs are stored, the Symantec Endpoint Protection Manager uses the first table until it is full. The management server then uses the second table. The data in the first table is kept intact until the second table fills. Then the management server starts to fill the first table again. This cycle is continuous.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

The key is either I_AGENT_PACKET_LOG_1_LOG_IDX or I_AGENT_PACKET_LOG_2_LOG_IDX. The LOG_IDX field serves as the table's unique identifier, but it is not formally classified as the table's primary key. This field has an index on it, but it is not the primary key index. This table has no primary key.

Table 1-5 Agent Packet Logs 1 and 2 schema

Database Field Name	Comment	Data Type
USN	A USN-based serial number; this ID is not unique.	bigint, not null
DOMAIN_ID	The GUID of the domain to which the log belongs.	char(32), not null
SITE_ID	The GUID of the site to which the log belongs.	char(32), not null
SERVER_ID	The GUID of the server to which the log belongs.	char(32), not null
GROUP_ID	The GUID of the group to which the log belongs.	char(32), not null
COMPUTER_ID	The GUID of the client computer to which the client packet log belongs.	char(32), not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
EVENT_ID	An event ID from the Symantec Endpoint Protection client. 401 = Raw Ethernet	int, not null
EVENT_TIME	The event-generated time (in GMT).	bigint, not null
AGENT_ID	The GUID of the client.	char(32), null
HARDWARE_KEY	The hash of the computer hardware information.	char(32), null
HOST_NAME	The host name of the client computer to which the client belongs.	nvarchar(256), varchar(256), null
LOCAL_HOST_IP	The IP address of the local computer (IPv4).	bigint, null
REMOTE_HOST_IP	The IP address of the remote computer (IPv4).	bigint, null
REMOTE_HOST_NAME	The name of the remote computer. It may be empty if the name resolution failed.	nvarchar(64), varchar(64), null

Table 1-5 Agent Packet Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
LOCAL_PORT	The TCP/UDP port in local computer (host byte-order). It is valid only on TSE_TRAFFIC_TCP and TSE_TRAFFIC_UDP. Otherwise, it is always zero.	int, null
REMOTE_PORT	The TCP/UDP port in remote computer (host byte-order). It is only valid on TSE_TRAFFIC_TCP and TSE_TRAFFIC_UDP. Otherwise, it is always zero.	int, null
TRAFFIC_DIRECTION	The direction of traffic. Enum (unknown = 0; inbound = 1; outbound = 2)	tinyint, null
BLOCKED	Whether the traffic was blocked. Possible values are as follows: Yes = 1 No = 0	tinyint, not null
APP_NAME	The full path name of the application involved. It may be empty if an unknown application is involved or if no application is involved. For example, the ping of death denial-of-service attack doesn't have an AppName because it attacks the operating system.	nvarchar(256), varchar(256), null
ALERT	ALERT reflects the alert attribute in the profile action. If the Network Threat Protection policy indicates that the event should be considered for server-side notification generation, the ALERT field is set to 1. Possible values are as follows: Yes = 1 No = 0	int, null

Table 1-5 Agent Packet Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
SEND_SNMP_TRAP	SEND_SNMP_TRAP reflects the send SNMP trap action. SEND_SNMP_TRAP is true if send is true. Possible values are as follows: Yes = 1 No = 0	tinyint, null
EVENT_DATA	Additional data in binary format. This field is optional.	varbinary(2000), null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null
LOG_IDX*	The log index unique ID.	char(32), null
LOCAL_HOST_IPV6	The local host IPv6.	varchar(32), null
REMOTE_HOST_IPV6	The remote host IPv6.	varchar(32), null
RULE_NAME	The rule name.	nvarchar(512), null

Agent Security Logs schema (AGENT_SECURITY_LOG_1 and AGENT_SECURITY_LOG_2 tables)

Table 1-6 describes the database schema for the client Security logs.

This schema has two tables. When logs are stored, the Symantec Endpoint Protection Manager uses the first table until it is full. The management server then uses the second table. The data in the first table is kept intact until the second

table fills. Then the management server starts to fill the first table again. This cycle is continuous.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

The key is either I_AGENT_SECURITY_LOG_1_AGENT_SECURITY_LOG_IDX or I_AGENT_SECURITY_LOG_2_AGENT_SECURITY_LOG_IDX. The AGENT_SECURITY_LOG_IDX field serves as the table's unique identifier, but it is not formally classified as the table's primary key. This field has an index on it, but it is not the primary key index. This table has no primary key.

Table 1-6 Agent Security Logs 1 and 2 schema

Database Field Name	Comment	Data Type
USN	A USN-based serial number; this ID is not unique.	bigint, not null
DOMAIN_ID	The GUID of the domain to which the log belongs.	char(32), not null
SITE_ID	The GUID of the site to which the log belongs.	char(32), not null
SERVER_ID	The GUID of the server to which the log belongs.	char(32), not null
GROUP_ID	The GUID of the group to which the log belongs.	char(32), not null
COMPUTER_ID	The GUID of the client computer that is associated with the client security log.	char(32), not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null

Table 1-6 Agent Security Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
EVENT_ID	Compliance events: 209 = Host Integrity failed (TSLOG_SEC_NO_AV) 210 = Host Integrity passed (TSLOG_SEC_AV) 221 = Host Integrity failed but it was reported as PASS 237 = Host Integrity custom log entry Firewall and IPS events: 207 = Active Response 211 = Active Response Disengaged 219 = Active Response Canceled 205 = Executable file changed 216 = Executable file change detected 217 = Executable file change accepted 218 = Executable file change denied 220 = Application Hijacking 201 = Invalid traffic by rule 202 = Port Scan 203 = Denial-of-service attack 204 = Trojan horse 206 = Intrusion Prevention System (Intrusion Detected, TSLOG_SEC_INTRUSION_ DETECTED) 208 = MAC Spoofing Application and Device control: 238 = Device control disabled device 239 = Buffer Overflow Event 240 = Software protection has thrown an exception	int, not null
EVENT_TIME	The event-generated time (in GMT).	bigint, not null

Table 1-6 Agent Security Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
SEVERITY	The level of severity that is defined in Security Rule. Possible values are as follows: Critical = 0 - 3 Major = 4 - 7 Minor = 8 - 11 Info = 12 - 15	int, not null
AGENT_ID	The GUID of the client.	char(32), null
HARDWARE_KEY	The hash of the computer hardware information.	char(32), null
HOST_NAME	The host name of the client computer.	nvarchar(256), varchar(256), null
LOCAL_HOST_IP	The IP address of local computer (IPv4).	bigint, null
REMOTE_HOST_IP	The IP address of remote computer (IPv4).	bigint, null
REMOTE_HOST_NAME	The name of remote computer. It may be empty if the name resolution failed.	nvarchar(64), varchar(64), null
TRAFFIC_DIRECTION	The direction of traffic. Enum (unknown = 0; inbound = 1; outbound = 2)	tinyint, null
NETWORK_PROTOCOL	The protocol type: Enum (OTHERS = 1; TCP = 2; UDP = 3; ICMP = 4)	tinyint, null

Table 1-6 Agent Security Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
HACK_TYPE	It is a reason if the Event ID is TSLOG_SEC_NO_AV. It is the intrusion ID if the Event ID is TSLOG_SEC_INTRUSION_ DETECTED. It is additional information if event ID is TSLOG_SEC_AV. Possible reasons are as follows: Process is not running - Bit 0 is 1 Signature is out of date - Bit 1 is 1 Recovery was tried - Bit 2 is 1	int, null
BEGIN_TIME	The start time of the security issue.	bigint, null
END_TIME	The end time of the security issue. End time is an optional field because the software may fail to detect the exact end time of traffic, like UDP. In those cases, the end time is equal to the begin time.	bigint, null
REPETITION	The number of attacks. When a hacker launches a mass attack, it may be damped to one event by the log system.	int, null
APP_NAME	The full path of the application involved. It may be empty if an unknown application is involved or if no application is involved. For example, the ping of death denial-of-service attack doesn't have an AppName because it attacks the operating system itself.	nvarchar(256), varchar(256), null
EVENT_DESC	A description of the event. Usually, the first line of the description is treated as "summary".	nvarchar(2000), varchar(4000), null
EVENT_DATA	Additional data in binary format. This field is optional.	varbinary(3000), null

Table 1-6 Agent Security Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
ALERT	ALERT reflects the alert attribute in profile action. If the Network Threat Protection policy indicates that the event should be considered for server-side notification generation, the ALERT field is set to 1. Possible values are as follows: Yes = 1 No = 0	tinyint, null
SEND_SNMP_TRAP	SEND_SNMP_TRAP reflects the send SNMP trap action. SEND_SNMP_TRAP is true if send is true. Possible values are as follows: Yes = 1 No = 0	tinyint, null
LOCAL_HOST_MAC	The MAC address of the local computer.	varchar(18), null
REMOTE_HOST_MAC	The MAC address of the remote computer.	varchar(18), null
LOCATION_NAME	The location that is used when the event occurs.	nvarchar(256), varchar(256), null
USER_NAME	The logon user name.	nvarchar(256), varchar(256), null
DOMAIN_NAME	The logon domain name.	nvarchar(256), varchar(256), null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null

Table 1-6 Agent Security Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
RESERVED_BINARY		varbinary(1900), null
AGENT_SECURITY_LOG_IDX*	The log index unique ID.	char(32), null
LOCAL_HOST_IPV6	The local host IPv6.	varchar(32), null
REMOTE_HOST_IPV6	The remote host IPv6.	varchar(32), null
LOCAL_PORT	The local port.	int, not null
REMOTE_PORT	The remote port.	int, not null
CIDS_SIGN_ID	The signature ID.	bigint, not null
STR_CIDS_SIGN_ID	The signature name.	int, not null
CIDS_SIGN_SUB_ID	The signature sub ID.	bigint, not null
INTRUSION_URL	The URL from where a malicious script was loaded. Internet Browser Protection uses this URL.	nvarchar(4200), not null
INTRUSION_PAYLOAD_URL	The redirection URL that is used to download other malicious scripts. Internet Browser Protection uses this URL.	nvarchar(4200), not null
HI_EXECUTION_ID	The ID that the Network Access Control client generates for each Host Integrity execution.	varchar(50), null
AGENT_VERSION	The version number of the client.	nvarchar(64), null
PROFILE_SERIAL_NO	The policy serial number.	varchar(64), null

Agent Status schema (AGENTSTATUS table)

Table 1-7 describes the database schema for client status information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_AGENTSTATUS.

Table 1-7 Agent Status schema

Database Field Name	Comment	Data Type
IDX*	Primary Key.	char(32), not null
AGENTTYPE	Possible values for AGENTTYPE include the following: SAV 10.x LogSender ClientInventory SAV 11.x AgentSweepingTask (Database maintenance) TopThreatsTask (Gathers top and latest threats information) VirusCatTask (Gathers virus properties) ThreatCatTask (Gathers risk properties)	varchar(255), not null
AGENTNAME	Name that is associated with this client. for LogSender clients: Server Group name for LogSenderSAVSMTP clients: mail gateway host name for ClientInventory clients: name of Parent Server else: blank	varchar(255), not null
LASTRUNGMT	Last time this client ran stored in GMT.	varchar(50), not null
REMOTE_TZ_OFFSET	The time zone offset.	int, not null
REPORTER_TZ_OFFSET	The time zone offset.	int, not null

Table 1-7 Agent Status schema *(continued)*

Database Field Name	Comment	Data Type
MAIL	Flag whether email has already been sent. Possible values are as follows: 1 = Yes 0 = No	int, not null
VERSION_BUILD	The version/build (major or minor build) of the client.	varchar(20), not null
MACHINE_NAME	The computer name of the client computer.	nvarchar(128), varchar(128), not null
SERVERGROUP_IDX	Pointer to IDENTITY_MAP table.	char(32), not null
LASTRUN_DATA	Extra data that is associated with the client run, if any.	nvarchar(255), varchar(255), null

Agent System Logs schema (AGENT_SYSTEM_LOG_1 and AGENT_SYSTEM_LOG_2 tables)

Table 1-8 describes the database schema for the client System logs.

This schema has two tables. When logs are stored, the Symantec Endpoint Protection Manager uses the first table until it is full. The management server then uses the second table. The data in the first table is kept intact until the second table fills. Then the management server starts to fill the first table again. This cycle is continuous.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

The key is either I_AGENT_SYSTEM_LOG_1_LOG_IDX or I_AGENT_SYSTEM_LOG_2_LOG_IDX. The LOG_IDX field serves as the table's unique identifier, but it is not formally classified as the table's primary key. This field has an index on it, but it is not the primary key index. This table has no primary key.

Table 1-8 Agent System Logs 1 and 2 schema

Database Field Name	Comment	Data Type
USN	A USN-based serial number; this ID is not unique.	bigint, not null
DOMAIN_ID	The GUID of the domain to which the log belongs.	char(32), not null
SITE_ID	The GUID of the site to which the log belongs.	char(32), not null
SERVER_ID	The GUID of the server to which the log belongs.	char(32), not null
GROUP_ID	The GUID of the group to which the log belongs.	char(32), not null
COMPUTER_ID	The GUID of the client computer that is associated with the client system log.	char(32), not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null

Table 1-8 Agent System Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
EVENT_ID		int, not null

Table 1-8 Agent System Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
	An event ID from the Symantec Endpoint Protection client. AGENT_SYSTEM_INSTALL_EVENT_TYPES = Installation events: Possible values include the following: 0x12070001 = Internal error 0x12070101 = Install complete 0x12070102 = Restart recommended 0x12070103 = Restart required 0x12070104 = Installation failed 0x12070105 = Uninstallation complete 0x12070106 = Uninstallation failed 0x12071037 = Symantec AntiVirus installed 0x12071038 = Symantec Firewall installed 0x12071039 = Uninstall 0x1207103A = Uninstall rolled-back AGENT_SYSTEM_SERVICE_EVENT_TYPES = Service events: Possible values include the following: 0x12070201 = Service starting 0x12070202 = Service started 0x12070203 = Service start failure 0x12070204 = Service stopped 0x12070205=Service stop failure 0x1207021A = Attempt to stop service AGENT_SYSTEM_CONFIG_EVENT_TYPES = Configuration events: Possible values include the following: 0x12070206 = Config import complete 0x12070207 = Config import error 0x12070208 = Config export complete 0x12070209 = Config export error AGENT_SYSTEM_HI_EVENT_TYPES = Host Integrity events:	

Table 1-8 Agent System Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
	Possible values include the following: 0x12070210 = Host Integrity disabled 0x12070211 = Host Integrity enabled 0x12070220 = NAP integration enabled AGENT_SYSTEM_IMPORT_EVENT_TYPES = Import events: Possible values include the following: 0x12070214 = Successfully imported advanced rule 0x12070215 = Failed to import advanced rule 0x12070216 = Successfully exported advanced rule 0x12070217 = Failed to export advanced rule AGENT_SYSTEM_CLIENT_EVENT_TYPES = Client events: Possible values include the following: 0x12070218 = Client Engine enabled 0x12070219 = Client Engine disabled 0x12071046 = Proactive Threat Scanning is not supported on this platform 0x12071047 = Proactive Threat Scanning Load Error AGENT_SYSTEM_SERVER_EVENT_TYPES = Server events: Possible values include the following: 0x12070301 = Server connected 0x12070302 = No server response 0x12070303 = Server connection failed 0x12070304 = Server disconnected 0x120B0001 = Cannot reach server 0x120B0002 = Reconnected server AGENT_SYSTEM_PROFILE_EVENT_TYPES = Policy events: Possible values include the following: 0x12070306 = New policy received 0x12070307 = New policy applied 0x12070308 = New policy failed	

Table 1-8 Agent System Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
	0x12070309 = Cannot download policy 0x120B0005 = Cannot download policy 0x1207030A = Have latest policy 0x120B0004 = Have latest policy AGENT_SYSTEM_AV_EVENT_TYPES = Antivirus engine events: Possible values include the following: 0x12071006 = Scan Omission 0x1207100B = Virus Behavior Detected 0x1207100C = Configuration Changed 0x12071010 = Definition File Download 0x12071012 = Sent To Quarantine Server 0x12071013 = Delivered To Symantec 0x12071014 = Security Response Backup 0x12071015 = Scan Aborted 0x12071016 = Symantec AntiVirus Auto-Protect Load Error 0x12071017 = Symantec AntiVirus Auto-Protect Enabled 0x12071018 = Symantec AntiVirus Auto-Protect Disabled 0x1207101A = Scan Delayed 0x1207101B = Scan Restarted 0x12071027 = Symantec AntiVirus is using old virus definitions 0x12071041 = Scan Suspended 0x12071042 = Scan Resumed 0x12071043 = Scan Duration Too Short 0x12071045 = Scan Enhancements Failed AGENT_SYSTEM_LICENSE_EVENT_TYPES = License events: Possible values include the following: 0x1207101E = License Warning 0x1207101F = License Error 0x12071020 = License in Grace Period 0x12071023 = License Installed	

Table 1-8 Agent System Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
	0x12071025 = License Up-to-date AGENT_SYSTEM_SECURITY_EVENT_TYPES = Security events: Possible values include the following: 0x1207102B = Computer not compliant with security policy 0x1207102C = Computer compliant with security policy 0x1207102D = Tamper Attempt AGENT_SYSTEM_OTHER_EVENT_TYPES = Other events: Possible values include the following: 0x1207020A = email post OK 0x1207020B = email post failure 0x1207020C = Update complete 0x1207020D = Update failure 0x1207020E = Manual location change 0x1207020F = Location changed 0x12070212 = Old Rasdll detected 0x12070213 = Autoupdate postponed 0x12070305 = Mode changed 0x1207030B = Cannot apply HI script 0x12070500 = System message from device control 0x12070600 = System message from anti-buffer overflow driver 0x12071021 = Access Denied Warning 0x12071022 = Log Forwarding Error 0x12071044 = Client moved	
EVENT_TIME	The event-generated time (in GMT).	bigint, not null

Table 1-8 Agent System Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
SEVERITY	The type of event. Possible values are as follows: INFO = 0 WARNING = 1 ERROR = 2 FATAL = 3	int, not null
AGENT_ID	The GUID of the client.	char(32), null
HARDWARE_KEY	The hash of the computer hardware information.	char(32), null
HOST_NAME	The host name of client computer.	nvarchar(256), varchar(256), null
CATEGORY	CATEGORY is not used now.	int, null
EVENT_SOURCE	The data source, such as NETPORT, NATSRV, etc.	varchar(32), not null
EVENT_DESC	A description of the event. Usually, the first line of the description is treated as “summary.”	nvarchar(1024), varchar(2048), null
EVENT_DATA	Additional data in binary format. This field is optional.	varbinary(2000), null
SEND_SNMP_TRAP	SEND_SNMP_TRAP reflects the send SNMP trap action. SEND_SNMP_TRAP is true if send is true. Possible values are as follows: Yes = 1 No = 0	tinyint, null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null

Table 1-8 Agent System Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
RESERVED_BINARY		varbinary(2000), null
LOG_IDX*	The log index unique ID.	char(32), null

Agent Traffic Logs schema (AGENT_TRAFFIC_LOG_1 and AGENT_TRAFFIC_LOG_2 tables)

Table 1-9 describes the database schema for the client Traffic logs.

This schema has two tables. When logs are stored, the Symantec Endpoint Protection Manager uses the first table until it is full. The management server then uses the second table. The data in the first table is kept intact until the second table fills. Then the management server starts to fill the first table again. This cycle is continuous.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

The key is either I_AGENT_TRAFFIC_LOG_1_LOG_IDX or I_AGENT_TRAFFIC_LOG_2_LOG_IDX. The LOG_IDX field serves as the table's unique identifier, but it is not formally classified as the table's primary key. This field has an index on it, but it is not the primary key index. This table has no primary key.

Table 1-9 Agent Traffic Logs 1 and 2 schema

Database Field Name	Comment	Data Type
USN	A USN-based serial number; this ID is not unique.	bigint, not null
DOMAIN_ID	The GUID of the domain to which the log belongs.	char(32), not null
SITE_ID	The GUID of the site to which the log belongs.	char(32), not null
SERVER_ID	The GUID of the server to which the log belongs.	char(32), not null

Table 1-9 Agent Traffic Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
GROUP_ID	The GUID of the group to which the log belongs.	char(32), not null
COMPUTER_ID	The GUID of the client computer that is associated with the client traffic log.	char(32), not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
EVENT_ID	An event ID from Symantec Endpoint Protection client. Possible values are as follows: 301 = TCP initiated 302 = UDP datagram 303 = Ping request 304 = TCP completed 305 = Traffic (other) 306 = ICMP packet 307 = Ethernet packet 308 = IP packet	int, not null
EVENT_TIME	The event-generated time (in GMT).	bigint, not null
SEVERITY	Severity as defined in the Security Rule. Possible values are as follows: Critical = 0 - 3 Major = 4 - 7 Minor = 8 - 11 Info = 12 - 15	int, not null
AGENT_ID	The GUID of the client.	char(32), null
HARDWARE_KEY	The hash of the computer hardware information.	char(32), null
HOST_NAME	The host name of the client computer.	nvarchar(256), varchar(256), null

Table 1-9 Agent Traffic Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
LOCAL_HOST_IP	The IP address of the local computer (IPv4).	bigint, null
REMOTE_HOST_IP	The IP address of the remote computer (IPv4).	bigint, null
REMOTE_HOST_NAME	The name of the remote computer. It may be empty if the name resolution failed.	nvarchar(64), varchar(64), null
NETWORK_PROTOCOL	The protocol type: Enum (OTHERS = 1; TCP = 2; UDP = 3; ICMP = 4).	tinyint, null
LOCAL_PORT	The TCP/UDP port in the local computer (host byte-order). It is only valid on TSE_TRAFFIC_TCP and TSE_TRAFFIC_UDP. Otherwise, it is always zero.	int, null
REMOTE_PORT	The TCP/UDP port in the remote computer (host byte-order). It is only valid on TSE_TRAFFIC_TCP and TSE_TRAFFIC_UDP. Otherwise, it is always zero.	int, null
TRAFFIC_DIRECTION	The direction of traffic. Enum (unknown = 0; inbound = 1; outbound = 2)	tinyint, null
BEGIN_TIME	The start time of the security issue.	bigint, null
END_TIME	The end time of the security issue. End time is an optional field because we may fail to detect the exact end time of traffic, like UDP. In those cases, the end time is equal to begin time.	bigint, null
REPETITION	The number of attacks. Sometimes, when a hacker launches a mass attack, it may be damped to one event by the log system.	int, null

Table 1-9 Agent Traffic Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
APP_NAME	The full path of application involved. It may be empty if an unknown application is involved or if no application is involved. For example, the ping of death denial-of-service attack doesn't have AppName because it attacks the operating system itself.	nvarchar(256), varchar(256) , null
BLOCKED	Specify if the traffic was blocked. Possible values are as follows: Yes = 1 No = 0	tinyint, not null
RULE_ID	The ID of rule that the event triggered. It is always 0 if rule ID is not specified in security rule. The field is helpful to security rule troubleshooting. If multiple rules matched, it logs the rule that has final decision on PacketProc (pass/block/drop).	char(32), null
RULE_NAME	The name of rule that the event triggered. It is always an empty string if a rule name is not specified in the security rule. It is also used for troubleshooting. In theory, an IT admin can know the rule by its ID. However, a name gives the user a direct view of a rule that can be used.	nvarchar(256), varchar(256), null
ALERT	ALERT reflects the alert attribute in the profile action. If the Network Threat Protection policy indicates that the event should be considered for server-side notification generation, the ALERT field is set to 1. Possible values are as follows: Yes = 1 No = 0	tinyint, null

Table 1-9 Agent Traffic Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
SEND_SNMP_TRAP	It reflects the send SNMP trap action. SEND_SNMP_TRAP is true if send is true. Possible values are as follows: Yes = 1 No = 0	tinyint, null
LOCAL_HOST_MAC	The MAC address of local computer.	varchar(18), null
REMOTE_HOST_MAC	The MAC address of remote computer.	varchar(18), null
LOCATION_NAME	The location that was used when event occurs.	nvarchar(256), varchar(256), null
USER_NAME	The logon user name.	nvarchar(256), varchar(256), null
DOMAIN_NAME	The logon domain name.	nvarchar(256), varchar(256), null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null
LOG_IDX*	The log index unique ID.	char(32), null
LOCAL_HOST_IPV6	The local host IPv6.	varchar(32), null
REMOTE_HOST_IPV6	The remote host IPv6.	varchar(32), null

Alert Filter schema (ALERTFILTER table)

Table 1-10 describes the database schema for alert filter information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_ALERTFILTER.

Table 1-10 Alert Filter schema

Database Field Name	Comment	Data Type
ALERTFILTER_IDX*	Primary Key.	char(32), not null
USER_ID	The user ID.	char(32), not null
FILTERNAME	The user-specified name of the filter.	nvarchar(510), not null
STARTDATEFROM	The start date.	datetime, not null
STARTDATETO	The end date.	datetime, not null
RELATIVEDATETYPE	Possible values are as follows: 0 = past week 1 = past month 2 = past three months 3 = past year 4 = past 24 hours 5 = current month	int, not null
FILTERACKNOWLEDGED	Possible values are as follows: 1 = Acknowledged 0 = Unacknowledged	nvarchar(255), varchar(255), not null

Table 1-10 Alert Filter schema *(continued)*

Database Field Name	Comment	Data Type
FILTERSUBJECT	Possible values are as follows: AF = Authentication failure CL = Client list changed CS = Client security alert ED = Enforcer Down WL = Forced or commercial application detected LA = New learned application NV = New risk detected NS = New software package VO = Virus outbreak DF = Server health 1V = Single risk event SE = System event UM = Unmanaged computer ID = Virus definitions out-of-date	nvarchar(255), varchar(255), not null
FILTERCREATEDBY	The GUID of the administrator who created this alert filter.	nvarchar(255), varchar(255), not null
LASTCOLUMN	Not used.	varchar(255), not null
SERVERGROUP	Not used.	nvarchar(255), varchar(255), not null
CLIENTGROUP	Not used.	nvarchar(255), varchar(255), not null
PARENTSERVER	Not used.	nvarchar(255), varchar(255), not null
COMPUTER	Not used.	nvarchar(255), varchar(255), not null
THREATNAME	Not used.	nvarchar(255), varchar(255), not null
THREATCATEGORY	Not used.	varchar(255), not null
SOURCE	Not used.	varchar(255), not null
ACTUALACTION	Not used.	varchar(255), not null

Table 1-10 Alert Filter schema *(continued)*

Database Field Name	Comment	Data Type
LIMITROWS	The number of rows to use for pagination.	int, not null
USERELATIVE	Use relative dates ('on') or absolute dates.	char(2), not null
REPORTINPUTS	Special parameters if a report needs them.	nvarchar(64), varchar(64), not null
NOTIFICATIONNAME	The name of selected notification condition.	nvarchar(255), varchar(255), not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row: 0 = not deleted 1 = deleted	tinyint, not null

Alert Message schema (ALERTMSG table)

Table 1-11 describes the database schema for alert message information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_ALERTMSG.

Table 1-11 Alert Message schema

Database Field Name	Comment	Data Type
ALERT_IDX*	Primary Key (one of 1 through 9).	int, not null

Table 1-11 Alert Message schema *(continued)*

Database Field Name	Comment	Data Type
ALERT	ALERT is a hard-coded English string that is used as a lookup It corresponds to an event ID from the Symantec Endpoint Protection client. Possible values are as follows: 1 = Virus found 2 = Security risk found 3 is not used 4 is not used 5 = Commercial application detected 6 = Forced proactive threat detected 7 = Proactive detection now permitted 8 = Potential risk found 9 = Risk sample was submitted to Symantec	varchar(128), not null

Alerts schema (ALERTS table)

Table 1-12 describes the database schema for alerts information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_ALERTS.

Table 1-12 Alerts schema

Database Field Name	Comment	Data Type
IDX*	Primary Key.	char(32), not null
ALERT_IDX	Pointer to table ALERTMSG.	int, not null
COMPUTER_IDX	Foreign key to SEM_COMPUTER.COMPUTER_ID.	char(32), not null

Table 1-12 Alerts schema *(continued)*

Database Field Name	Comment	Data Type
SOURCE	A hard-coded English string that is used as a lookup key for the following scan types: "Scheduled Scan" "Manual Scan" "Real Time Scan" "Integrity Shield" "Definition downloader" "System" "Startup Scan" "DefWatch" "Manual Quarantine" "Reboot Processing" "Heuristic Scan"	varchar(50), not null
VIRUSNAME_IDX	Pointer to table VIRUS.	char(32), not null
NOOFVIRUSES	The number of events for the aggregated event record. This number can be due to client-side aggregation, server-side compression, or both.	int, not null
FILEPATH	The file path of attacked file.	nvarchar(255), varchar(255), not null
DESCRIPTION	A description of the event.	nvarchar(255), varchar(255), not null
ACTUALACTION_IDX	Pointer to table ACTUALACTION, this is the action taken on the risk.	int, not null
REQUESTEDACTION_IDX	Pointer to table ACTUALACTION; this is the action requested by the policy.	int, not null
SECONDARYACTION_IDX	Pointer to table ACTUALACTION; this is the secondary action requested by the policy.	int, not null
ALERTDATETIME	The time of event occurrences.	datetime, not null
ALERTINSERTTIME	The time at which the event was inserted in to the database.	datetime, not null

Table 1-12 Alerts schema *(continued)*

Database Field Name	Comment	Data Type
SERVERGROUP_IDX	Pointer to table IDENTITY_MAP; this is the Symantec Endpoint Protection Manager domain GUID.	char(32), not null
USER_NAME	The name of the user that was logged onto the computer when the event took place.	nvarchar(64), varchar(64), not null
PARENTSERVER_IDX	Pointer to table IDENTITY_MAP; this is the Symantec Endpoint Protection Manager server GUID.	char(32), not null
CLIENTGROUP_IDX	Pointer to table IDENTITY_MAP; this is the Symantec Endpoint Protection Manager group GUID.	char(32), not null
SOURCE_COMPUTER_NAME	The source of the threat. It is logged when threat tracer is enabled in the Virus and Spyware policy.	nvarchar(64), varchar(64), not null
SOURCE_COMPUTER_IP	The source of the threat. It is logged when threat tracer is enabled in the antivirus and antispyware policy.	bigint, not null
MOTHER_IDX	Pointer to the related compressed event in the ALERTS table. This is the compressed event created by database maintenance. A value here means that this event has been aggregated server-side and is a child event.	char(32), not null
LAST_LOG_SESSION_GUID	An ID that is used by the client to keep track of related threat events.	char(32), not null
ALERTENDDATETIME	The time at which the event ended. This is the end of the aggregated event time.	datetime, not null
HPP_APP_IDX	Pointer to HPP_APPLICATION table.	varchar(32), not null
SITE_IDX	Pointer to table IDENTITY_MAP; this is the Symantec Endpoint Protection Manager site GUID.	char(32), null
VBIN_ID	The client-side ID of the quarantined threat, if quarantined.	bigint, not null

Table 1-12 Alerts schema *(continued)*

Database Field Name	Comment	Data Type
SCAN_ID	Pointer to the scan table event that picked up this event.	bigint, not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row: 0 = not deleted 1 = deleted	tinyint, not null
LOCAL_HOST_IP	The IP address of the client when the detection was made, in the format "0.0.0.0"	bigint, null
AV_PRODUCT	The name of the antivirus product.	varchar(256), null
AV_PRODUCT_VERSION	The version number of the antivirus product.	varchar(64), null

Anomaly Detection schema (ANOMALYDETECTION table)

Table 1-13 describes the database schema for anomaly detection information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_ANOMALYDETECTION.

Table 1-13 Anomaly Detection schema

Database Field Name	Comment	Data Type
ANOMALY_DETECTION_IDX*	Primary Key.	char(32), not null

Table 1-13 Anomaly Detection schema *(continued)*

Database Field Name	Comment	Data Type
ANOMALY_DETECTION_OPERATION_ID	Pointer to table 'Anomalydetectionoperation'.	int, not null
ANOMALY_DETECTION_TYPE_ID	Pointer to table 'Anomalydetectiontype'.	int, not null
ACTION_OPERAND	The file or the registry key on which this action took place.	nvarchar(512), varchar(512), not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row: 0 = not deleted 1 = deleted	tinyint, not null
ACTION_OPERAND_HASH	The hash value for the ACTION_OPERAND column.	char(32), null

Anomaly Detection Operation schema (ANOMALYDETECTIONOPERATION table)

Table 1-14 describes the database schema for anomaly detection operation information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_ANOMALYDETECTIONOPERATION.

Table 1-14 Anomaly Detection Operation schema

Database Field Name	Comment	Data Type
DETECTION_OPERATION_ID*	0-8	int, not null

Table 1-14 Anomaly Detection Operation schema *(continued)*

Database Field Name	Comment	Data Type
DETECTION_OPERATION_DESC	Detection_Operation_ID, Detection_Operation_Desc. A hard-coded English string that is used for a lookup Possible values are as follows: 0 = Unknown 1 = Scan 2 = Present 3 = Not Present 4 = Equal 5 = Not Equal 6 = Equal (Case-insensitive) 7 = Not Equal (Case-insensitive) 8 = Scan Memory	varchar(255), not null

Anomaly Detection Type schema (ANOMALYDETECTIONTYPE table)

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_ANOMALYDETECTIONTYPE.

Table 1-15 Anomaly Detection Type schema

Database Field Name	Comment	Data Type
DETECTION_TYPE_ID*	Primary Key.	int, not null

Table 1-15 Anomaly Detection Type schema *(continued)*

Database Field Name	Comment	Data Type
DETECTION_TYPE_DESC	Detection_Type_ID, Detection_Type_Desc. A hard-coded English string that is used for a lookup Possible values are as follows: 1000 = Registry 1001 = File 1002 = Process 1003 = Batch File 1004 = INI File 1005 = Service 1006 = Infected File 1007 = COM Object 1008 = Hosts File Entry 1009 = Directory 1010 = Layered Service Provider	varchar(255), not null

Anomaly Detections schema (ANOMALYDETECTIONS table)

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as the Primary Key, PK_ANOMALYDETECTIONS.

Table 1-16 Anomaly Detections schema

Database Field Name	Comment	Data Type
ALERT_EVENT_IDX	Foreign key to ALERTS.IDX.	char(32), not null
ANOMALY_DETECTION_IDX	Pointer to table 'anomalydetection'.	char(32), not null

Table 1-16 Anomaly Detections schema *(continued)*

Database Field Name	Comment	Data Type
STATUS	The scan detection status. Currently always 1 to mean "successful detection performed". Other values are reserved for future use.	int, not null
LOG_SESSION_GUID	The LOG_SESSION_GUID is an ID that the client uses to keep track of related threat events.	char(32), not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row: 0 = not deleted 1 = deleted	tinyint, not null
ID*	Primary Key (added 11.0.1).	char(32), not null

Anomaly Remediation schema (ANOMALYREMEDIATION table)

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_ANOMALYREMEDIATION.

Table 1-17 Anomaly Remediation schema

Database Field Name	Comment	Data Type
ANOMALY_REMEDIATION_IDX*	Primary Key.	char(32), not null
ANOMALY_REMEDIATION_ OPERATION_ID	Pointer to table 'anomalyremediationoperation'.	int, not null

Table 1-17 Anomaly Remediation schema *(continued)*

Database Field Name	Comment	Data Type
ANOMALY_REMEDIATION_TYPE_ID	Pointer to table 'anomalyremediationtype'.	int, not null
ACTION_OPERAND	The file or the registry key on which this action took place.	nvarchar(512), varchar(512), not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row: 0 = not deleted 1 = deleted	tinyint, not null
ACTION_OPERAND_HASH	The hash value for the ACTION_OPERAND column.	char(32), null

Anomaly Remediation Operation schema (ANOMALYREMEDIATIONOPERATION table)

Table 1-18 describes the database schema for anomaly remediation operation information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_ANOMALYREMEDIATIONOPERATION.

Table 1-18 Anomaly Remediation Operation schema

Database Field Name	Comment	Data Type
REMEDIATION_OPERATION_ID*	Primary Key	int, not null

Table 1-18 Anomaly Remediation Operation schema *(continued)*

Database Field Name	Comment	Data Type
REMEDIATION_OPERATION_DESC		varchar(255), not null

Table 1-18 Anomaly Remediation Operation schema *(continued)*

Database Field Name	Comment	Data Type
	Remediation_Operation_ID, Remediation_Operation_Desc. A hard-coded English string that is used for a lookup. Possible values are as follows: 0 = Unknown 1 = Delete 2 = Delete Line 3 = Move 4 = Create Empty File 5 = Set 6 = Terminate 7 = Suspend 8 = Stop 9 = Remove 10 = Handle Threat 11 = Set IP Address 12 = Set Domain Name 13 = Deny Access 999 = Invalid 1001 = Move 1002 = Rename 1003 = Delete 1004 = Leave Alone 1005 = Clean 1006 = Remove Macros 1007 = Save As 1008 = Move Back 1010 = Rename Back 1011 = Undo 1012 = Bad	

Table 1-18 Anomaly Remediation Operation schema *(continued)*

Database Field Name	Comment	Data Type
	1013 = Backup 1014 = Pending 1015 = Partial 1016 = Terminate 1017 = Exclude 1018 = Reboot Processing 1019 = Clean By Deletion 1020 = Access Denied	

Anomaly Remediation Type schema (ANOMALYREMEDIATIONTYPE table)

Table 1-19 describes the database schema for anomaly remediation type information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_ANOMALYREMEDIATIONTYPE.

Table 1-19 Anomaly Remediation Type schema

Database Field Name	Comment	Data Type
REMEDIATION_TYPE_ID*	Primary Key.	int, not null

Table 1-19 Anomaly Remediation Type schema *(continued)*

Database Field Name	Comment	Data Type
REMEDIATION_TYPE_DESC	The number is the REMEDIATION_TYPE_ID and the string on the right of the equal sign is the REMEDIATION_TYPE_DESC that corresponds to the numeric ID. The English string is used as a lookup key. Possible values are as follows: 2000 = Registry 2001 = File 2002 = Process 2003 = Batch File 2004 = INI File 2005 = Service 2006 = Infected File 2007 = COM Object 2008 = Hosts File Entry 2009 = Directory 2010 = Layered Service Provider 2011 = Internet Browser Cache	varchar(255), not null

Anomaly Remediations schema (ANOMALYREMEDIATIONS table)

Table 1-20 describes the database schema for anomaly remediations information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_ANOMALYREMEDIATIONS.

Table 1-20 Anomaly Remediations schema

Database Field Name	Comment	Data Type
ALERT_EVENT_IDX	Foreign key to ALERTS.IDX.	char(32), not null
ANOMALY_REMEDIATION_IDX	Pointer to table 'anomalyremediation'.	char(32), not null
STATUS	1 = successful remediation, 0 = failed remediation, no default.	int, not null
LOG_SESSION_GUID	The ID that the client uses to keep track of related threat events.	char(32), not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row: 0 = Not deleted 1 = deleted	tinyint, not null
ID*	Primary Key (added 11.0.1).	char(32), not null

Audit Report schema (AUDIT_REPORT table)

Table 1-21 describes the database schema for audit report information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_AUDITREPORT.

Table 1-21 Audit Report schema

Database Field Name	Comment	Data Type
AUDITFILTER_IDX*	Primary Key.	char(32), not null
USER_ID*	The GUID of the administrator who created this filter.	char(32), not null

Table 1-21 Audit Report schema *(continued)*

Database Field Name	Comment	Data Type
FILTERNAME*	The name of the filter.	nvarchar(510), not null
STARTDATEFROM	The start time for the filter.	datetime, not null
STARTDATETO	The end time for the filter.	datetime, not null
RELATIVEDATETYPE	Possible values are as follows: 0 = past week 1 = past month 2 = past three months 3 = past year 4 = past 24 hours 5 = current month	int, not null
EVENTTYPE	Possible values are as follows: 0 = Policy added 1 = Policy deleted 2 = Policy edited 3 = Add shared policy upon system install 4 = Add shared policy upon system upgrade 5 = Add shared policy upon domain creation	int, null
SERVERGROUPLIST	Comma-separated domain names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
PARENTSERVERLIST	Comma-separated server names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
USERLIST	Comma-separated user names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
POLICYNAMELIST	Comma-separated policy names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null

Table 1-21 Audit Report schema *(continued)*

Database Field Name	Comment	Data Type
SITELIST	Comma-separated site names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
SORTORDER	The column/field by which to sort data.	varchar(32), not null
SORTDIR	Possible values are as follows: DESC = descending sort ASC = ascending sort	varchar(5), not null
LIMITROWS	The number of rows to use for pagination.	int, not null
USERELATIVE	Use relative dates ('on') or absolute dates.	char(2), not null
REPORT_IDX	Not used.	int, not null
REPORTINPUTS	Special parameters if a report needs them.	nvarchar(64), varchar(64), not null
USN	A USN-based serial number. This ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted flag: 0 = Not deleted 1 = Deleted	tinyint, not null

Basic Metadata schema (BASIC_METADATA table)

Table 1-22 describes the database schema for basic metadata information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_BASIC_METADATA.

Table 1-22 Basic Metadata schema

Database Field Name	Comment	Data Type
CHECKSUM	The checksum of the XML content.	char(32), not null
CONTENT	The XML content of the schema object.	image, not null
DELETED	Deleted flag: 0 = Deleted 1 = Not deleted	tinyint, not null
ID*	The GUID of the schema object.	char(32), not null
OWNER	The GUID of the owner. It only applies to a private object.	char(32), null
TIME_STAMP	The time that the database record was modified; used to resolve the merge conflict.	bigint, not null
TYPE	The type name of the schema object.	varchar(256), not null
USN	The update serial number; used by replication.	bigint, not null
DOMAIN_ID	The GUID of the domain to which the object belongs. SemRootConfig and SemSite do not have DOMAIN_ID.	char(32), null
REF_ID	The object reference ID.	varchar(32), null
NAME	The object name.	nvarchar(2000), varchar(2000), null
DESCRIPTION	The object description.	nvarchar(256), varchar(256), null
LAST_MODIFY_TIME	The last modify time.	bigint, null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null

Table 1-22 Basic Metadata schema *(continued)*

Database Field Name	Comment	Data Type
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null

Behavior Report schema (BEHAVIOR_REPORT table)

Table 1-23 describes the database schema for behavior report information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_BEHAVIORREPORT.

Table 1-23 Behavior report schema

Database Field Name	Comment	Data Type
BEHAVIORFILTER_IDX*	Primary Key.	char(32), not null
USER_ID*	The GUID of user who created this filter.	char(32), not null
FILTERNAME*	The name of the filter.	nvarchar(510), not null
STARTDATEFROM	The filter start date.	datetime, not null
STARTDATETO	The filter end date.	datetime, not null
RELATIVEDATETYPE	Possible values are as follows: 0 = past week 1 = past month 2 = past three months 3 = past year 4 = past 24 hours 5 = current month	int, not null

Table 1-23 Behavior report schema *(continued)*

Database Field Name	Comment	Data Type
BEHAVIORTYPE	Possible values are as follows: 1 = Application type 2 = Device control type	tinyint, not null
SEVERITY	Possible values are as follows: 1 = Critical 5 = Major 9 = Minor 13 = Information	int, null
EVENTTYPE	For Application Control. Possible values are as follows: 501 = Application Control Driver 502 = Application Control Rules 999 = Tamper Protection	int, null
ACTION	Possible values are as follows: 0 = Allow 1 = Block 2 = Ask 3 = Continue 4 = Terminate	tinyint, null
SERVERGROUPLIST	Comma-separated domain names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
CLIENTGROUPLIST	Comma-separated group names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
PARENTSERVERLIST	Comma-separated server names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
COMPUTERLIST	Comma-separated computer names by which to filter. These names can contain wildcard characters.	nvarchar(512), varchar(512), not null

Table 1-23 Behavior report schema *(continued)*

Database Field Name	Comment	Data Type
SITELIST	Comma-separated site names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
CALLERPROCESSLIST	Comma-separated process names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
IPADDRESSLIST	Comma-separated IP by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
USERLIST	Comma-separated user names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
TEST_MODE	Possible values are as follows: 1 = Yes 0 = No	tinyint, null
SORTORDER	The table column to sort by.	varchar(32), not null
SORTDIR	Possible values are as follows: DESC = descending order ASC = Ascending order	varchar(5), not null
LIMITROWS	The number of rows to show for pagination.	int, not null
USERELATIVE	Use relative dates ('on') or absolute dates.	char(2), not null
REPORT_IDX	Not used.	int, not null
REPORTINPUTS	Special parameters if a report needs them.	nvarchar(64), varchar(64), not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null

Table 1-23 Behavior report schema *(continued)*

Database Field Name	Comment	Data Type
DELETED	Deleted flag: 0 = Not deleted 1 = Deleted	tinyint, not null
FILE_UPDOWN	Specifies whether the file is greater than or less than. Used with the FILE_SIZE field to filter. 0 = Don't filter 1 = Greater than 2 = Less than	tinyint, not null
FILE_SIZE	The size of the file that is associated with the application control violation, in megabytes. It is used to filter.	tinyint, not null

Binary File schema (BINARY_FILE table)

Table 1-24 describes the database schema for binary file information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_BINARY_FILE.

Table 1-24 Binary File schema

Database Field Name	Comment	Data Type
CHECKSUM	The checksum of XML content.	char(32), null
CONTENT	The XML content of the schema object.	image, null
DELETED	The deleted flag of the schema object. Possible values are as follows: 1 = Deleted 0 = Not Deleted	tinyint, not null

Table 1-24 Binary File schema *(continued)*

Database Field Name	Comment	Data Type
ID*	The GUID of the schema object.	char(32), not null
OWNER	The GUID of the owner. It only applies to private object	char(32), null
TIME_STAMP	The time that the database record was modified; used to resolve the merge conflict.	bigint, not null
TYPE	The type name of the schema object.	varchar(256), null
USN	The update serial number; used by replication.	bigint, not null
DOMAIN_ID	The GUID of the domain to which the binary file belongs.	char(32), null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null

Command schema (COMMAND table)

Table 1-25 describes the database schema for command information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_COMMAND.

Table 1-25 Command schema

Database Field Name	Comment	Data Type
HARDWARE_KEY*	The hash of the computer hardware information.	char(32), not null
COMMAND_ID*	The GUID of the command object. This GUID corresponded to the ID in the Basic Metadata table.	char(32), not null
DOMAIN_ID	The domain ID currently being administered when the command is created.	char(32), not null
USN	The update serial number; used by replication.	bigint, not null
BEGIN_TIME	The time that the command was launched at the client (in GMT).	bigint, not null
LAST_UPDATE_TIME	The time of the last status that the client reported (in GMT).	bigint, not null
STATE_ID	Command status: a numeric value that corresponds to one of the following values: 0 = INITIAL 1 = RECEIVED 2 = IN_PROGRESS 3 = COMPLETED 4 = REJECTED 5 = CANCELLED 6 = ERROR When first created, the command's status = INITIAL. It indicates that the endpoint has not received it yet.	int, not null

Table 1-25 Command schema *(continued)*

Database Field Name	Comment	Data Type
SUB_STATE_ID	Command-specific status. Possible values are as follows: 0 = Success 1 = Client did not execute the command 2 = Client did not report any status 3 = Command was a duplicate and not executed 4 = Spooled command cannot restart 100 = Success 101 = Security risk found 102 = Scan was suspended 103 = Scan was aborted 105 = Scan did not return status 110 = Auto-Protect cannot be turned on 120 = LiveUpdate download is in progress 121 = LiveUpdate download failed 131 = Quarantine delete failed 132 = Quarantine delete partial success	int, null
SUB_STATE_DESC	Command-specific extra information, such as the number of files that were scanned or an error message.	nvarchar(260), varchar(260), null
ESTIMATED_DURATION	The client estimation of command duration in minutes. 0 = no estimate or negligible time.	int, not null
PERCENT_COMPLETE	Progress (0-100%) of the command that was based on estimated duration.	tinyint, not null
TIME_STAMP	The time when the command was added into the database, in milliseconds since 1970.	bigint, not null

Table 1-25 Command schema *(continued)*

Database Field Name	Comment	Data Type
DELETED	The deleted flag of the schema object: 1 = Deleted 0 = Not Deleted	tinyint, not null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		varchar(260), null
RESERVED_BINARY		varbinary(1000), null

Command Report schema (COMMAND_REPORT table)

Table 1-26 describes the database schema for command report information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_COMMANDREPORT.

Table 1-26 Command Report schema

Database Field Name	Comment	Data Type
COMMANDFILTER_IDX*	Primary Key.	char(32), not null
USER_ID*	The GUID of user who created this filter.	char(32), not null
FILTERNAME*	The name of the filter.	nvarchar(510), not null
STARTDATEFROM	The start time.	datetime, not null

Table 1-26 Command Report schema *(continued)*

Database Field Name	Comment	Data Type
STARTDATETO	The end time.	datetime, not null
RELATIVEDATETYPE	Possible values are as follows: 0 = past week 1 = past month 2 = past three months 3 = past year 4 = past 24 hours 5 = current month	int, not null
STATE_ID	Command status. Possible values are as follows: 0 = Not received 1 = Received 2 = In progress 3 = Completed 4 = Rejected 5 = Canceled 6 = Error	int, null

Table 1-26 Command Report schema *(continued)*

Database Field Name	Comment	Data Type
SUB_STATE_ID	Status Details. Possible values are as follows: 0 = Success 1 = Client did not execute the command 2 = Client did not report any status 3 = Command was a duplicate and not executed 4 = Spooled command cannot restart 101 = Security risk found 102 = Scan was suspended 103 = Scan was aborted 105 = Scan did not return status 110 = Auto-Protect cannot be turned on 120 = LiveUpdate download is in progress 121 = LiveUpdate download failed 131 = Quarantine delete failed 132 = Quarantine delete partial success	int, null
PERCENT_COMPLETE	The command progress.	tinyint, null
COMPUTERLIST	A comma-separated list of computer names to filter. These names can contain wildcard characters.	nvarchar(512), varchar(512), not null
SORTORDER	The column name in the table to sort by.	varchar(32), not null
SORTDIR	Possible values are as follows: DESC = Descending order ASC = Ascending order	varchar(5), not null
LIMITROWS	The number of rows to use for pagination.	int, not null

Table 1-26 Command Report schema *(continued)*

Database Field Name	Comment	Data Type
USERELATIVE	Use relative dates ('on') or absolute dates.	char(2), not null
REPORT_IDX	Not used.	int, not null
REPORTINPUTS	Special parameters if a report needs them.	nvarchar(64), varchar(64), not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted rows: 0 = not deleted 1 = deleted	tinyint, not null

Compliance Report schema (COMPLIANCE_REPORT table)

Table 1-27 describes the database schema for compliance report information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_COMPLIANCEREPORT.

Table 1-27 Compliance Report schema

Database Field Name	Comment	Data Type
COMPLIANCEFILTER_IDX*	Primary Key.	char(32), not null
USER_ID*	The GUID of user who created this filter.	char(32), not null
FILTERNAME*	The filter name.	nvarchar(510), not null

Table 1-27 Compliance Report schema *(continued)*

Database Field Name	Comment	Data Type
STARTDATEFROM	The start date.	datetime, not null
STARTDATETO	The end date.	datetime, not null
RELATIVEDATETYPE	Possible values are as follows: 0 = past week 1 = past month 2 = past three months 3 = past year 4 = past 24 hours 5 = current month	int, not null
COMPLIANCE_TYPE	Possible values are as follows: 1 = Enforcer Server 2 = Enforcer Client 3 = Enforcer Traffic 4 = Host Compliance 5 = Attack (Firewall logs) 6 = Device Control	tinyint, not null
SEVERITY	Possible values are as follows: 1 = Critical (which filters on SEVERITY >= 0 AND SEVERITY <= 3) 5 = Major (which filters on SEVERITY >= 4 AND SEVERITY <= 7) 9 = Minor (which filters on SEVERITY >= 8 AND SEVERITY <= 11) 13 = Info (which filters on SEVERITY >= 12 AND SEVERITY <= 15)	int, null

Table 1-27 Compliance Report schema *(continued)*

Database Field Name	Comment	Data Type
EVENT_ID		int, null

Table 1-27 Compliance Report schema *(continued)*

Database Field Name	Comment	Data Type
	Events for Enforcer Server. Possible values are as follows: 1 = Enforcer registered 2 = Enforcer failed to register 5 = Enforcer downloaded policy 7 = Enforcer downloaded sylink.xml 9 = Server received Enforcer log 12 = Server received Enforcer information Events for Enforcer Traffic. Possible values are as follows: 17 = Incoming traffic blocked 18 = Outgoing traffic blocked 33 = Incoming traffic allowed 34 = Outgoing traffic allowed Events for Host compliance. Possible values are as follows: 209 = Host Integrity failed 210 = Host Integrity passed 221 = Host Integrity check failed but reported as PASS 237 = Host Integrity custom log entry Events for Attack (firewall). Possible values are as follows: 207 = Active Response 211 = Active Response disengaged 219 = Active Response canceled 217 = Executable file change accepted 218 = Executable file change denied 220 = Application Hijack	

Table 1-27 Compliance Report schema *(continued)*

Database Field Name	Comment	Data Type
	201 = N/A (invalid traffic by rule) 202 = Port Scan 203 = Denial-of-service attack 204 = Trojan horse 206 = Intrusion Prevention 208 = MAC Spoofing Events for Device control: 238 = Device control disabled device	
BLOCKED	Possible values are as follows: 0 = Blocked 1 = Not Blocked	tinyint, null
NETWORK_PROTOCOL	Possible values are as follows: 1 = Other 2 = TCP 3 = UDP 4 = ICMP	tinyint, null
TRAFFIC_DIRECTION	Possible values are as follows: 1 = Inbound 2 = Outbound 0 = Unknown	tinyint, null
SERVERGROUPLIST	Comma-separated domain names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
CLIENTGROUPLIST	Comma-separated group names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
PARENTSERVERLIST	Comma-separated server names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null

Table 1-27 Compliance Report schema *(continued)*

Database Field Name	Comment	Data Type
COMPUTERLIST	Comma separate computer names by which to filter. These names can contain wildcard characters.	nvarchar(512), varchar(512), not null
IPADDRESSLIST	Comma-separated IP list by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
USERLIST	Comma-separated user names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
SITELIST	Comma-separated site names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
ENFORCERLIST	Comma-separated Enforcer names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
REMOTEHOSTLIST	Comma-separated remote computer names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
REMOTEIPLIST	Comma-separated remote IP list by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
LOCAL_PORT	The port number.	int, null
HACK_TYPE	Possible values are as follows: 0 = Process is not running 1 = Signature is out-of-date 2 = Recovery was tried	int, null

Table 1-27 Compliance Report schema *(continued)*

Database Field Name	Comment	Data Type
ACTION	For Enforcer Client. Possible values are as follows: Authenticated Disconnected Passed Rejected Failed	varchar(32), not null
ENFORCER_TYPE	For Enforcer Client. Possible values are as follows: 0 = Gateway Enforcer 1 = LAN Enforcer 2 = DHCP Enforcer 3 = Integrated Enforcer 4 = NAP Enforcer 5 = Peer-to-Peer Enforcer	tinyint, null
OS_TYPE	Possible values are as follows: 600 = Windows Vista and Windows Server 2008 502 = Windows 2003 and Windows XP 64 bit 501 = Windows XP 500 = Windows 2000 400 = Windows NT 000 = Other	int, null
SORTORDER	The log column to sort.	varchar(32), not null
SORTDIR	Possible values are as follows: DESC = Descending ASC = Ascending	varchar(5), not null
LIMITROWS	The number of rows to use for pagination.	int, not null

Table 1-27 Compliance Report schema *(continued)*

Database Field Name	Comment	Data Type
USERELATIVE	Use relative dates ('on') or absolute dates.	char(2), not null
REPORT_IDX	Not used.	int, not null
REPORTINPUTS	Special parameters if a report needs them.	nvarchar(64), varchar(64), not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted entry; 0 = Not deleted 1 = Deleted	tinyint, not null
FULL_CHARTS	An administrator-specified list of charts to include in the Network Threat Protection Full Report.	varchar(255), not null

Computer Application schema (COMPUTER_APPLICATION table)

Table 1-28 describes the database schema for computer application information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_COMPUTER_APPLICATION.

Table 1-28 Computer Application schema

Database Field Name	Comment	Data Type
AGENT_ID*	The GUID of the client.	char(32), not null

Table 1-28 Computer Application schema *(continued)*

Database Field Name	Comment	Data Type
DOMAIN_ID*	The GUID of the domain to which the client belongs.	char(32), not null
APP_HASH*	The hash value of the learned application record.	char(32), not null
LOCATION_ID*	The GUID of the location.	char(32), not null
COMPUTER_ID	The GUID of the computer.	char(32), not null
GROUP_ID	The group GUID.	char(32), not null
LAST_ACCESS_TIME	The last access time of the application on the computer (in GMT).	bigint, null
USN	The update serial number; used by replication.	bigint, not null
TIME_STAMP	The time that the database record was modified; used to resolve the merge conflict.	bigint, not null
DELETED	The deleted flag of the schema object. Possible values are as follows: 1 = Deleted 0 = Not Deleted	tinyint, not null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null
CREATOR_SHA2	The SHA-256 of the application that created the application that is described in the record.	char(64), null

Table 1-28 Computer Application schema *(continued)*

Database Field Name	Comment	Data Type
DOWNLOAD_URL	The URL from which a application was downloaded to the client computer. The application is listed because application learning is turned on, because the administrator set up an application to watch, or because the application was detected as a threat.	varchar(512), null
DETECTION	Specifies whether the application was involved in a detection on the client computer.	tinyint, not null

Data Handler schema (DATA_HANDLER table)

Table 1-29 describes the database schema for data handler information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_DATA_HANDLER.

Table 1-29 Data Handler schema

Database Field Name	Comment	Data Type
IDX*	Primary Key.	char(32), not null
TECH_ID	Technology extension. Possible values are as follows: AvMan CAvMan LuMan legacy SEP	varchar(255), not null

Table 1-29 Data Handler schema *(continued)*

Database Field Name	Comment	Data Type
LF_EXT	File extension. Possible values are as follows: .dat .AgentStatus .SecurityRisk .VirusScans .VirusLogs .Inventory	varchar(255), not null
LF_SORT	Sort files. Possible values are as follows: 0 = Ascending by file modification time 1 = Descending by file modification time	tinyint, not null
LF_HANDLER	Classes that handle data files. Possible values are as follows: AvMan = com.sygate.scm.server.logreader.av.LogHandler CAvMan = com.sygate.scm.server.logreader.cav.CommonLogHandler Legacy agentstatus = com.sygate.scm.server.logreader.av.AgentStatusHandler Legacy inventory = com.sygate.scm.server.logreader.av.InventoryHandler Legacy security and virus logs = com.sygate.scm.server.logreader.av.LogHandler	varchar(255), not null
STATE_HANDLER	Classes that handle state files. Possible values are as follows: SEP = com.sygate.scm.server.statereader.sep.StateHandler AvMan = com.sygate.scm.server.statereader.av.StateHandler LuMan = com.sygate.scm.server.statereader.lu.StateHandler	varchar(255), not null
VERSION	Handler version	tinyint, not null

Enforcer Client Logs 1 and 2 schema (ENFORCER_CLIENT_LOG_1 and ENFORCER_CLIENT_LOG_2 tables)

Table 1-30 describes the database schema for the Enforcer Client logs.

There are two tables for this schema. When logs are stored, Symantec Endpoint Protection Manager uses the first table until it is full. The management server then uses the second table. The data in the first table is kept intact until the second table fills. Then the management server starts to fill the first table again. This cycle is continuous.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

The key is either I_ENFORCER_CLIENT_LOG_1_LOG_IDX or I_ENFORCER_CLIENT_LOG_2_LOG_IDX. The LOG_IDX field serves as the table's unique identifier, but it is not formally classified as the table's primary key. This field has an index on it, but it is not the primary key index. This table has no primary key.

Table 1-30 Enforcer Client Logs 1 and 2 schema

Database Field Name	Comment	Data Type
USN	A USN-based serial number; this ID is not unique.	bigint, not null
DOMAIN_ID	Not used (logged as '00000000000000000000000000000000')	char(32), not null
SITE_ID	The GUID of the site to which the log belongs.	char(32), not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
EVENT_ID	No event IDs defined, logged as 0.	int, not null
EVENT_TIME	The event-generated time (in GMT).	bigint, not null
ENFORCER_ID	The GUID of the Enforcer.	char(32), not null

Table 1-30 Enforcer Client Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
ENFORCER_TYPE	Possible values are as follows: 0 = Gateway Enforcer 1 = LAN Enforcer 2 = DHCP Enforcer 3 = Integrated Enforcer 4 = NAP Enforcer 5 = Peer-to-Peer Enforcer	tinyint, not null
CLIENT_ID	Not used; logged as a 0-length string.	char(32), null
REMOTE_HOST	The remote host name.	varchar(256), null
ACTION	The Enforcer's action on this client. It is a hard-coded English string that is used as a lookup Possible values are as follows: Authenticated = Client UID is correct Rejected = Client UID is wrong or there's no client running Disconnected = Client disconnects from Enforcer or Enforcer service stops Passed = Client has passed Host Integrity check Failed = Client has failed Host Integrity check	varchar(256), null
PERIOD	The period in seconds before the Enforcer takes action on the client. Only valid when action is equal to Rejected and Disconnected. For other actions, this field must be 0.	int, null
EVENT_DESC	A description of the event. Usually, first line of the description is treated as “summary.”	nvarchar(256), varchar(256), null
REMOTE_HOST_MAC	The remote host MAC address.	varchar(17), null

Table 1-30 Enforcer Client Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
REMOTE_HOST_INFO	The remote host information.	nvarchar(128), varchar(128), null
EXTENDED_INFO		nvarchar(1024), varchar(1024), null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1	Peer-to-Peer Enforcer.	nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null
LOG_IDX		char(32), null
HI_EXECUTION_ID	The ID that the Network Access Control client generates for each Host Integrity execution.	varchar(50), null
HI_STATUS	The Host Integrity status of the Network Access Control client.	char(32), null
UID_STATUS	The UID status, such as authenticated or failed.	char(32), null
POLICY_STATUS	The policy status, such as passed, failed, or unknown.	char(32), null
RADIUS_STATUS	The RADIUS status.	char(32), null

Enforcer System Logs 1 and 2 schema (ENFORCER_SYSTEM_LOG_1 and ENFORCER_SYSTEM_LOG_2 tables)

Table 1-31 describes the database schema for the Enforcer System logs.

There are two tables for this schema. When logs are stored, Symantec Endpoint Protection Manager uses the first table until it is full. The management server

then uses the second table. The data in the first table is kept intact until the second table fills. Then the management server starts to fill the first table again. This cycle is continuous.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

The key is either I_ENFORCER_SYSTEM_LOG_1_LOG_IDX or I_ENFORCER_SYSTEM_LOG_2_LOG_IDX. The LOG_IDX field serves as the table's unique identifier, but it is not formally classified as the table's primary key. This field has an index on it, but it is not the primary key index. This table has no primary key.

Table 1-31 Enforcer System Logs 1 and 2 schema

Database Field Name	Comment	Data Type
USN	A USN-based serial number; this ID is not unique.	bigint, not null
SITE_ID	The GUID of the site to which the log belongs.	char(32), not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null

Table 1-31 Enforcer System Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
EVENT_ID		int, null

Table 1-31 Enforcer System Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
	An event ID from the Symantec Endpoint Protection client (in hex). Possible values are as follows: 0x101 = Connected to the management server 0x102 = Lost connection to the management server 0x103 = Applied a policy that was downloaded from the management server 0x104 = Failed to apply a policy that was downloaded from the management server 0x107 = Applied management server configuration 0x108 = Failed to apply the management server configuration 0x110 = Registered to the NAP management server 0x111 = Unregistered from the NAP management server 0x112 = Failed to register to the NAP management server 0x201 = Enforcer started 0x202 = Enforcer stopped 0x203 = Enforcer paused 0x204 = Enforcer resumed 0x205 = Enforcer disconnected from server 0x301 = Enforcer failover enabled 0x302 = Enforcer failover disabled 0x303 = Enforcer in standby mode 0x304 = Enforcer in primary mode 0x305 = Enforcer short	

Table 1-31 Enforcer System Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
	0x306 = Enforcer loop 0x401 = Forward engine pause 0x402 = Forward engine start 0x403 = DNS Enforcer enabled 0x404 = DNS Enforcer disabled 0x405 = DHCP Enforcer enabled 0x406 = DHCP Enforcer disabled 0x407 = Allow all enabled 0x408 = Allow all disabled 0x501 = Seat number change 0x601 = Failed to create a policy parser 0x602 = Failed to import a policy that was downloaded from the management server 0x603 = Failed to export a policy that was downloaded from the management server 0x701 = Incorrect customized attribute	
EVENT_TIME	The event-generated time (in GMT).	bigint, not null
ENFORCER_ID	GUID of the Enforcer	char(32), not null
ENFORCER_TYPE	Possible values are as follows: 0 = Gateway Enforcer 1 = LAN Enforcer 2 = DHCP Enforcer 3 = Integrated Enforcer 4 = NAP Enforcer 5 = Peer-to-Peer Enforcer	tinyint, not null

Table 1-31 Enforcer System Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
SEVERITY	The type of event. Possible values are as follows: 0 = INFO 1 = WARNING 2 = ERROR 3 = FATAL	int, not null
EVENT_DESC	A description of the event. Usually, the first line of the description is treated as “summary".	nvarchar(256), varchar(256), null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null
LOG_IDX*	The log index unique ID.	char(32), null

Enforcer Traffic Logs 1 and 2 schema (ENFORCER_TRAFFIC_LOG_1 and ENFORCER_TRAFFIC_LOG_2 tables)

Table 1-32 describes the database schema for the Enforcer Traffic logs.

There are two tables for this schema. When logs are stored, Symantec Endpoint Protection Manager uses the first table until it is full. The management server then uses the second table. The data in the first table is kept intact until the second table fills. Then the management server starts to fill the first table again. This cycle is continuous.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

The key is either I_ENFORCER_TRAFFIC_LOG_1_LOG_IDX or I_ENFORCER_TRAFFIC_LOG_2_LOG_IDX. The LOG_IDX field serves as the table's unique identifier, but it is not formally classified as the table's primary key. This field has an index on it, but it is not the primary key index. This table has no primary key.

Table 1-32 Enforcer Traffic Logs 1 and 2 schema

Database Field Name	Comment	Data Type
USN	A USN-based serial number; this ID is not unique.	bigint, not null
DOMAIN_ID	Not used (logged as '00000000000000000000000000000000')	char(32), not null
SITE_ID	The GUID of the site to which the log belongs.	char(32), not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
EVENT_ID	An event ID from the Symantec Endpoint Protection client. Possible values are as follows: 17 = Incoming traffic blocked 18 = Outgoing traffic blocked 33 = Incoming traffic allowed 34 = Outgoing traffic allowed	int, null
EVENT_TIME	The event-generated time (in GMT).	bigint, not null
ENFORCER_ID	The GUID of the Enforcer.	char(32), not null

Table 1-32 Enforcer Traffic Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
ENFORCER_TYPE	Possible values are as follows: 0 = Gateway Enforcer 1 = LAN Enforcer 2 = DHCP Enforcer 3 = Integrated Enforcer 4 = NAP Enforcer 5 = Peer-to-Peer Enforcer	tinyint, not null
CLIENT_ID	Not used; logged as a 0-length string.	char(32), null
LOCAL_HOST_IP	The IP address of local computer (IPv4).	bigint, not null
REMOTE_HOST_IP	The IP address of remote computer (IPv4).	bigint, not null
NETWORK_PROTOCOL	The protocol type: Enum (OTHERS = 1; TCP = 2; UDP = 3; ICMP = 4)	tinyint, not null
LOCAL_PORT	The TCP/UDP port in the local computer (host byte-order). It is only valid on TSE_TRAFFIC_TCP and TSE_TRAFFIC_UDP. Otherwise, it is always zero.	int, not null
REMOTE_PORT	The TCP/UDP port in the remote computer (host byte-order). It is only valid on TSE_TRAFFIC_TCP and TSE_TRAFFIC_UDP. Otherwise, it is always zero.	int, not null
TRAFFIC_DIRECTION	The direction of the traffic. Enum (unknown = 0; inbound = 1; outbound = 2)	tinyint, not null
BEGIN_TIME	The start time of the Enforcer event.	bigint, null
END_TIME	The end time of the Enforcer event.	bigint, null

Table 1-32 Enforcer Traffic Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
BLOCKED	Specifies if the traffic was blocked. Possible values are as follows: 0 = blocked 1 = Not blocked. **Note:** The values in this table and those in the AGENT_TRAFFIC_LOG_x tables are different.	tinyint, not null
TOTAL_BYTES	The total length of all packets in the traffic.	int, not null
REPETITION	The number of attacks. When a hacker launches a mass attack, it may be damped to one event by the log system.	int, null
ALERT	Reserved.	tinyint, not null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null
LOG_IDX*		char(32), null

Firewall Report schema (FIREWALL_REPORT table)

Table 1-33 describes the database schema for firewall report information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first

value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_FIREWALLREPORT.

Table 1-33 Firewall Report schema

Database Field Name	Comment	Data Type
FIREWALLFILTER_IDX*	Primary Key.	char(32), not null
USER_ID	The GUID of the user who created this filter.	char(32), not null
FILTERNAME	The filter name.	nvarchar(510), not null
STARTDATEFROM	The start date.	datetime, not null
STARTDATETO	The end date.	datetime, not null
RELATIVEDATETYPE	Possible values are as follows: 0 = past week 1 = past month 2 = past three months 3 = past year 4 = past 24 hours 5 = current month	int, not null
FIREWALLTYPE	Possible values are as follows: 1 = Traffic 2 = Packet	int, not null
SEVERITY	Possible values are as follows: 1 = Critical 5 = Major 9 = Minor 13 = Info	int, null

Table 1-33 Firewall Report schema *(continued)*

Database Field Name	Comment	Data Type
EVENTTYPE	Events for Traffic. Possible values are as follows: 307 = Ethernet packet 306 = ICMP packet 308 = IP packet 303 = Ping request 301 = TCP initiated 304 = TCP completed 302 = UDP datagram 305 = Other Events for Packet: 401 = Raw Ethernet	int, null
BLOCKED	Possible values are as follows: 1 = Blocked 0 = Not blocked	int, null
PROTOCOL	Possible values are as follows: 1 = Other 2 = TCP 3 = UDP 4 = ICMP	int, null
DIRECTION	Possible values are as follows: 1 = Inbound 2 = Outbound 0 = Unknown	int, null
LOCALPORT	The port number.	int, null
SITELIST	Comma-separated site names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null

Table 1-33 Firewall Report schema *(continued)*

Database Field Name	Comment	Data Type
SERVERGROUPLIST	Comma-separated domain names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
CLIENTGROUPLIST	Comma-separated group names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
PARENTSERVERLIST	Comma-separated server names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
COMPUTERLIST	Comma-separated computer names by which to filter. These names can contain wildcard characters.	nvarchar(512), varchar(512), not null
IPADDRESSLIST	Comma-separated IP list by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
REMOTEHOSTLIST	Comma-separated remote computer names by which to filter.	nvarchar(255), varchar(255), not null
REMOTEIPADDRLIST	Comma-separated remote IP list by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
USERLIST	Comma-separated user names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
SORTORDER	The column in the table to sort by.	varchar(32), not null
SORTDIR	The direction in which to sort. Possible values are as follows: DESC = Descending ASC = Ascending	varchar(5), not null
LIMITROWS	The number of rows to use for pagination.	int, not null
USERELATIVE	Use relative dates ('on') or absolute dates.	char(2), not null
REPORT_IDX	Not used.	int, not null

Table 1-33 Firewall Report schema *(continued)*

Database Field Name	Comment	Data Type
REPORTINPUTS	Special parameters if report needs them	nvarchar(64), varchar(64), not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Delete row. 0 = Not deleted 1 = Deleted	tinyint, not null
FULL_CHARTS	Not used.	varchar(255), not null

Group Host Integrity status schema (GROUP_HI_STATUS table)

Table 1-34 lists schema information that about the group for which Host Integrity is enabled or disabled.

An asterisk (*) by a database field name indicates that the field acts as the Primary Key, PK_GROUP_HI_STATUS.

Table 1-34 Group Host Integrity schema

Database Field Name	Comment	Data Type
DOMAIN_ID	The domain name that the group belongs to.	char(32), not null
GROUP_ID*	The group ID.	char(32), not null
HI_ENABLED	Specifies whether Host Integrity is enabled.	tinyint, not null

GUI Parameters schema (GUIPARMS table)

Table 1-35 describes the database schema for GUI parameters information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_GUIPARMS.

Table 1-35 GUI Parameters schema

Database Field Name	Comment	Data Type
GUIPARMS_IDX*	Primary Key.	int, not null
PARAMETER	The parameter name.	varchar(255), not null
VALUE	The parameter value.	nvarchar(255), varchar(255), not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Delete row: 0 = Not deleted 1 = Deleted	tinyint, not null

GUP List schema (GUP_LIST table)

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_GUP_LIST.

Table 1-36 describes the database schema for Group Update Provider lists.

Table 1-36 GUP List schema

Database Field Name	Comment	Data Type
GUP_ID*	Primary key.	char(32), not null

Table 1-36 GUP List schema *(continued)*

Database Field Name	Comment	Data Type
COMPUTER_ID	The referencing Computer_ID from the SEM Computer table.	char(32), not null
IP_ADDRESS	The Group Update Provider's IP address.	bigint, not null
PORT	The Group Update Provider's port.	int, not null
USN	A USN-based serial number; this is not a unique ID.	bigint, not null
TIME_STAMP	The time when the event is logged into system (GMT), which is server side time	bigint, not null
DELETED	Delete row; 0 = Not deleted, 1 = Deleted	tinyint, not null

History schema (HISTORY table)

Table 1-37 describes the database schema for history information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_HISTORY.

Table 1-37 History schema

Database Field Name	Comment	Data Type
HISTORY_IDX*	Primary Key, Index.	char(32), not null
HISTORYCONFIG_IDX	Pointer to the History Configuration table.	char(32), not null
EVENT_DATETIME	The snapshot time in GMT.	bigint, not null
STAT_TYPE	The kind of data; a hard-coded English key.	varchar(64), not null
TARGET	The data.	nvarchar(256), varchar(256), not null

Table 1-37 History schema *(continued)*

Database Field Name	Comment	Data Type
STATISTIC	Summary statistic.	nvarchar(256), varchar(256), not null

History Configuration schema (HISTORYCONFIG table)

Table 1-38 describes the database schema for history configuration information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_HISTORYCONFIG.

Table 1-38 History Configuration schema

Database Field Name	Comment	Data Type
HISTORYCONFIG_IDX*	Primary Key.	char(32), not null
USER_ID	The GUID of the user who created this scheduled report.	char(32), not null
TZ_OFFSET	The time zone that is offset from when the administrator creates the scheduled report so that data can be formatted to the administrator's local time.	int, not null
FILTERNAME	The filter that is used by this scheduled report.	nvarchar(255), varchar(255), not null

Table 1-38 History Configuration schema *(continued)*

Database Field Name	Comment	Data Type
REPORT_IDX		varchar(10), not null

Table 1-38 History Configuration schema *(continued)*

Database Field Name	Comment	Data Type
	Format is Reporttype-number. For example, I-0 is the Virus Definitions Distribution. Possible values are as follows: I = Computer Status Report 0 = Virus Definitions Distribution 1 = Computers Not Checked Into Server 2 = Symantec Endpoint Protection Product Versions 3 = Intrusion Prevention Signature Distribution 4 = Client Inventory 5 = Compliance Status Distribution 6 = Client Online Status 7 = Clients With Latest Policy 8 = Client Count by Group 9 = Security Status Summary 10 = Protection Content Versions 11 =Client Migration 100 = Client Software Rollout (Snapshots) 101 = Clients Online/Offline Over Time (Snapshots) 102 = Clients With Latest Policy Over Time (Snapshots) 103 = Non-Compliant Clients Over Time (Snapshots) 104 = Virus Definition Rollout (Snapshots) A = Audit Report 0 = Policies Used B = Application and Device Control Report 0 = Top Groups With Most Alerted Application Control Logs 1 = Top Targets Blocked	

Table 1-38 History Configuration schema *(continued)*

Database Field Name	Comment	Data Type
	2 = Top Devices Blocked C = Compliance Report 0 = Network Compliance Status 1 = Compliance Status 2 = Clients by Compliance Failure Summary 3 = Compliance Failure Details 4 = Non-compliant Clients by Location F = Network Threat Protection Report 0 = Top Targets Attacked 1 = Top Sources of Attack 2 = Top Types of Attack 3 = Top Blocked Applications 4 = Attacks Over Time 5 = Security Events by Severity 6 = Blocked Applications Over Time 7 = Traffic Notifications Over Time 8 = Top Traffic Notifications 9 = Full Report R = Risk Report 0 = Infected and At Risk Computers 1 = Detection Action Summary 2 = Risk Detections Count 3 = New Risks Detected in the Network 4 = Top Risk Detections Correlation 5 = Risk Distribution Summary 6 = Risk Distribution Over Time 8 = Proactive Threat Detection Results 9 = Proactive Threat Distribution 10 = Proactive Threat Detection Over Time 11 = Action Summary for Top Risks	

Table 1-38 History Configuration schema *(continued)*

Database Field Name	Comment	Data Type
	12 = Number of Notifications 14 = Number of Notifications Over Time 13 = Weekly Outbreaks 7 = Comprehensive Risk Report S = Scan Report 0 = Scan Statistics Histogram 1 = Computers by Last Scan Time 2 = Computers Not Scanned Y = System Report 0 = Top Clients That Generate Errors 1 = Top Servers That Generate Errors 2 = Top Enforcers That Generate Errors 3 = Database Replication Failures Over Time 4 =Site Status Report	
STARTTIME	When to start generating the report; this establishes its scheduled time within the repeat schedule.	datetime, not null
LASTRUN	When the report was last generated (in GMT).	bigint, not null
RUNHOURS	Repeat schedule for this report in hours, for example: 1 = Every 1 hour 24 = Every 1 day 168 = Every week 720 = Every month	int, not null
NAME	The name of this scheduled report.	nvarchar(255), varchar(255), not null
EMAIL	A comma-separated list of email addresses to send the report to.	nvarchar(255), varchar(255), not null
DESCRIPTION	Administrator-provided description for this report.	nvarchar(255), varchar(255), not null

Table 1-38 History Configuration schema *(continued)*

Database Field Name	Comment	Data Type
DISABLED	Specifies whether the scheduled report is disabled or not. Possible values are as follows: 0 = No 1 = Yes	tinyint, not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row: 0 = Not Deleted 1 = Deleted	tinyint, not null
IS_MAIL_TO_SYS_ADMIN	Specifies whether to send a message to the system administrator.	tinyint, not null
FILTER_USER_ID	The user ID filter.	char(32), null
TZ_NAME	The time zone in which the administrator created a notification. The emailed reports display the dates in the administrator's local time zone.	varchar(255), not null

Home Page Configuration schema (HOMEPAGECONFIG table)

Table 1-39 describes the database schema for home page configuration information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_HOMEPAGECONFIG.

Table 1-39 Home Page Configuration schema

Database Field Name	Comment	Data Type
HOMEPAGECONFIG_IDX*	Primary Key.	char(32), not null
USER_NAME	The Admin GUID.	char(32), not null
PARAMETER	The parameter name.	varchar(255), not null
VALUE	The parameter value.	nvarchar(255), varchar(255), not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row: 0 = Not Deleted 1 = Deleted	tinyint, not null

HPP Alerts schema (HPP_ALERTS table)

Table 1-40 describes the database schema for the TruScan proactive threat scan event information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_HPP_ALERTS.

Table 1-40 HPP Alerts schema

Database Field Name	Comment	Data Type
IDX*	Primary Key.	char(32), not null
SENSITIVITY	The engine sensitivity setting that produced the detection (0...100).	int, not null
DETECTION_SCORE	The score of the detection (0...100).	tinyint, not null
COH_ENGINE_VERSION	The version of the TruScan engine.	varchar(64), not null

Table 1-40 HPP Alerts schema *(continued)*

Database Field Name	Comment	Data Type
DIS_SUBMIT	The recommendation of whether or not this detection should be submitted to Symantec. Possible values are as follows: 0 = No 1 = Yes	tinyint, not null
WHITELIST_REASON	The reason for whitelisting. Possible values are as follows: 0 = Not on the permitted application list 100 = Symantec permitted application list 101 = Administrator permitted application list 102 = User permitted application list	int, not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row: 0 = Not Deleted 1 = Deleted	tinyint, not null
DISPOSITION	Displays a value of 127 if there was no reputation data available for this detection.	tinyint, not null

Table 1-40 HPP Alerts schema *(continued)*

Database Field Name	Comment	Data Type
CONFIDENCE	The confidence level that produces the conviction: 0 = Unknown. 0 is actually not a valid value. 1-9 = Symantec knows very little about the file, or the file is unknown 10-24 = Low 25-64 = Medium 65-99 = High >= 100 = Extremely high [100..] Default = 0.	int, not null
PREVALENCE	The prevalence data for the application: 0 = Unknown 1-50 = Very low 51-100 = Low 101-150 = Moderate 151-200 = High 201-255, >255 = Very high Default = 0.	int, not null
URL	The URL from which a file was downloaded to the client computer. The application was detected as a threat. Default = " "	nvarchar(512), not null
WEB_DOMAIN	The Web domain.	nvarchar(126), not null
DOWNLOADER	The application that created the application that is detected as a threat. Default = " "	varchar(256), not null

Table 1-40 HPP Alerts schema *(continued)*

Database Field Name	Comment	Data Type
CIDS_ONOFF	Indicates the CIDS status: 0 = Off 1 = On 2 = Not installed 127 = Unknown Default = 127.	tinyint, not null
RISK_LEVEL	The risk level for the convicted threat. 0 = Unknown 1 or 2 = Low 3 = Medium 4 = High Default = 0.	tinyint, not null
AGREEMENT_ACK	Not used.	varchar(256), not null

HPP Application schema (HPP_APPLICATION table)

Table 1-41 describes the database schema for information for the applications that TruScan proactive threat scans detect.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_HPP_APPLICATION.

Table 1-41 HPP Application schema

Database Field Name	Comment	Data Type
APP_IDX*	Primary Key.	char(32), not null
APP_HASH	The hash for this application.	varchar(64), not null

Table 1-41 HPP Application schema *(continued)*

Database Field Name	Comment	Data Type
HASH_TYPE	The hash algorithm that was used. Possible values are as follows: 0 = MD5 1 = SHA-1 2 = SHA-256	tinyint, not null
COMPANY_NAME	The company name.	nvarchar(260), varchar(260), not null
APP_NAME	The application name.	nvarchar(260), varchar(260), not null
APP_VERSION	The application version.	nvarchar(256), varchar(256), not null
APP_TYPE	The application type. Possible values are as follows: 0 = Trojan horse worm 1 = Trojan horse worm 2 = Key logger 100 = Remote control	int, not null
FILE_SIZE	The file size.	bigint, not null
DETECTION_TYPE	The detection type. Possible values are as follows: 0 = heuristic 1 = commercial application	tinyint, not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row: 0 = Not Deleted 1 = Deleted	tinyint, not null

Table 1-41 HPP Application schema *(continued)*

Database Field Name	Comment	Data Type
HELP_VIRUS_IDX	Foreign key to VIRUS table, which provides a help ID for online Symantec write-up. See "Virus schema (VIRUS table)" on page 233.	char(32), null
FIRST_SEEN	The first seen date for the convicted application. Default is 0.	bigint, not null

Hypervisor pattern schema (HYPERVISOR_PATTERN)

Table 1-42 lists schema information for mapping hypervisor vendors.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

Table 1-42 Hypervisor pattern schema

Database Field Name	Comment	Data Type
BIOS_SERIALNUMBER_PREFIX	The prefix for the BIOS serial number.	varchar(128), null
MOTHERBOARD_MANUFACTURER_PREFIX	The prefix for the motherboard manufacturer.	varchar(128), null
BIOS_MANUFACTURER_PREFIX	The prefix for the BIOS manufacturer.	varchar(128), null
HYPERVISOR_VENDOR_ID	The vendor ID that links to the HYPERVISOR_VENDOR table. See "Hypervisor vendor schema (HYPERVISOR_VENDOR table)" on page 114.	tinyint, not null

Hypervisor vendor schema (HYPERVISOR_VENDOR table)

Table 1-43 describes the database schema information about hypervisor vendors.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_HYPERVISOR_VENDOR.

Table 1-43 Hypervisor vendor schema

Database Field Name	Comment	Data Type
HYPERVISOR_VENDOR_ID*	The vendor ID that links to HYPERVISOR_VENDOR.	tinyint, not null
HYPERVISOR_VENDOR_NAME	The vendor name.	varchar(128), null

Identity Map schema (IDENTITY_MAP table)

Table 1-44 describes the database schema for identity map information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_IDENTITY_MAP.

Table 1-44 Identity Map schema

Database Field Name	Comment	Data Type
ID*	The GUID of an object.	char(32), not null
NAME	The name of the object.	nvarchar(2000), varchar(2000), null
TYPE	The Object Type Name.	varchar(256), null
DOMAIN_ID	The GUID of the domain.	char(32), null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null

Table 1-44 Identity Map schema *(continued)*

Database Field Name	Comment	Data Type
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null
DELETED	Removes a deleted client group from certain reports. Possible values are as follows: 0 = Not deleted 1 = Deleted	tinyint, null

Inventory Current Risk schema (INVENTORYCURRENTRISK1 table)

Table 1-45 describes the database schema for inventory current risk information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_INVENTORYCURRENTRISK1.

Table 1-45 Inventory Current Risk schema

Database Field Name	Comment	Data Type
COMPUTER_IDX*	The foreign key to SEM_COMPUTER.COMPUTER_ID. See "SEM Computer schema (SEM_COMPUTER table)" on page 181.	char(32), not null
FILE_KEY*	The foreign key to ALERTS.IDX. See "Alerts schema (ALERTS table)" on page 48.	char(32), not null

Table 1-45 Inventory Current Risk schema *(continued)*

Database Field Name	Comment	Data Type
FIRST_INFECTED_TIME	The time that the unremediated risk was first detected.	bigint, not null
SOURCE	The type of scan that detected the risk.	varchar(50), not null
SCAN_TIME	The last scan time.	bigint, not null
FILENAME	The risk file name.	nvarchar(510), not null
SHA256	The SHA-256 of the risk file.	char(64), not null
VIRUSNAME_IDX	The foreign key to the VIRUS table. See "Virus schema (VIRUS table)" on page 233.	char(64), not null
DEFDATE	The date of the virus definitions that were used during the last scan.	int, not null
USN	The update serial number that is used to detect data changes.	bigint, not null
TIME_STAMP	The server time (GMT) when the event is logged on to the computer.	bigint, not null
DELETED	Deleted row: 0 = Not deleted 1 = Deleted	tinyint, not null
LOGON_USER	The user who was logged on when the risk was first detected.	nvarchar(512), null

Inventory Report schema (INVENTORYREPORT table)

Table 1-46 describes the database schema for inventory report information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_INVENTORYREPORT.

Table 1-46 Inventory Report schema

Database Field Name	Comment	Data Type
INVENTORYFILTER_IDX*	Primary Key.	char(32), not null
USER_ID*	The administrator GUID.	char(32), not null
FILTERNAME*	The user-specified name for this saved filter.	nvarchar(510), not null
LASTCHECKINTIME	The last time of check in with management server.	datetime, not null
LASTSCANTIME	The last time that the computer was scanned. Possible values are as follows: 0 = past week 1 = past month 2 = past three months 3 = past year 4 = past 24 hours 5 = current month	int, null
RELATIVEDATETYPE	The last check in time, if relative filtering was used. Possible values are as follows: 0 = past week 1 = past month 2 = past three months 3 = past year 4 = past 24 hours 5 = current month	int, not null
OPERATOR	Not used.	tinyint, not null

Table 1-46 Inventory Report schema *(continued)*

Database Field Name	Comment	Data Type
PATTERN_IDX	A hard-coded English string that is used as key (filters for Antivirus signature version). Possible values are as follows: WITHIN_RELATIVE_30 = Within the last 30 days WITHIN_RELATIVE_90 = Within the last 90 days OUTSIDE_RELATIVE_30 = Older than the last 30 days OUTSIDE_RELATIVE_90 = Older than the last 90 days or a virus definition revision that results in an < = query on that revision.	varchar(255), not null
PRODUCTVERSION	The product version by which to filter.	varchar(32), not null
PROFILE_VERSION	The profile version by which to filter	varchar(64), not null
IDS_VERSION	The intrusions detection system signature version by which to filter.	varchar(64), not null
GOOD	Not used.	varchar(5), not null
LICENSE_STATUS	Not used.	tinyint, null
STATUS	Possible values are as follows: 1 = online 0 = offline 127 = No filter (all)	tinyint, null
ONOFF	Auto-Protect Status. Possible values are as follows: 0 = filter for off 127 = No filter (all)	tinyint, null

Table 1-46 Inventory Report schema *(continued)*

Database Field Name	Comment	Data Type
TAMPER_ONOFF	Tamper Protection Status. Possible values are as follows: 0 = filter for off 127 = No filter (all)	tinyint, null
REBOOT_REQUIRED	Restart Required Status. Possible values are as follows: 1 = filter for needs restart 127 = No filter (all)	tinyint, null
AVENGINE_ONOFF	Antivirus Engine Status. Possible values are as follows: 0 = filter for off 127 = No filter (all)	tinyint, null
TPM_DEVICE	TPM device installed. Possible values are as follows: 1 = filters on device is installed 127 = No filter (all)	tinyint, null
SERVERGROUPLIST	A comma-separated list of domain names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
CLIENTGROUPLIST	A comma-separated list of group names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
PARENTSERVERLIST	A comma-separated list of server names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
SITELIST	A comma-separated list of site names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null

Table 1-46 Inventory Report schema *(continued)*

Database Field Name	Comment	Data Type
R_OS_TYPE	Possible values are as follows: 0000 = All non-Windows 0001=All Windows 0002=All Mac 0004= Mac OS X 10.4 0005= Mac OS X 10.5 0006= Mac OS X 10.6 0601 = Windows 7 0600 = Windows Vista 0502 = Windows 2003 and Windows XP 64-bit 0501 = Windows XP 0500 = Windows 2000 0400 = Windows NT 9999 = Windows Server 2008 -1 = No filter (all)	int, null
HI_STATUS	Filters on the following compliance statuses: 0 = Fail 1 = Success 2 = Pending 3 = Disabled 4 = Ignore 127 = No filter (all)	tinyint, null

Table 1-46 Inventory Report schema *(continued)*

Database Field Name	Comment	Data Type
HI_REASONCODE	Filters on the following reasons: 0 = Pass 101 = Antivirus version is out-of-date 102 = Antivirus is not running 103 = Script failed 104 = Check is incomplete 105 = Check is disabled A comma-separated, wild-carded list of computer names by which to filter. These names can contain wildcard characters. 127 = Location changed -1 = No filter (all)	int, null
SERVICE_PACK	OS service pack or % for no filter (all).	nvarchar(64), varchar(64), not null
WORSTINFECTION_IDX	Not used.	int, null
COMPUTERLIST	A comma-separated, wild-carded list of computer names by which to filter.	nvarchar(512), varchar(512), not null
IDADDRESSLIST	A comma-separated, wild-carded list of IP addresses by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
USERLIST	A comma-separated, wild-carded list of user names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
INFECTED	On = filter for infected machines	varchar(2), not null
SORTORDER	The column to use to sort for the Computer Status log.	varchar(32), not null
SORTDIR	Ascending or descending.	varchar(5), not null
FILVIEW	Not used.	varchar(16), not null
CLIENTTYPE	Not used.	varchar(32), not null

Table 1-46 Inventory Report schema *(continued)*

Database Field Name	Comment	Data Type
LIMITROWS	The number of rows to use for pagination.	int, not null
USERELATIVE	Use relative dates ('on') or absolute dates.	char(2), not null
REPORT_IDX	Not used.	int, not null
REPORTINPUTS	Special parameters if a report needs them.	nvarchar(64), varchar(64), not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row 0 = Not Deleted 1 = Deleted	tinyint, not null
FIREWALL_ONOFF	The Network Threat Protection status, which includes the following values: 0 = filter for off 127 = No filter (all)	tinyint, null
DA_ONOFF	The Download Advisor state, which includes the following values: 0 = Off 1 = On 2 = Not installed 3 = Off in admin policy 127 = Unknown	tinyint, null

Table 1-46 Inventory Report schema *(continued)*

Database Field Name	Comment	Data Type
PTP_ONOFF	The Proactive Threat Protection state, which includes the following values: 0 = Off 1 = On 2 = Not installed 3 = Off in admin policy 127 = Unknown Default = 127	tinyint, null
CIDS_ONOFF	The CIDS state, which includes the following values: 0 = Off 1 = On 2 = Not installed 3 = Off in admin policy 127 = Unknown Default = 127.	tinyint, null

Table 1-46 Inventory Report schema *(continued)*

Database Field Name	Comment	Data Type
DEPLOY_STATUS		tinyint, null

Table 1-46 Inventory Report schema *(continued)*

Database Field Name	Comment	Data Type
	An integer sent by the client to represent the current deployment status. The integer can be generated by the client itself or by the installer. These filter options include the following values: 302448896=Symantec Endpoint Protection Manager indicated an upgrade package for the client. 302448897=The client accepted the upgrade package 302448898=The client rejected the upgrade package 302449152=The client requested the upgrade package information 302449153=The client received the upgrade package information 302449408=The client did not allow the upgrade package to start downloading 302449409=The client successfully downloaded and verified the upgrade package 302449664=The client failed to apply the upgrade package 302449665=The client failed to patch the delta 302449666=The client failed to launch the upgrade package installer 302449667=The client successfully launched the final upgrade package installer 302449920=The client requests the full version of the upgrade package due to the delta's failure 302456832=Installation successful	

Table 1-46 Inventory Report schema *(continued)*

Database Field Name	Comment	Data Type
	302460928=Installation repair successful 302465024=Uninstallation successful 302469120=Installation failed and rolled back 302469121=Installation failed due to insufficient disk space 302469122=Installation failed due to a launch condition 302469123=Installation failed due to a consumer product found 302469124=Restart pending 302456833=Files copied 302469125=Installation failed due to a legacy full version package found 302469126=Installation failed due to non-elevated privileges 302469127=Installation failed due to an incompatible operating system See "SEM Agent schema (SEM_AGENT table)" on page 154.	
CIDS_BROWSER_IE_ONOFF	The Internet Explorer browser protection operational state filter option (0-4 enumeration). These values are also used for SEM_AGENT.CIDS_BROWSER_IE_ONOFF. See "SEM Agent schema (SEM_AGENT table)" on page 154.	tinyint, null
CIDS_BROWSER_FF_ONOFF	The Firefox browser protection operational state filter option (0-4 enumeration). This values are also used for SEM_AGENT.CIDS_BROWSER_FF_ONOFF.	tinyint, null

LAN Device Detected schema (LAN_DEVICE_DETECTED table)

The LAN Device Detected data table is not used in Symantec Network Access Control.

Table 1-47 describes the database schema for LAN Device Detected information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_LAN_DEVICE_DETECTED.

Table 1-47 LAN Device Detected schema

Database Field Name	Comment	Data Type
LAN_DEVICE_ID	The GUID of the device.	char(32), not null
AGENT_ID	The GUID of the client.	char(32), not null
COMPUTER_ID	The GUID of the client computer.	char(32), not null
HASH*	Link with the computer HARDWARE_KEY, Group GUID.	char(32), not null
MAC_ADDRESS*	The MAC address of the device.	varchar(18), not null
IP_ADDRESS	The IP Address of the device.	bigint, not null
DEVICE_DETECTED_TIME	The GUID of the domain.	bigint, null
ALERT	Reserved.	tinyint, null
SEND_SNMP_TRAP	Reflects the send SNMP trap action. SEND_SNMP_TRAP is true if send is true.	tinyint, null
USN	The update serial number; used by replication.	bigint, not null
TIME_STAMP	The time that the database record was modified; used to resolve the merge conflict.	bigint, not null

Table 1-47 LAN Device Detected schema *(continued)*

Database Field Name	Comment	Data Type
DELETED	The deleted flag of the schema object. Possible values are as follows: 1 = Deleted 0 = Not Deleted	tinyint, not null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		varchar(260), null
RESERVED_BINARY		varbinary(2000), null

LAN Device Excluded schema (LAN_DEVICE_EXCLUDED table)

The LAN Device Excluded data table is not used in Symantec Network Access Control.

Table 1-48 describes the database schema for LAN Device Excluded information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_LAN_DEVICE_EXCLUDED.

Table 1-48 LAN Device Excluded schema

Database Field Name	Comment	Data Type
EXCLUDED_ID*	The GUID of the record.	char(32), not null

Table 1-48 LAN Device Excluded schema *(continued)*

Database Field Name	Comment	Data Type
HASH	Link with the computer HARDWARE_KEY, Group GUID.	char(32), not null
EXCLUDE_MODE		tinyint, not null
MAC_ADDRESS	The MAC address of the device.	varchar(18), null
IP_ADDRESS	The IP Address of the device.	bigint, null
SUBNET_MASK	The subnet mask of the device.	bigint, null
IP_RANGE_START	The start of IP Address range.	bigint, null
IP_RANGE_END	The end of IP Address range.	bigint, null
USN	The update serial number; used by replication.	bigint, not null
TIME_STAMP	The time that the database record was modified; used to resolve merge conflicts.	bigint, not null
DELETED	The deleted flag of the schema object. Possible values are as follows: 0 = Deleted 1 = Not Deleted	tinyint, not null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		varchar(260), null
RESERVED_BINARY		varbinary(2000), null

Legacy Agent schema (LEGACY_AGENT table)

The Legacy Agent data table is not used in Symantec Network Access Control.

Table 1-49 describes the database schema for legacy client information, which is used for product migration.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_LEGACY_AGENT.

Table 1-49 Legacy Agent schema

Database Field Name	Comment	Data Type
LEGACY_AGENT_ID*	The client ID from a version 5.x client. Primary Key.	char(32), not null
GROUP_PATH	The group full path in SEM5.	char(260), not null
POLICY_MODE	User/Computer mode.	int, not null
LAN_SENSOR	If the client is a LAN_SENSOR.	int, not null
CLIENT_ID	The GUID in the SEM_CLIENT table.	char(32), not null
COMPUTER_ID	The GUID in the SEM_COMPUTER table.	char(32), not null
AGENT_ID	The GUID in the SEM_AGENT table.	char(32), not null
USN	Update serial number; used by replication.	bigint, not null
TIME_STAMP	The time that the database record was modified; used to resolve merge conflicts.	bigint, not null
DELETED	The deleted flag of the schema object. Possible values are as follows: 1 = Deleted 0 = Not Deleted	tinyint, not null
RESERVED_INT1		int, null

Table 1-49 Legacy Agent schema *(continued)*

Database Field Name	Comment	Data Type
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null

License schema (LICENSE table)

Table 1-50 describes the database schema information for licenses.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as the Primary Key, PK_LICENSE.

Table 1-50 License schema

Database Field Name	Comment	Data Type
ID*	The GUID of the record. Primary Key.	char(32), not null
DELETED	The deleted flag of the schema object: 1 = Deleted 0 = Not deleted	tinyint, not null
CHAINID	The foreign key to the LICENSE_CHAIN table. See "License chain schema (LICENSE_CHAIN table)" on page 134.	char(32), not null
TIME_STAMP	The time that the record was modified. Used to resolve merge conflicts.	bigint, not null

Table 1-50 License schema *(continued)*

Database Field Name	Comment	Data Type
TYPE	The license type.	char(1), not null
SERIAL_NUM	The license serial number.	char(16), not null
FULFILLMENT_ID	The license fulfilment ID.	char(16), not null
SERIAL_ID	The license serial ID.	char(16), not null
USN	The update serial number. Used to detect data changes.	bigint, not null
START_DATE	The license start date time.	bigint, not null
END_DATE	The license end date time.	bigint, not null
METER_COUNT	The seat count.	int, not null
GRACE_COUNT_PCT	The percentage of clients in the grace period.	int, not null
WARN_POLICY	The number of days before the license end date and when an expiration warning is sent.	bigint, not null
GRACE_POLICY	The number of days of the grace period. The end date includes the grace days. Expiration date = End date - Grace days	bigint, not null
PRODUCT_ID	The product code, which includes the product type (SEPE or SEPSB), version, and suffix.	varchar(32), not null
PRODUCT_NAME	The full product name, such as Symantec Endpoint Protection Small Business Edition 12.1 Trial License.	varchar(128), not null
WARN_DATE	The date to start the warning window, which is based on the END_DATE and WARN_POLICY: End date - Warn days	bigint, null
EXPIRE_DATE	Expiration date = End date - Grace days	bigint, null

Table 1-50 License schema *(continued)*

Database Field Name	Comment	Data Type
GRACE_COUNT	The actual grace count, which is based on the seat count and grace percentage.	int, null
PRODUCT_TYPE	The value that defines the product type: SEPE = Symantec Endpoint Protection, full version SEPSB = Symantec Endpoint Protection Small Business Edition	varchar(32), not null
RENEWAL_URL	The URL to obtain the license renewal, which is created using the slic library API.	varchar(256), null

License chain schema (LICENSE_CHAIN table)

Table 1-51 lists schema information that about license chains.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as the Primary Key, PK_LICENSE_CHAIN.

Table 1-51 License chain schema

Database Field Name	Comment	Data Type
ID*	The GUID of the record. Primary Key.	char(32), not null
CHECKSUM	The foreign key to the LICENSE_CHAIN tale.	char(32), not null
TIME_STAMP	The time that the record was modified. Used to resolve merge conflicts.	bigint, not null
USN	The update serial number. Used to detect data changes.	bigint, not null

Table 1-51 License chain schema *(continued)*

Database Field Name	Comment	Data Type
EXPIRE_DATE	The expiration date of the license chain. Expiration date = End date - Grace days	bigint, null
METER_COUNT	The seat count.	int, not null
GRACE_COUNT	The actual grace count, which is based on the seat count and grace percentage.	int, null
WARN_DATE	The date to start the warning window, which is based on the END_DATE and WARN_POLICY: End date - Warn days	bigint, null
RENEWAL_URL	The URL to obtain the license renewal, which is created using the slic library API.	varchar(256), null
CLIENT_PRODUCT_TYPE	The value that defines the product type: 0=Symantec Endpoint Protection 1=Symantec Network Access Control	tinyint, not null
DELETED	The deleted flag of the schema object: 1 = Deleted 0 = Not deleted	tinyint, not null

Local Metadata schema (LOCAL_METADATA table)

Table 1-52 describes the database schema for local metadata information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_LOCAL_METADATA.

Table 1-52 Local Metadata schema

Database Field Name	Comment	Data Type
ID*	The GUID.	char(32), not null
TYPE	The type of local_metadata. Supports only SemLocalSettings at this moment.	varchar(256), null
CHECKSUM	The checksum of the XML content.	char(32), null
CONTENT	The XML content of the schema object.	image, null
DELETED	The deleted flag of the schema object. Possible values are as follows: 0 = Deleted 1 = Not Deleted	tinyint, not null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null

Log Configuration schema (LOG_CONFIG table)

Table 1-53 describes the database schema for log configuration information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_LOG_CONFIG.

Table 1-53 Log Configuration schema

Database Field Name	Comment	Data Type
LOG_TYPE*	Type of the logs. Possible values are as follows: 101=SERVER_SYSTEM_LOG 102 = SERVER_ADMIN_LOG 103 = SERVER_POLICY_LOG 104 = SERVER_CLIENT_LOG 105 = SERVER_ENFORCER_LOG 201 = AGENT_SYSTEM_LOG 202 = AGENT_SECURITY_LOG 203 = AGENT_TRAFFIC_LOG 204 = AGENT_PACKET_LOG 205 = AGENT_BEHAVIOR_LOG 301 = ENFORCER_SYSTEM_LOG 302 = ENFORCER_CLIENT_LOG 303 = ENFORCER_TRAFFIC_LOG	int, not null
TABLE_LIST	The name of the tables to switch logs.	varchar(250), not null
THRESHOLD	The threshold of the log count.	int, not null
EXPIRATION	The expiration date of the logs.	int, not null
CURRENT_TABLE	The current log table name.	varchar(60), not null
CURRENT_ROWS	The current log count in the log table.	int, not null
SWITCH_TIME	The last log switch time.	bigint, null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null

Table 1-53 Log Configuration schema *(continued)*

Database Field Name	Comment	Data Type
RESERVED_varchar1		nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null

Notification schema (NOTIFICATION table)

Table 1-54 describes the database schema for notification information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_NOTIFICATION.

Table 1-54 Notification schema

Database Field Name	Comment	Data Type
NOTAG_IDX*	Primary Key, Index of notification.	char(32), not null

Table 1-54 Notification schema *(continued)*

Database Field Name	Comment	Data Type
TYPE	Possible values are as follows: VO = Risk outbreak SO = Outbreak on single computers VM = Outbreak by number of computers 1V = Single risk event NV = New risk detected ID = Virus definitions out-of-date AF = Authentication failure AFS = Authentication failure on a single server SE = System event CS = Client security alert CSS = Client security alert on individual computers CSM = Client security alert by number of computers LA = New learned application CL = Client list changed DF = Server health UM = Unmanaged computers NS = New software package ED = Enforcer is down WL = Forced or Commercial application detected	varchar(30), not null
USER_ID	The administrator GUID.	char(32), not null
TZ_OFFSET	The time zone when the administrator created the notification so that emailed reports can display dates in the administrator's local time zone.	int, not null

Table 1-54 Notification schema *(continued)*

Database Field Name	Comment	Data Type
SERVERGROUP	The name(s) of the server group(s) to which this notification applies. A comma-separated list that allows wildcard characters.	nvarchar(255), varchar(255), not null
CLIENTGROUP	The name(s) of the client group(s) to which this notification applies. A comma-separated list that allows wildcard characters.	nvarchar(255), varchar(255), not null
PARENTSERVER	The name(s) of the parent server(s) to which this notification applies. A comma-separated list that allows wildcard characters.	nvarchar(255), varchar(255), not null
COMPUTER	The name(s) of the computer(s) to which this notification applies.	nvarchar(255), varchar(255), not null
VIRUS	The name(s) of the virus(es) to which this notification applies. A comma-separated list that allows wildcard characters.	nvarchar(255), varchar(255), not null
SOURCE	The scan to which this notification applies. A hard-coded English string that is used as key. Possible values are as follows: % = all Scheduled Scan Manual Scan Real Time Scan Heuristic Scan Console Definition downloader System Startup Scan Idle Scan Manual Quarantine	varchar(255), not null

Table 1-54 Notification schema *(continued)*

Database Field Name	Comment	Data Type
ACTACTION	Possible values are as follows: % = No filter (all) 1 = Quarantined 3 = Deleted 4 = Left alone 5 = Cleaned 6 = Cleaned or macros deleted 14 = Pending repair 15 = Partially repaired 16 = Process termination pending restart 17 = Excluded 19 = Cleaned by deletion 20 = Access denied 21 = Process terminated 22 = No repair available 23 = All actions failed 98 = Suspicious	varchar(255), not null
HYPERLINK2	The hyperlink used to generate report.	nvarchar(255), varchar(255), not null
NTIMES	The number of occurrences that must occur to trigger this notification.	int, not null
XMINUTES	The time window in which ntimes events must occur to trigger the notification.	int, not null
EMAIL	A comma-separated email list to send email to when this notification is triggered.	nvarchar(255), varchar(255), not null
LASTRUN	The time stamp when this notification was last analyzed.	bigint, not null

Table 1-54 Notification schema *(continued)*

Database Field Name	Comment	Data Type
TRIGGERED	The time when the alert was last triggered.	bigint, not null
LASTRUN_DATA	Any extra data that is needed to give details in the notification email.	varchar(50), not null
CATEGORY	The virus category to which this notification applies. Possible values are as follows: >= -1 is no filter (all) >= 1 filters for Category 1 (Very Low) and above >= 2 filters for Category 2 (Low) and above >= 3 filters for Category 3 (Moderate) and above >= 4 filters for Category 4 (Severe) and above >= 5 filters for Category 5 (Very Severe) = -1 filters for unknown	varchar(10), not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row: 0 = Not Deleted 1 = Deleted	tinyint, not null
SYSTEM_EVENT	Which groups of system events.	int, not null
SECURITY_EVENT	Which groups of security events.	int, not null
DAMPER	The minimum quiet time between alerts in minutes; 0 means autodamper, which is 60 minutes	int, not null

Table 1-54 Notification schema *(continued)*

Database Field Name	Comment	Data Type
BATCH_FILE_NAME	The batch file or executable to be executed when the notification is triggered.	nvarchar(64), varchar(64), not null
NAME	The name of notification configuration.	nvarchar(255), varchar(255), not null
IS_MAIL_TO_SYS_ADMIN	The flag to send a message to the system administrator.	tinyint, not null
CLIENTPACKAGE_TYPE	The client package type.	int, not null
TZ_NAME	The time zone when the administrator creates a notification. Emailed reports display the time in the administrator's time zone.	varchar(255), not null

Notification Alerts schema (NOTIFICATIONALERTS table)

Table 1-55 describes the database schema for notification alerts information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_NOTIFICATIONALERTS.

Table 1-55 Notification Alerts schema

Database Field Name	Comment	Data Type
IDX*	Primary Key, Index of notification alert.	char(32), not null
NOTAG_IDX	The notification that triggered this alert. A pointer to table 'notification'.	char(32), not null
ALERTDATETIME	The time stamp when the alert was generated.	datetime, not null
SUBJECT	The subject of the alert.	nvarchar(255), varchar(255), not null

Table 1-55 Notification Alerts schema *(continued)*

Database Field Name	Comment	Data Type
MSG	The notification alert message text.	nvarchar(2048), not null
HYPERLINK	The link to the report with details about the alert situation.	nvarchar(512), varchar(512), not null
ACKNOWLEDGED	The flag that indicates whether the alert has been acknowledged.	int, not null
ACKNOWLEDGED_USERID	The GUID of the user who acknowledged this notification.	char(32), not null
ACKNOWLEDGED_TIME	The time when the notification was acknowledged.	datetime, not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row: 0 = Not deleted 1 = deleted	tinyint, not null

Pattern schema (PATTERN table)

Table 1-56 describes the database schema for pattern information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_PATTERN.

Table 1-56 Pattern schema

Database Field Name	Comment	Data Type
PATTERN_IDX*	Primary Key.	char(32), not null
CLIENT_MONIKER	The moniker for this content.	varchar(40), not null

Table 1-56 Pattern schema *(continued)*

Database Field Name	Comment	Data Type
PATTERN_TYPE	Virus definition = VIRUS_DEFS. Possible values are as follows: DECABI DEUCE_SIG ERASER_ENGINE PTS_CONTENT PTS_ENGINE SYKNAPPS_CAL SYKNAPPS_ENGINE SYKNAPPS_WHITELIST	nvarchar(128), varchar(128), not null
SEQUENCE	The sequence number that is associated with this definition.	int, not null
PATTERNDATE	The date when this content was released.	datetime, not null
REVISION	The revision number for this content.	int, not null
VERSION	The version number for this content.	varchar(255), not null
INSERTDATETIME	The time when this pattern information was entered into the database.	datetime, not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row: 0 = Not deleted 1 = Deleted	tinyint, not null

Process State schema (PROCESS_STATE table)

Table 1-57 describes the database schema for process synchronization.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

Table 1-57 Process state schema

Database Field Name	Comment	Data Type
ID*	Primary Key.	char(32), not null
TYPE	Sets the process state for process synchronization.	varchar(256), not null
STATUS	-1 = PROCESS_STATE_NA 0 = PROCESS_STATE_UNLOCKED 1 = PROCESS_STATE_LOCKED	int, not null
TIME_STAMP	The time (GMT) when this database record was entered or modified in the database.	bigint, not null
UPDATE_OWNER	Server ID + process name	varchar(255), null

Reports schema (REPORTS table)

The Reports data table is not used.

Table 1-58 describes the database schema for report information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_REPORTS.

Table 1-58 Reports schema (not used)

Database Field Name	Comment	Data Type
ID*	The GUID of the report object.	char(32), not null
TYPE	The type of report.	varchar(256), not null
REPORT_TIME	The report sample time.	bigint, not null

Table 1-58 Reports schema (not used) *(continued)*

Database Field Name	Comment	Data Type
SITE_ID	The GUID of the site from which the report was generated.	char(32), not null
DOMAIN_ID	The GUID of the domain to which the report belongs. The reports for system administrator do not have DOMAIN_ID.	char(32), null
CHECKSUM	The checksum of the XML content.	char(32), not null
CONTENT	The XML content of the schema object.	image, not null
USN	The update serial number; used by replication.	bigint, not null
TIME_STAMP	The time that the database record was modified; used to resolve merge conflicts.	bigint, not null
DELETED	The deleted flag of the schema object. Possible values are as follows: 1 = Deleted 0 = Not Deleted	tinyint, not null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		varchar(260), null
RESERVED_BINARY		varbinary(2000), null

Scan Report schema (SCANREPORT table)

Table 1-59 describes the database schema for scan report information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_SCANREPORT.

Table 1-59 Scan Report schema

Database Field Name	Comment	Data Type
SCANFILTER_IDX*	Primary Key.	char(32), not null
USER_ID*	The administrator GUID.	char(32), not null
FILTERNAME*	The user-specified name for this saved filter.	nvarchar(510), not null
STARTTIMEFROM	The start date.	datetime, not null
STARTTIMETO	The end date.	datetime, not null
RELATIVEDATETYPE	Possible values are as follows: 0 = past week 1 = past month 2 = past three months 3 = past year 4 = past 24 hours 5 = current month	int, not null
DURATION	The length of the scan.	int, not null
FILESCANNED	The number of files scanned.	bigint, not null
THREATS	The number of risks the scan found.	int, not null
FILESINFECTED	The number of files the scan found.	bigint, not null
SCANSTARTMESSAGE	The scan description.	nvarchar(255), varchar(255), not null
STATUS	The scan status as a hard-coded English key. Possible values are as follows: Completed, Cancelled, Started, % means no filter (all)	varchar(32), not null

Table 1-59 Scan Report schema *(continued)*

Database Field Name	Comment	Data Type
SERVERGROUPLIST	A comma-separated list of server groups by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
CLIENTGROUPLIST	A comma-separated list of client groups by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
PARENTSERVERLIST	A comma-separated list of parent servers by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
COMPUTERLIST	A comma-separated list of computers by which to filter. These names can contain wildcard characters.	nvarchar(512), varchar(512), not null
IPADDRESSLIST	A comma-separated list of IP addresses by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
USERLIST	A comma-separated list of users by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
LASTCOLUMN	Not used.	varchar(32), not null

Table 1-59 Scan Report schema *(continued)*

Database Field Name	Comment	Data Type
SORTORDER	Possible values are as follows: 'I.Computer' 'P.Parentserver' 'G.Clientgroup' 'C.Clientuser' 'S.Servergroup' 'SC.Startdatetime' 'SC.Duration' 'SC.Totalfiles' (total files scanned) 'SC.Threats' 'SC.Infected' (total files infected)	varchar(32), not null
SORTDIR	Sort direction. Possible values are as follows: desc = Descending asc = Ascending	varchar(5), not null
LIMITROWS	The number of rows to use for pagination.	int, not null
USERELATIVE	Use relative dates ('on') or absolute dates.	char(2), not null
REPORT_IDX	Not used.	int, not null
REPORTINPUTS	Special parameters if a report needs them.	nvarchar(255), varchar(255), not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row: 0 = Not Deleted 1 = Deleted	tinyint, not null

Table 1-59 Scan Report schema *(continued)*

Database Field Name	Comment	Data Type
R_OS_TYPE	Possible values are as follows: 0000 = All non-Windows 0001=All Windows 0002=All Mac 0004= Mac OS X 10.4 0005= Mac OS X 10.5 0006= Mac OS X 10.6 0601 = Windows 7 0600 = Windows Vista 0502 = Windows 2003 and Windows XP 64-bit 0501 = Windows XP 0500 = Windows 2000 0400 = Windows NT 9999 = Windows Server 2008 -1 = No filter (all)	int, null

Scans schema (SCANS table)

Table 1-60 describes the database schema for scans information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_SCANS.

Table 1-60 Scans schema

Database Field Name	Comment	Data Type
SCAN_IDX*	Primary Key.	char(32), not null
SCAN_ID	The scan ID provided by the client.	bigint, not null

Table 1-60 Scans schema *(continued)*

Database Field Name	Comment	Data Type
STARTDATETIME	The start time for the scan.	datetime, not null
STOPDATETIME	The stop time for the scan.	datetime, not null
STATUS	The scan status as a hard-coded English key. Possible values are as follows: completed = Completed canceled = Canceled started = Started	varchar(20), not null
DURATION	The length of the scan in seconds.	int, not null
COMPUTER_IDX	Foreign key to SEM_COMPUTER.COMPUTER_ID.	char(32), not null
CLIENTUSER1	The user who was logged in when the scan started.	nvarchar(64), varchar(64), not null
CLIENTUSER2	The user who was logged in when the scan ended.	nvarchar(64), varchar(64), not null
SERVERGROUP_IDX	Pointer to table IDENTITY_MAP (domain GUID).	char(32), not null
PARENTSERVER_IDX	Pointer to table IDENTITY_MAP (server GUID).	char(32), not null
CLIENTGROUP_IDX	Pointer to table IDENTITY_MAP (group GUID).	char(32), not null
MESSAGE1	The scan message when scan started.	nvarchar(255), varchar(255)not null
MESSAGE2	The scan message when the scan ended.	nvarchar(255), varchar(255), not null
THREATS	The number of threats that the scan found.	bigint, not null
INFECTED	The number of files that the scan found infected.	bigint, not null
TOTALFILES	The number of files scanned.	bigint, not null
OMITTED	The number of files omitted.	bigint, not null

Table 1-60 Scans schema *(continued)*

Database Field Name	Comment	Data Type
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row: 0 = Not deleted 1 = Deleted	tinyint, not null
SCAN_TYPE	The type of scan. Possible values are as follows: ScanNow_Quick = Active Scan ScanNow_Full = Full Scan ScanNow_Custom = Admin-defined Scan	varchar(64), not null
COMMAND_ID	Pointer to table SEM_JOB; command ID that started this scan (if any).	varchar(32), null

SCF Inventory schema (SCFINVENTORY table)

The SCF Inventory data table is not used.

Table 1-61 describes the database schema for SCF inventory information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_SCFINVENTORY.

Table 1-61 SCF Inventory schema (not used)

Database Field Name	Comment	Data Type
AGENT_ID*	Pointer to table SEM_AGENT.	char(32), not null
IPSSIGDATE	The date of the IPS signature.	datetime, null

Table 1-61 SCF Inventory schema (not used) *(continued)*

Database Field Name	Comment	Data Type
IPSSIGREV	The revision of the IPS signature.	int, null
SCFVERSION	The firewall version.	varchar(255), not null
SCFPOLICYFILE		nvarchar(510), not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row: 0 = Not deleted 1 = Deleted	tinyint, not null

SE Global schema (SE_GLOBAL table)

Table 1-62 describes the database schema for the system sequence number.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

No primary key is specified for this table.

Table 1-62 SE Global schema

Database Field Name	Comment	Data Type
SEQ_NUM	The latest USN on the site.	bigint, not null

SEM Agent schema (SEM_AGENT table)

Table 1-63 describes the database schema for client information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_SEM_AGENT.

Table 1-63 SEM Agent schema

Database Field Name	Comment	Data Type
AGENT_ID*	The GUID of the client.	char(32), not null
AGENT_TYPE	The type of client installed. Possible values are as follows: 105 = Symantec Endpoint Protection 151 = Symantec Network Access Control	varchar(64), null

Table 1-63 SEM Agent schema *(continued)*

Database Field Name	Comment	Data Type
R_OS_TYPE		int, null

Table 1-63 SEM Agent schema *(continued)*

Database Field Name	Comment	Data Type
	The operating system type on the client computer. Possible values are as follows: 262145 = Windows 95 262401 = Windows 95 OSR2 264705 = Windows 98 264961 = Windows 98 SE 285185 = Windows Millennium 17039362 = Windows NT Workstation 4.0 17104898 = Windows 2000 Professional 17105154 = Windows XP Professional 17105170 = Windows XP Home Edition 17105186 = Windows XP Home Embedded 17170434 = Windows Vista Ultimate Edition 17170435 = Windows Vista Home Basic Edition 17170436 = Windows Vista Home Premium Edition 17170437 = Windows Vista Enterprise Edition 17170439 = Windows Vista Business Edition 17170444 = Windows Vista Starter Edition 17170690 = Windows 7 50593794 = Windows NT Server 4.0 50593810 = Windows NT Server 4.0, Enterprise Edition 50659330 = Windows 2000 Server 50659346 = Windows 2000 Datacenter Server 50659362 = Windows 2000 Advanced Server 50659842 = Windows Server 2003 Family Standard Edition 50659858 = Windows Server 2003 Family Datacenter Edition 50659874 = Windows Server 2003 Family Enterprise Edition 50659890 = Windows Server 2003 Family Web Edition 50724882=Windows Server 2008 269091840 = Mac OS X 10.4	

Table 1-63 SEM Agent schema *(continued)*

Database Field Name	Comment	Data Type
	269092096 = Mac OS X 10.5 269092352 = Mac OS X 10.6 0 = OS Type Unspecified	
COMPUTER_ID	The GUID of the registered computer.	char(32), null
DOMAIN_ID	The GUID of the domain.	char(32), null
GROUP_ID	The current group GUID of the client.	char(32), null
AGENT_VERSION	The version of the client software.	nvarchar(64), varchar(64), null
PROFILE_VERSION	The current profile version of the client.	varchar(64), null
PROFILE_SERIAL_NO	The current profile serial number of the client.	varchar(64), null
PROFILE_CHECKSUM	The current profile checksum of the client.	char(32), null
IDS_VERSION	The current IDS version of the client.	varchar(64), null
IDS_SERIAL_NO	The current IDS serial number of client.	varchar(64), null
IDS_CHECKSUM	The current IDS checksum of the client.	char(32), null
HI_STATUS	The Host Integrity status. Possible values are as follows: 0 = Fail 1 = Success 2 = Pending 3 = Disabled 4 = Ignore	int, null

Table 1-63 SEM Agent schema *(continued)*

Database Field Name	Comment	Data Type
HI_REASONCODE	The reason why Host Integrity failed. Possible values are as follows: 0 = Pass 101 = Antivirus version is out-of-date 102 = Antivirus is not running 103 = Script failed 104 = Check is incomplete 105 = Check is disabled 127 = Location changed	int, null
HI_REASONDESC	The Host Integrity description	nvarchar(4000), null
CREATION_TIME	The create time of the client.	bigint, null
STATUS	The online status of the client. Possible values are as follows: 0 = offline 1 = online	tinyint, null
LAST_UPDATE_TIME	The last online time of the client.	bigint, null
LAST_SERVER_ID	The last connected server GUID.	char(32), null
LAST_SITE_ID	The last connected site GUID.	char(32), null
ATTRIBUTE_EXTENSION	Not used.	nvarchar(2000), varchar(2000), null
FULL_NAME	The employee's full name.	nvarchar(256), varchar(256), null
EMAIL	The employee's email address.	nvarchar(129), varchar(129), null
JOB_TITLE	The employee's job title.	nvarchar(128), varchar(128), null

Table 1-63 SEM Agent schema *(continued)*

Database Field Name	Comment	Data Type
DEPARTMENT	The employee's department.	nvarchar(128), varchar(128), null
EMPLOYEE_NUMBER	The employee's number.	varchar(32), null
EMPLOYMENT_STATUS	The employee's status.	varchar(16), null
OFFICE_PHONE	The employee's office number.	varchar(32), null
MOBILE_PHONE	The employee's mobile number.	varchar(32), null
HOME_PHONE	The employee's home phone number.	varchar(32), null
USN	The update serial number; used by replication.	bigint, not null
TIME_STAMP	The time that the database record was modified; used to resolve merge conflicts.	bigint, not null
DELETED	The deleted flag of the schema object. Possible values are as follows: 1 = Deleted 0 = Not Deleted	tinyint, not null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null

Table 1-63 SEM Agent schema *(continued)*

Database Field Name	Comment	Data Type
PATTERN_IDX	Pointer to table 'pattern'.	char(32), not null
AP_ONOFF	Auto-Protect status. Possible values are as follows: 1 = On 2 = Not installed 0 = Off 127 = Not reporting	tinyint, not null
INFECTED	Indicates whether the computer is infected. Possible values are as follows: 0 = Not infected 1 = Infected	tinyint, not null

Table 1-63 SEM Agent schema *(continued)*

Database Field Name	Comment	Data Type
WORSTINFECTION_IDX	Worst detection. Possible values are as follows: 0 = (Severity 0) Viral 1 = (Severity 1) Non-Viral malicious 2 = (Severity 2) Malicious 3 = (Severity 3) Antivirus - Heuristic 5 = (Severity 5) Hack tool 6 = (Severity 6) Spyware 7 = (Severity 7) Trackware 8 = (Severity 8) Dialer 9 = (Severity 9) Remote access 10 = (Severity 10) Adware 11 = (Severity 11) Jokeware 12 = (Severity 12) Client compliancy 13 = (Severity 13) Generic load point 14 = (Severity 14) Proactive Threat Scan - Heuristic 15 = (Severity 15) Cookie 9999 = No detections	int, not null
LAST_SCAN_TIME	The last scan time for this client (in GMT).	bigint, not null
LAST_VIRUS_TIME	The last time that a virus was detected on the client computer (in GMT).	bigint, not null
CONTENT_UPDATE	Accepts content updates. Possible values are as follows: 1 = yes 0 = no	tinyint, not null

Table 1-63 SEM Agent schema *(continued)*

Database Field Name	Comment	Data Type
AVENGINE_ONOFF	RTVScan status. Possible values are as follows: 1 = On 2 = Not installed 0 = Off 127 = Not reporting	tinyint, not null
TAMPER_ONOFF	Tamper Protection status. Possible values are as follows: 1 = On 2 = Not installed 0 = Off 127 = No reporting status	tinyint, not null
MAJOR_VERSION	The Symantec Endpoint Protection version.	int, not null
MINOR_VERSION	The minor version.	int, not null
REBOOT_REQUIRED	Indicates whether a restart is required: 0 = No 1 = Yes	tinyint, not null
REBOOT_REASON	Format is <component> = <reason ID>;<component> = <reason ID>... Components are as follows: AVMAN = Antivirus LUMAN = LiveUpdate FW = Network Threat Protection GUP = Group Update Provider Reasons are as follows: 1 = Risk remediation to complete 2 = Product patch to apply 3 = Content download to apply	varchar(128), not null
LICENSE_STATUS	For future use.	int, not null

Table 1-63 SEM Agent schema *(continued)*

Database Field Name	Comment	Data Type
LICENSE_EXPIRY	For future use.	bigint, not null
TIMEZONE	The time zone offset of the client computer.	int, not null
FIREWALL_ONOFF	Indicates the firewall status: 1 = On 2 = Not installed 0 = Off 127 = Not reporting	tinyint, not null
FREE_MEM	The free memory available.	bigint, null
FREE_DISK	The free disk space available.	bigint, null
LAST_DOWNLOAD_TIME	The last download time.	bigint, not null
CURRENT_CLIENT_ID	The client that logs on to this client.	char(32), null
LICENSE_ID	The Symantec Endpoint Protection license ID.	char(32), null
IS_GRACE	Indicates whether the license is in the grace period.	tinyint, not null
SNAC_LICENSE_ID	The Symantec Network Access Control license ID.	char(32), null
PTP_ONOFF	The Proactive Threat Protection status, which includes the following values: 0 = Off 1 = On 2 = Not installed 3 = Off in admin policy 127 = Unknown. Default is 127	tinyint, not null
LAST_HEURISTIC_THREAT_TIME	The last time that SONAR detected a risk.	bigint, not null

Table 1-63 SEM Agent schema *(continued)*

Database Field Name	Comment	Data Type
BASH_STATUS	The SONAR status, which includes the following values: 0 = Off 1= On 2 = Not installed 3 = Off in the policy 4 = Malfunction. It is meant to be for more granular op-state, but currently it is the same as PTP_ONOFF.	tinyint, not null
DA_ONOFF	The Download Advisor status, which includes the following values: 0 = Off 1 = On 2 = Not installed 3 = Off in the policy 127 = Unknown. Default is 127	tinyint, not null
CIDS_DRV_ONOFF	The CIDS status, which includes the following values: 0 = Off 1 = On 2 = Not installed 3 = Off in the policy 127 = Unknown. Default is 127	tinyint, not null
CIDS_SILENT_MODE	Indicates whether the IDS driver is installed as an internal component for another protection technology. 0 = No 1 = Yes	tinyint, not null

Table 1-63 SEM Agent schema *(continued)*

Database Field Name	Comment	Data Type

CIDS_DRV_MULF_CODE	The IDS error code if its operational status = 4.
	
	`enum NetworkProtectionErrors`
	`{`
	`eIPSOk = 0,`
	`eIPSGeneralError,`
	`eDriverNotLoaded,`
	`eAutoblockFailure,`
	`eIDSEngineManagerFailure,`
	`eSignatureManagerFailure,`
	`eNetworkExclisionManagerFailure,`
	`eNetworkInfoManagerFailure,`
	`eUDPTrafficManagerFailure,`
	`eSymEfaManagerFailure,`
	`eProcessTrackerFailure,`
	`eSettingsManagerFailure,`
	`eWFPHookManagerFailure,`
	`eLastNetworkProtectionError = 0xffffffff`
	`};`
		tinyint, not null

CIDS_BROWSER_IE_ONOFF	The Internet Explorer browser protection operational status (0-4 enumeration).	tinyint, not null
CIDS_BROWSER_FF_ONOFF	The Firefox browser protection operational status (0-4 enumeration).	tinyint, not null
CIDS_ENGINE_VERSION	The IDS engine version.	varchar(20), null
CIDS_DEFSET_VERSION	The IDS definition version. You do not use this version number in queries. Instead, use the definition version that is in the SEM_CONTENT table and the PATTERN table. See "Pattern schema (PATTERN table)" on page 144. See "SEM Content schema (SEM_CONTENT table)" on page 184.	varchar(20), null

Table 1-63 SEM Agent schema *(continued)*

Database Field Name	Comment	Data Type
DEPLOY_STATUS		int, not null

Table 1-63 SEM Agent schema *(continued)*

Database Field Name	Comment	Data Type
	The integer used to represent the current deployment status. The integer can be generated by the client itself or by the installer. 302448896 = Symantec Endpoint Protection Manager indicated an upgrade package for the client. 302448897 = The client accepted the upgrade package 302448898 = The client rejected the upgrade package 302449152 = The client requested the upgrade package information 302449153 = The client received the upgrade package information 302449408 = The client did not allow the upgrade package to start downloading 302449409 = The client successfully downloaded and verified the upgrade package 302449664 = The client failed to apply the upgrade package 302449665 = The client failed to patch the delta 302449666 = The client failed to launch the upgrade package installer 302449667 = The client successfully launched the final upgrade package installer 302449920 = The client requests the full version of the upgrade package due to the delta's failure 302456832 = Installation successful 302460928 = Installation repair successful 302465024 = Uninstallation successful 302469120 = Installation failed and rolled back 302469121 = Installation failed due to insufficient disk space 302469122 = Installation failed due to a launch condition 302469123 = Installation failed due to a consumer product found 302469124 = Restart pending 302456833 = Files copied	

Table 1-63 SEM Agent schema *(continued)*

Database Field Name	Comment	Data Type
	302469125 = Installation failed due to a legacy full version package found 302469126 = Installation failed due to non-elevated privileges 302469127 = Installation failed due to an incompatible operating system	
DEPLOY_MSG	A customizable details message that the client sends about the deployment status.	nvarchar(8000), null
DEPLOY_PRE_VER	The client version before a deployment action.	varchar(64), null
DEPLOY_TARGET_VER	The client version the deployment action tries to target.	varchar(64), null
DEPLOY_RUNNING_VER	The current client version.	varchar(64), null
DEPLOY_TIMESTAMP	The time of the deployment action.	bigint, not null

SEM Application schema (SEM_APPLICATION table)

Table 1-64 describes the database schema for application information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_SEM_APPLICATION.

Table 1-64 SEM Application schema

Database Field Name	Comment	Data Type
DOMAIN_ID*	The GUID of the domain.	char(32), not null
APP_HASH*	The checksum of the learned application, which includes the name, path, file checksum, file size, and so on.	char(32), not null

Table 1-64 SEM Application schema *(continued)*

Database Field Name	Comment	Data Type
APPLICATION_NAME	The name of the learned application.	nvarchar(260), varchar(260), not null
APPLICATION_PATH	The path of the learned application.	nvarchar(260), varchar(260), null
APP_DESCRIPTION	The description of the learned application.	nvarchar(1024), varchar(1024), null
CHECKSUM	The file checksum of the application binary.	char(32), not null
FILE_SIZE	The file size of the application binary.	bigint, null
VERSION	The file version of the application binary.	varchar(256), null
LAST_MODIFY_TIME	The last modification time of the application binary.	bigint, null
USN	The update serial number; used by replication.	bigint, not null
TIME_STAMP	The time that the database record was modified; used to resolve the merge conflicts.	bigint, not null
DELETED	The deleted flag of the schema object. Possible values are as follows: 1 = Deleted 0 = Not Deleted	tinyint, not null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null

Table 1-64 SEM Application schema *(continued)*

Database Field Name	Comment	Data Type
SHA2	The FileSHA2 algorithm.	char(64), null
SIGNER_NAME	The signer name.	nvarchar(512), null
SHA1	The FileSHA1 algorithm.	char(40), null
INTERESTING	Indicates whether the application was flagged for detection by the administrator by using the Detect Process option in the Exceptions policy.	tinyint, not null
COMPANY_NAME	The company name.	nvarchar(520), null

SEM Client schema (SEM_CLIENT table)

Table 1-65 describes the database schema for the client information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_SEM_CLIENT.

Table 1-65 SEM Client schema

Database Field Name	Comment	Data Type
CLIENT_ID*	The GUID of the client. Primary Key.	char(32), not null
DOMAIN_ID	The GUID of the domain.	char(32), null
GROUP_ID	The GUID of the group.	char(32), null
GROUP_IS_OU	If the client is from Active Directory.	tinyint, null
OU_GUID	The GUID of the Organizational Unit if the client is from the Active Directory.	char(32), null
POLICY_MODE	Enum {USER_MODE, COMPUTER_MODE}	int, null

Table 1-65 SEM Client schema *(continued)*

Database Field Name	Comment	Data Type
COMPUTER_ID	The GUID of the registered computer.	char(32), null
HARDWARE_KEY	The hash of the computer hardware information.	char(32), null
COMPUTER_NAME	The computer name.	nvarchar(64), varchar(64), null
COMPUTER_DOMAIN_NAME	The computer description.	nvarchar(256), varchar(256), null
DESCRIPTION	The domain name of the computer.	nvarchar(256), varchar(256), null
USER_NAME	The user logon name.	nvarchar(512), null
FULL_NAME	The full name of the user.	nvarchar(512), null
USER_DOMAIN_NAME	The user logon domain name.	nvarchar(256), varchar(256), null
HASH	The hash of the following: POLICY_MODE COMPUTER_NAME COMPUTER_DOMAIN_NAME USER_NAME USER_DOMAIN_NAME	char(32), not null
PIN_MARK	A flag to mark whether this client should be synchronized with Active Directory.	tinyint, null
EXTRA_FEATURE		int, null
CREATOR		tinyint, null
CREATION_TIME	The create time of the client.	bigint, null
USN	The update serial number; used by replication.	bigint, not null
TIME_STAMP	The time that the database record was modified; used to resolve the merge conflicts.	bigint, not null

Table 1-65 SEM Client schema *(continued)*

Database Field Name	Comment	Data Type
DELETED	The deleted flag of the schema object. Possible values are as follows: 1 = Deleted 0 = Not Deleted	tinyint, not null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null

SEM Compliance Criteria schema (SEM_COMPLIANCE_CRITERIA table)

Table 1-66 describes the database schema for compliance criteria information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_SEM_COMPLIANCE_CRITERIA.

Table 1-66 SEM Compliance Criteria schema

Database Field Name	Comment	Data Type
CRITERIA_IDX*	Primary Key.	char(32), not null
AGENT_SECURITY_LOG_IDX*	Foreign key to V_AGENT_SECURITY.AGENT_SECURITY_LOG_IDX.	char(32), not null

Table 1-66 SEM Compliance Criteria schema *(continued)*

Database Field Name	Comment	Data Type
ACTION	ACTION is a hard-coded English key with one of two possible values: "check" or "remediation".	varchar(64), not null
RULE_NAME	The administrator-provided rule name from the policy.	nvarchar(256), varchar(256), not null
RULE_TYPE	RULE_TYPE is a hard-coded English key with one of the following possible values: antivirus antispyware patch service pack firewall custom unknown - fallback when processing log at the server and action ends up null or blank	varchar(64), not null

Table 1-66 SEM Compliance Criteria schema *(continued)*

Database Field Name	Comment	Data Type
CRITERIA		varchar(256), not null

Table 1-66 SEM Compliance Criteria schema *(continued)*

Database Field Name	Comment	Data Type
	CRITERIA is a hard-coded English key with one of the following possible values: as_is_installed as_is_running as_signature_ok av_is_installed av_is_running av_signature_ok file_age_ok file_date_ok file_size_ok file_version_ok file_download file_exists file_checksum_ok file_execute fw_is_installed fw_is_running patch_is_installed reg_value_incr reg_key_exists reg_value_ok reg_value_exists reg_value_set timestamp_ok msg_dlg_ok os_ok os_lang_ok process_is_running – means either user application or service	

Table 1-66 SEM Compliance Criteria schema *(continued)*

Database Field Name	Comment	Data Type
	file_delete service_pack_ok hi_setup remediation – to provide an overall status of remediation unknown – fallback at the server if the criteria type is null or blank	
TARGET	The target of the criteria. For example, it can be the antivirus product name, the firewall product name, the file name, the registry key, the registry value. It can also be the patch version, the OS version, the process name, or the service name.	nvarchar(256), varchar(256), not null
RESULT	RESULT takes one of the following possible values: pass fail ignore error postponed – just for remediation criteria unknown – fallback at the server if the criteria or rule ends up without a final status	varchar(64), not null

Table 1-66 SEM Compliance Criteria schema *(continued)*

Database Field Name	Comment	Data Type
ERROR	ERROR takes one of the following possible values: unknown = unknown product_unknown = product unknown file_notfound = file not found filename_invalid = invalid file name parameter_invalid = invalid condition parameter parameter_undefined = condition parameter was not specified in the policy bad_url = URL format is invalid filedownload_op_err = URL not accessible or failed to create destination file time_out = action timed out connection_lost = connection was lost access_violation = access violation on file access_denied = access denied remediation_abort = user aborted remediation remediation_postpone = user postponed remediation createdir_failed = directory creation failed system_err = system error runas_noprivilege = a required privilege is not held by the client internal_err = internal error os_unknown = failed to detect operating system type	varchar(128), not null

Table 1-66 SEM Compliance Criteria schema *(continued)*

Database Field Name	Comment	Data Type
DESCRIPTION		nvarchar(256), varchar(256), not null

Table 1-66 SEM Compliance Criteria schema *(continued)*

Database Field Name	Comment	Data Type
	Additional compliance check details. Either exception text or one of the following values: Checksum_blank = fingerprint value is empty Failed_to_get_modification_date = failed to get modification date NAN = not a number Cannot_parse_URL = cannot parse URL URL_not_accessible_or_failed_ to_create_destination_file = URL not accessible or failed to create destination file Download_exceeded_limit = download exceeded limit Destination = destination file access violation By_User = action initiated by user Access_denied_by_server = access denied by server Download_file = download file not found Process_time_out = process timed out Failed_to_detect_OS_type = failed to detect OS type Application_name_is_empty = application name is empty Probably_software_is_not_installed = probably the software is not installed Signature_age_in_seconds_failed = cannot compute signature age Failed_to_parse_URL = failed to parse URL	

Table 1-66 SEM Compliance Criteria schema *(continued)*

Database Field Name	Comment	Data Type
	Missing_or_no_OS_version_info = missing or no version information After_script_file_running = after script file run OS_ignore = operating system check was ignored Save_failed = save failed No_previous_time = no previous time OK_or_YES = user response was OK or Yes = user response was Cancel or No Fail_to_get_current_OS_language_version = cannot retrieve current operating system language	
USN	The update serial number; used by replication.	bigint, not null
TIME_STAMP	The time that the database record was modified; used to resolve merge conflicts.	bigint, not null
DELETED	The deleted flag of the schema object. Possible values are as follows: 1 = Deleted 0 = Not Deleted	tinyint, not null

SEM Computer schema (SEM_COMPUTER table)

Table 1-67 describes the database schema for computer information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_SEM_COMPUTER.

Table 1-67 SEM Computer schema

Database Field Name	Comment	Data Type
COMPUTER_ID*	The GUID of the computer. The computer can be added from both the console and the client. Primary Key.	char(32), not null
DOMAIN_ID	The GUID of the domain.	char(32), null
HARDWARE_KEY	The hash of the computer hardware information.	char(32), null
COMPUTER_NAME	The computer name.	nvarchar(64), varchar(64), null
COMPUTER_DOMAIN_NAME	The computer description.	nvarchar(256), varchar(256), null
COMPUTER_DESCRIPTION	The domain name of the computer.	nvarchar(256), varchar(256), null
PROCESSOR_TYPE	The processor type.	nvarchar(64), varchar(64), null
PROCESSOR_CLOCK	The processor clock.	bigint, null
PROCESSOR_NUM	The number of processors.	int, null
MEMORY	The physical memory in KB.	bigint, null
BIOS_VERSION	The BIOS version.	varchar(128), null
TPM_DEVICE	The TPM device ID.	int, null
OPERATION_SYSTEM	The operation system name.	nvarchar(256), null
SERVICE_PACK	The service pack.	nvarchar(64), varchar(64), null
CURRENT_LOGIN_USER	The user who is logged in.	nvarchar(512), null
CURRENT_LOGIN_DOMAIN	The Windows domain.	nvarchar(256), varchar(256), null
DNS_SERVER1		bigint, null
DNS_SERVER2		bigint, null
WINS_SERVER1		bigint, null
WINS_SERVER2		bigint, null
DHCP_SERVER		bigint, null
MAC_ADDR1		varchar(17), null

Table 1-67 SEM Computer schema *(continued)*

Database Field Name	Comment	Data Type
IP_ADDR1		bigint, null
GATEWAY1		bigint, null
SUBNET_MASK1		bigint, null
MAC_ADDR2		varchar(17), null
IP_ADDR2		bigint, null
GATEWAY2		bigint, null
SUBNET_MASK2		bigint, null
MAC_ADDR3		varchar(17), null
IP_ADDR3		bigint, null
GATEWAY3		bigint, null
SUBNET_MASK3		bigint, null
MAC_ADDR4		varchar(17), null
IP_ADDR4		bigint, null
GATEWAY4		bigint, null
SUBNET_MASK4		bigint, null
USN	The update serial number; used by replication.	bigint, not null
TIME_STAMP	The time that the database record was modified; used to resolve the merge conflicts.	bigint, not null
DELETED	The deleted flag of the schema object. Possible values are as follows: 1 = Deleted 0 = Not Deleted	tinyint, not null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null

Table 1-67 SEM Computer schema *(continued)*

Database Field Name	Comment	Data Type
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null
DISK_TOTAL	The total disk space.	bigint, null
DISK_DRIVE	The drive letter that is referred to by DISK_TOTAL.	varchar(3), null
OS_LANG	The operating system language ID, for example, English = 0x09.	int, null
HYPERVISOR_VENDOR_ID	The foreign key to HYPERVISOR_VENDOR table. See "Hypervisor vendor schema (HYPERVISOR_VENDOR table)" on page 114.	tinyint, null

SEM Content schema (SEM_CONTENT table)

Table 1-68 describes the database schema for content information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_SEM_CONTENT.

Table 1-68 SEM Content schema

Database Field Name	Comment	Data Type
AGENT_ID*	The GUID of the client.	char(32), not null
PATTERN_IDX*	Pointer to pattern table.	char(32), not null

Table 1-68 SEM Content schema *(continued)*

Database Field Name	Comment	Data Type
USN	The update serial number; used by replication.	bigint, not null
TIME_STAMP	The time that the database record was modified; used to resolve the merge conflicts.	bigint, not null
DELETED	The deleted flag of the schema object. Possible values are as follows: 1 = Deleted 0 = Not Deleted	tinyint, not null

SEM Job schema (SEM_JOB table)

Table 1-69 describes the database schema for job information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_SEM_JOB.

Table 1-69 SEM Job schema

Database Field Name	Comment	Data Type
COMMAND_ID*	The GUID of the command object. This GUID corresponds to the ID in the Basic Metadata table.	char(32), not null
USN	The update serial number; used by replication.	bigint, not null

Table 1-69 SEM Job schema *(continued)*

Database Field Name	Comment	Data Type
COMMAND_NAME	A hard-coded English string that indicates which command was launched. This string is the same string that is placed in the XML for pre-defined name. Possible values are as follows: Update_Now = Update Content ScanNow_Full = Full Scan ScanNow_Quick = Active Scan ScanNow_Custom = Custom Scan Update_ScanNow_Full = Update Content and Scan Full Update_ScanNow_Quick = Update Content and Scan Quick Update_ScanNow_Custom = Update Content and Scan Custom CancelScan = Cancel Scan Reboot = Restart ApOn = Turn Auto-Protect On ApOff = Turn Auto-Protect Off FwOn = Turn Firewall On FwOff = Turn Firewall Off DeleteQuarantine = Delete from Quarantine	varchar(64), not null
COMMAND_DESC	A detailed description of the command.	nvarchar(350), varchar(350), null
SOURCE_SITE_ID	The GUID of the site from which the command was generated.	char(32), not null
SOURCE_ADMIN_ID	The GUID of the administrator who issued the command.	char(32), not null
CREATE_TIME	The time that the command was issued at the console by the administrator.	bigint, not null

Table 1-69 SEM Job schema *(continued)*

Database Field Name	Comment	Data Type
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row: 1 = Deleted 0 = Not Deleted	tinyint, not null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		varchar(260), null
RESERVED_BINARY		varbinary(1000), null

SEM Operating system schema (SEM_OS_INFO table)

Table 1-70 describes the database schema for operating system text information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

Table 1-70 SEM Operating system text schema

Database Field Name	Comment	Data Type
OPERATING_SYSTEM*	The operating system text.	nvarchar(256), not null
OS_NAME	The operating system name.	nvarchar(128), not null
OS_TYPE	The operating system type.	nvarchar(128), not null

Table 1-70 SEM Operating system text schema *(continued)*

Database Field Name	Comment	Data Type
SPC_OS_NAME	The operating system name in Symantec Protection Center.	nvarchar(128), not null
SPC_OS_TYPE	The operating system type in Symantec Protection Center.	nvarchar(128), not null
OS_MAJOR	The operating system, major version.	int, null
OS_MINOR	The operating system, minor version.	int, null
SPC_OS_VERSION	The operating system version in Symantec Protection Center.	nvarchar(128), not null
OS_FAMILY	The operating system family.	int, not null
DELETED	The deleted flag of the schema object. Possible values are as follows: 1 = Deleted 0 = Not deleted	tinyint, not null
USN	The update serial number, which is used by replication.	bigint, not null
TIME_STAMP	The time that the database record was modified, which is used to resolve the merge conflicts.	bigint, not null
I18N_KEY	The key value for the i18n display.	varchar(64), not null

SEM Replication state schema (SEM_REPLICATION_STATE table)

Table 1-71 describes the database schema for the update serial number (USN) generation.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

Table 1-71 SEM Replication state schema

Database Field Name	Comment	Data Type
REMOTE_SITE_ID	The GUID of the site.	char(32), not null
LOCAL_SERVER_ID	The GUID of the server.	char(32), not null
REPLICATION_STATE	Indicates whether or not replication is in process. 0 = Not 1 = Replication is in process	tinyint, not null
LAST_UPDATE_TIME	The last USN update time.	bigint, not null
USN_LIFETIME	Caches the USN life time.	bigint, not null

Serial Numbers schema (SERIAL_NUMBERS table)

Table 1-72 describes the database schema for serial number information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

No primary key is specified for this table.

Table 1-72 Serial Numbers schema

Database Field Name	Comment	Data Type
GROUP_ID	The GUID of a group.	char(32), not null
PROFILE_SERIAL_NO	The profile serial number of the group.	varchar(64), not null

Server Admin Logs 1 and 2 schema (SERVER_ADMIN_LOG_1 and SERVER_ADMIN_LOG_2 tables)

Table 1-73 describes the database schema for the Server Administration logs.

There are two tables for this schema. When logs are stored, Symantec Endpoint Protection Manager uses the first table until it is full. The management server

then uses the second table. The data in the first table is kept intact until the second table fills. Then the management server starts to fill the first table again. This cycle is continuous.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

No primary key is specified for this table.

Table 1-73 Server Admin Logs 1 and 2 schema

Database Field Name	Comment	Data Type
USN	A USN-based serial number; this ID is not unique.	bigint, not null
DOMAIN_ID	The GUID of the domain to which the log belongs.	char(32), null
SITE_ID	The GUID of the site to which the log belongs.	char(32), not null
SERVER_ID	The GUID of the server to which the log belongs.	char(32), not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
SEVERITY	Enum (SEVERE, WARNING, INFO, CONFIG, FINE, FINER, FINEST).	int, not null
ADMIN_NAME	The Administrator's name.	nvarchar(250), varchar(250), not null

Table 1-73 Server Admin Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
EVENT_ID		int, not null

Table 1-73 Server Admin Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
	The unique ID of the admin event. Possible values are as follows: 0x1001 = Login succeeded 0x1002 = Login failed 0x1003 = Log out 0x1004 = Account locked 0x1005 = Account unlocked 0x1006 = Account disabled 0x1007 = Account enabled 0x1008 = Administrator created 0x1009 = Administrator deleted 0x100A = Administrator renamed 0x100B = Password changed 0x100C = Administrator properties are changed 0x100D = Domain is created 0x100E = Domain is deleted 0x100F = Domain properties are changed 0x1020 = Domain is disabled 0x1021 = Domain is enabled 0x1022 = Domain is renamed 0x2001 = Group is created 0x2002 = Group is deleted 0x2003 = Group is renamed 0x2004 = Group is moved 0x2005 = Group properties are changed 0x2006 = User is created 0x2007 = User is deleted 0x2008 = User is moved	

Table 1-73 Server Admin Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
	0x2009 = User is copied 0x200A = User policy mode is switched 0x200B = User properties are changed 0x200C = Computer is created 0x200D = Computer is deleted 0x200E = Computer is moved 0x200F = Computer is copied 0x2010 = Computer policy mode is switched 0x2011 = Computer properties are changed 0x2012 = Organizational Unit is imported 0x2013 = Domain user is imported 0x2014 = LDAP user is imported 0x3001 = Package is created 0x3002 = Package is deleted 0x3003 = Package is exported 0x3004 = Package is moved to recycle bin 0x3005 = Package is now current 0x3006 = Package is added to other domain 0x3007 = Package properties are changed 0x3008 = Package deployment created 0x3009 = Package deployment deleted 0x300A = Package deployment properties changed 0x300B = Package updated	

Table 1-73 Server Admin Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
	0x4001 = Replication partner is registered 0x4002 = Replication partner is deleted 0x4003 = Remote site is deleted 0x4004 = Site properties are changed 0x4005 = Server properties are changed 0x4006 = Database properties are changed 0x4007 = Partner properties are change 0x4008 = Site license is changed 0x4009 = Enforcer license changed 0x4010 = Replicate now 0x4011 = Back up now 0x4012 = External logging properties are changed 0x4013 = Site backup settings changed 0x4014 = Server deleted 0x4015 = Server certificate changed 0x4016 = Enforcer group properties changed	
EVENT_DESC	A description of the event. Usually, the first line of the description is treated as “summary".	nvarchar(256), varchar(256), null
MSG_ID	The event description ID. Use this ID to load the localized message. Only used when an exception is related to this event.	int, null
ERROR_CODE	ErrorCode can uniquely identify the error in source code. Used only when an exception is related to this event.	int, null

Table 1-73 Server Admin Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
STACK_TRACE	The stack trace of the exception. Used only when an exception is related to this event.	nvarchar(2000), varchar(2000), null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(520), null
RESERVED_BINARY		varbinary(2000), null
CLIENT_ID	The GUID of the client to which the log belongs.	char(32), null

Server Client Logs 1 and 2 schema (SERVER_CLIENT_LOG_1 and SERVER_CLIENT_LOG_2 tables)

Table 1-74 describes the database schema for the Server Client logs.

There are two tables for this schema. When logs are stored, Symantec Endpoint Protection Manager uses the first table until it is full. The management server then uses the second table. The data in the first table is kept intact until the second table fills. Then the management server starts to fill the first table again. This cycle is continuous.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

The key is either I_SERVER_CLIENT_LOG_1_LOG_IDX or I_SERVER_CLIENT_LOG_2_LOG_IDX. The LOG_IDX field serves as the table's unique identifier, but it is not formally classified as the table's primary key. This

field has an index on it, but it is not the primary key index. This table has no primary key.

Table 1-74 Server Client Logs 1 and 2 schema

Database Field Name	Comment	Data Type
USN	A USN-based serial number; this ID is not unique.	bigint, not null
DOMAIN_ID	The GUID of the domain to which the log belongs.	char(32), null
SITE_ID	The GUID of the site to which the log belongs.	char(32), not null
SERVER_ID	The GUID of the server to which the log belongs.	char(32), not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null

Table 1-74 Server Client Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
EVENT_ID	The unique ID of the client activity event. Possible values are as follows: 1 = Registration succeeded 2 = Registration failed 3 = Client reconnected 4 = Client disconnected 5 = Downloaded policy 6 = Downloaded Intrusion Prevention policy 7 = Downloaded sylink.xml 8 = Downloaded auto-upgrade file 9 = Server received log 10 = Log processing failed 11 = Server received learned application 12 = Server received client information 13 = Client information processing failed 14 = Hardware identity change 15 = Downloaded File Fingerprint list 20 = Downloaded content package 22 = Downloaded command 23 = Client downloaded globalindex.dax 24 = Client downloaded the Group Update Provider list 25 = Client computer is renamed 26 = When a new client registers, the client is added to the SEM_CLIENT table	int, not null

Table 1-74 Server Client Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
AGENT_ID	The GUID of the client.	char(32), not null
HOST_NAME	The computer name of the client.	nvarchar(256), varchar(256), null
USER_NAME	The logon user name of the client.	nvarchar(256), varchar(256), null
DOMAIN_NAME	The domain name of the client.	nvarchar(256), varchar(256), null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null
LOG_IDX*	The log index unique ID.	char(32), null
CLIENT_ID	The GUID of the client to which the log belongs.	char(32), null

Server Enforcer Logs 1 and 2 schema (SERVER_ENFORCER_LOG_1 and SERVER_ENFORCER_LOG_2 tables)

Table 1-75 describes the database schema for the Server Enforcer logs.

There are two tables for this schema. When logs are stored, Symantec Endpoint Protection Manager uses the first table until it is full. The management server then uses the second table. The data in the first table is kept intact until the second table fills. Then the management server starts to fill the first table again. This cycle is continuous.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first

value applies to MS SQL Server and the second value applies to the embedded database.

The key is either I_SERVER_ENFORCER_LOG_1_LOG_IDX or I_SERVER_ENFORCER_LOG_2_LOG_IDX. The LOG_IDX field serves as the table's unique identifier, but it is not formally classified as the table's primary key. This field has an index on it, but it is not the primary key index. This table has no primary key.

Table 1-75 Server Enforcer Logs 1 and 2 schema

Database Field Name	Comment	Data Type
USN	A USN-based serial number; this ID is not unique.	bigint, not null
SITE_ID	The GUID of the site to which the log belongs.	char(32), not null
SERVER_ID	The GUID of the server to which the log belongs.	char(32), not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null

Table 1-75 Server Enforcer Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
EVENT_ID		int, not null

Table 1-75 Server Enforcer Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
	The unique ID of the Enforcer activity. Possible values are as follows: 0x101 = Connected to the management server 0x102 = Lost connection to the management server 0x103 = Applied policy that is downloaded from the management server 0x104 = Failed to apply policy that is downloaded from the management server 0x107 = Applied management server configuration 0x108 = Failed to apply management server configuration 0x201 = Enforcer started 0x202 = Enforcer stopped 0x203 = Enforcer paused 0x204 = Enforcer resumed 0x205 = Enforcer disconnected from server 0x301 = Enforcer failover enabled 0x302 = Enforcer failover disabled 0x303 = Enforcer in standby mode 0x304 = Enforcer in primary mode 0x305 = Enforcer short 0x306 = Enforcer loop 0x401 = Forward engine pause 0x402 = Forward engine start 0x403 = DNS Enforcer enabled 0x404 = DNS Enforcer disabled	

Table 1-75 Server Enforcer Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
	0x405 = DHCP Enforcer enabled 0x406 = DHCP Enforcer disabled 0x407 = Allow all enabled 0x408 = Allow all disabled 0x501 = Seat number change 0x601 = Failed to create policy parser 0x602 = Failed to import policy that is downloaded from the management server 0x603 = Failed to export policy that is downloaded from the management server 0x701 = Incorrect customized attribute	
ENFORCER_ID	The GUID of the Enforcer.	char(32), not null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(520), null
RESERVED_BINARY		varbinary(2000), null
LOG_IDX*		char(32), null

Server Policy Logs 1 and 2 schema (SERVER_POLICY_LOG_1 and SERVER_POLICY_LOG_2 tables)

Table 1-76 describes the database schema for the Server Policy logs.

There are two tables for this schema. When logs are stored, the Symantec Endpoint Protection Manager uses the first table until it is full. The management server then uses the second table. The data in the first table is kept intact until the second table fills. Then the management server starts to fill the first table again. This cycle is continuous.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

No primary key is specified for this table.

Table 1-76 Server Policy Logs 1 and 2 schema

Database Field Name	Comment	Data Type
USN	A USN-based serial number; this ID is not unique.	bigint, not null
DOMAIN_ID	The GUID of the domain which was administered.	char(32), null
SITE_ID	The GUID of the site to which the log belongs.	char(32), not null
SERVER_ID	The GUID of the server to which the log belongs.	char(32), not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null

Table 1-76 Server Policy Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
EVENT_ID	The unique ID of the policy event. Possible values are as follows: 0 = Policy added 1 = Policy deleted 2 = Policy edited 3 = Add shared policy upon system install 4 = Add shared policy upon system upgrade 5 = Add shared policy upon domain creation	int, not null
OBJECT_ID	The GUID of the client's policy.	char(32), not null
ADMIN_ID	The GUID of the administrator who modified the policy.	char(32), not null
EVENT_DESC	A description of the event. Usually, the first line of the description is treated as “summary".	nvarchar(512), null
EVENT_DATA	Additional data in binary format. This field is optional.	varbinary(2000), null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null

Server System Logs 1 and 2 schema (SERVER_SYSTEM_LOG_1 and SERVER_SYSTEM_LOG_2 tables)

Table 1-77 describes the database schema for the Server System logs.

There are two tables for this schema. When logs are stored, the Symantec Endpoint Protection Manager uses the first table until it is full. The management server then uses the second table. The data in the first table is kept intact until the second table fills. Then the management server starts to fill the first table again. This cycle is continuous.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

No primary key is specified for this table.

Table 1-77 Server System Logs 1 and 2 schema

Database Field Name	Comment	Data Type
USN	A USN-based serial number; this ID is not unique.	bigint, not null
DOMAIN_ID	Not used, logged as a 0-length string.	char(32), null
SITE_ID	The GUID of the site to which the log belongs.	char(32), not null
SERVER_ID	The GUID of the server to which the log belongs.	char(32), not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null

Table 1-77 Server System Logs 1 and 2 schema *(continued)*

Database Field Name	Comment	Data Type
SEVERITY	Enum (SEVERE, WARNING, INFO, CONFIG, FINE, FINER, FINEST): >= 400 = Finer and above >=500 = Fine and above >=700 = Configuration and above >=800 = Informational and above >=900 = Warning and above >=1000 = Severe and above	int, not null
EVENT_ID	The unique ID of the system event.	int, not null
EVENT_DESC	A description of the event; usually, the first line of description is treated as a “summary.”	nvarchar(2000), varchar(2000), null
MSG_ID	The event description ID. Use this ID to load a localized message. Only used when an exception is related to this event.	int, null
ERROR_CODE	ErrorCode can unique identify the error in source code. Only used when an exception is related to this event.	int, null
STACK_TRACE	Stack trace of exception. Only used when an exception is related to this event.	nvarchar(2000), varchar(2000), null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null

System Report schema (SYSTEM_REPORT table)

Table 1-78 describes the database schema for system report information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_SYSTEMREPORT.

Table 1-78 System Report schema

Database Field	Comment	Data Type
SYSTEMFILTER_IDX*	Primary Key.	char(32), not null
USER_ID*	The ID of the administrator who created this filter. The foreign key to the USER_ID column in the ADMINUSER table. See "Admin User schema (ADMINUSER table)" on page 16.	char(32), not null
FILTERNAME*	The filter name that the administrator provided during the save filter operation.	nvarchar(510), not null
STARTDATEFROM	The time filter start date.	datetime, not null
STARTDATETO	The time filter end date.	datetime, not null
RELATIVEDATETYPE	Possible values are as follows: 0 = past week 1 = past month 2 = past three months 3 = past year 4 = past 24 hours 5 = Enforcer Activity	int, not null
SYSTEM_TYPE	Possible values are as follows: 1 = Administrative 2 = Client server activity 3 = Server activity 4 = Client activity 5 = Enforcer Activity	tinyint, not null

Table 1-78 System Report schema *(continued)*

Database Field	Comment	Data Type
SEVERITY	For Administrative, Client-Server, and Server Activity logs, possible values are as follows: 1000 = Error and above 900 = Warning and above 800 = Informational and above -1 = No filter (all) For Enforcer activity and Client activity, possible values are as follows: 0 = Informational and above 1 = Warning and above 2 = Error and above 3 = Fatal -1 = No filter (all)	int, null

Table 1-78 System Report schema *(continued)*

Database Field	Comment	Data Type
EVENT_ID		varchar(64), not null

Table 1-78 System Report schema *(continued)*

Database Field	Comment	Data Type
	Blank or % in this field means no filtering. For the Administrative System log. For this log type, this field stores the value on the left of the = sign, for example, ADMIN_ADMIN_TYPES. It is a hard-coded English string key. To the right of the = sign are the events that are queried when the user selects the group. ADMIN_ADMIN_TYPES = Administrator events. Possible values are as follows: 4097 = Login succeeded 4098 = Login failed 4099 = Logout 4050 = Account locked 4101 = Account unlocked 4102 = Account disabled 4103 = Account enabled 4104 = Administrator created 4105 = Administrator deleted 4106 = Administrator renamed 4107 = Password changed 4108 = Administrator properties are changed ADMIN_DOMAIN_TYPES = Domain events. Possible values are as follows: 4109 = Domain is created 4110 = Domain is deleted 4111 = Domain properties are changed 4128 = Domain is disabled 4129 = Domain is enabled 4130 = Domain is renamed ADMIN_GROUP_TYPES = Group events. Possible values are as follows: 8193 = Group is created	

Table 1-78 System Report schema *(continued)*

Database Field	Comment	Data Type
	8194 = Group is deleted 8195 = Group is renamed 8196 = Group is moved 8197 = Group properties are changed ADMIN_USER_TYPES = User events. Possible values are as follows: 8198 = User is created 8199 = User is deleted 8200 = User is moved 8201 = User is copied 8202 = User policy mode is switched 8203 = User properties are changed ADMIN_COMPUTER_TYPES = Computer events. Possible values are as follows: 8204 = Computer is created 8205 = Computer is deleted 8206 = Computer is moved 8207 = Computer is copied 8208 = Computer policy mode is switched 8209 = Computer properties are changed ADMIN_IMPORT_TYPES = Import events. Possible values are as follows: 8210 = Organizational Unit is imported 8211 = Domain user is imported 8212 = LDAP user is imported ADMIN_PACKAGE_TYPES = Package events. Possible values are as follows: 12289 = Package is created 12290 = Package is deleted 12291 = Package is exported	

Table 1-78 System Report schema *(continued)*

Database Field	Comment	Data Type
	12292 = Package is moved to recycle bin 12293 = Package is now current 12294 = Package is added to other domain 12295 = Package properties are changed 12296 = Package deployment created 12297 = Package deployment deleted 12298 = Package deployment properties changed 12299 = Package updated ADMIN_REPLICATION_TYPES = Replication events. Possible values are as follows: 16385 = Replication partner is registered 16386 = Replication partner is deleted 16400 = Replicate now ADMIN_OTHER_TYPES = Other events. Possible values are as follows: 16387 = Remote site is deleted 16388 = Site properties are changed 16389 = Server properties are changed 16390 = Database properties are changed 16391 = Partner properties are changed 16392 = Site license is changed 16393 = Enforcer license changed 16394 = Replicate now 16395 = Back up now 16396 = External logging properties are changed 16397 = Site backup settings changed 16398 = Server deleted 16399 = Server certificate changed 16401 = Back up now 16402 = External logging properties are changed	

Table 1-78 System Report schema *(continued)*

Database Field	Comment	Data Type
	16403 = Site backup settings changed 16404 = Server deleted 16405 = Server certificate changed 16406 = Enforcer group properties changed For the Client-Server Activity System log. For this log type, this field stores the event ID to query. 1 = Registration succeeded 2 = Registration failed 3 = Client reconnected 4 = Client disconnected 5 = Downloaded policy 6 = Downloaded Intrusion Prevention policy 7 = Downloaded sylink.xml 8 = Downloaded auto-upgrade file 9 = Server received log 10 = Log processing failed 11 = Server received learned application 12 = Server received client information 13 = Client information processing failed 14 = Hardware identity change 15 = Downloaded File Fingerprint list 20 = Downloaded content package 22 = Downloaded command For Server Activity System log. For this log type, this field stores the hard-coded English string key that is located to the left of the = sign. To the right are listed the events that are queried for by the group. SERVER_EVENT_TYPES = Server events. Possible values are as follows: 257 = Server startup successfully 258 = Server startup failed	

Table 1-78 System Report schema *(continued)*

Database Field	Comment	Data Type
	259 = Server shut down gracefully 260 = Server created SERVER_AGENT_EVENT_TYPES = Database maintenance events. Possible values are as follows: 267 = Client sweeping started 268 = Client sweeping summary 269 = Client sweeping succeeded 270 = Client sweeping failed 271 = Database logs have been swept SERVER_BACKUP_EVENT_TYPES = Backup events. Possible values are as follows: 1025 = Backup connection failed 1026 = Backup data fetch failed 1027 = Backup file write failed 1028 = Backup unknown failed 1029 = Backup success 1030 = Backup started SERVER_RADIUS_EVENT_TYPES = Radius Server events. Possible values are as follows: 1283 = Failed to start Radius Server. The radius port may be used by another process. 1284 = Failed to start Radius Server. Set non-Block IO socket failed. 1285 = Failed to start Radius Server. Create socket error. SERVER_REPLICATION_EVENT_TYPES = Replication events. Possible values are as follows: 769 = Replication from remote site started 770 = Replication failed to login to remote site 771 = Unable to fetch changed data from remote site 772 = Replication finished successfully	

Table 1-78 System Report schema *(continued)*

Database Field	Comment	Data Type
	773 = Replication failed 774 = Replication merge failed 775 = Unable to connect to remote site 776 = Name changed to resolve merge conflict 777 = Group full path name is too long for replication 778 = Retrieval of local changed data for remote site started 779 = Retrieval of local changed data for remote site finished successfully 780 = Retrieval of local changed data for remote site failed 781 = The database had chosen to terminate replication to end the deadlock 782 = Replication data is received SERVER_IMPORT_EVENT_TYPES = Import events. Possible values are as follows: 264 = Organization importing started 265 = Organization importing succeeded 266 = Organization importing failed SERVER_INTRUSION_PREVENTION_EVEN = Policy content updates. Possible values are as follows: 1537 = Added Intrusion Prevention Library 1538 = Deleted Intrusion Prevention Library 1539 = Updated Intrusion Prevention Library 1540 = Intrusion Prevention Library is up to date SERVER_LU_EVENT_TYPES = LiveUpdate events. Possible values are as follows: 1793 = LiveUpdate started 1794 = LiveUpdate succeeded 1795 = LiveUpdate failed 1796 = LiveUpdate manual task succeeded 1797 = LiveUpdate manual task failed	

Table 1-78 System Report schema *(continued)*

Database Field	Comment	Data Type
	1798 = LiveUpdate retry started 1799 = LiveUpdate retry succeeded 1800 = LiveUpdate retry failed and will try again 1801 = LiveUpdate manual task started 1802 = LiveUpdate retry over max window 1803 = LiveUpdate retry failed and will try again 1804 = LiveUpdate retry pass scheduled time 1805 = LiveUpdate All process launched 1806 = LiveUpdate All process exited abnormally 1807 = LiveUpdate next server 1808 = LiveUpdate All process finished 1809 = LiveUpdate All process failed to launch 1810 = LiveUpdate uploading content 1811 = LiveUpdate file path does not exist 1812 = LiveUpdate Content Catalog file has been inserted 1813 = LiveUpdate Content Catalog file has been updated 1814 = Client package has been downloaded 1815 = Client package patching failed 1816 = New LiveUpdate content has been downloaded 1817 = LiveUpdate wrong URL parameter 1824 = Antivirus and antispyware definitions Win64 11.0 MicroDefsB.CurDefs failed to update 1825 = Download is current 1826 = LiveUpdate rerun is triggered by content catalog update 1818 = Failed to download LiveUpdate content 1819 = LiveUpdate content cleaned up 1820 = Host Integrity template has been updated 1821 = LiveUpdate timed out 1822 = LiveUpdate schedule updated	

Table 1-78 System Report schema *(continued)*

Database Field	Comment	Data Type
	SERVER_NET_AUDIT_EVENT_TYPES = Find unprotected computers events. Possible values are as follows: 2049 = Search uncliented hosts started 2050 = Search uncliented hosts finished normally 2051 = Search uncliented hosts finished abnormally 2052 = Client remote started 2053 = Client remote finished normally 2054 = Client remote finished abnormally SERVER_OTHER_EVENT_TYPES = Other events. Possible values are as follows: 261 = Site created 262 = Package published 263 = Site license exceeded 272 = Server upgrade success 1282 = Connect mail server failed 1286 = Server error For the Client Activity System log. For this log, this field stores the hard-coded English string key that is located to the left of the = sign. To the right are listed the events that are queried by the group. The event IDs are in hex. AGENT_SYSTEM_INSTALL_EVENT_TYPES = Installation events. Possible values are as follows: 0x12070001 = Internal error 0x12070101 = Install complete 0x12070102 = Restart recommended 0x12070103 = Restart required 0x12070104 = Installation failed 0x12070105 = Uninstallation complete 0x12070106 = Uninstallation failed 0x12071037 = Symantec AntiVirus installed	

Table 1-78 System Report schema *(continued)*

Database Field	Comment	Data Type
	0x12071038 = Symantec Firewall installed 0x12071039 = Uninstall 0x1207103A = Uninstall rolled-back AGENT_SYSTEM_SERVICE_EVENT_TYPES = Service events. Possible values are as follows: 0x12070201 = Service starting 0x12070202 = Service started 0x12070203 = Service start failure 0x12070204 = Service stopped 0x12070205 = Service stop failure 0x1207021A = Attempt to stop service AGENT_SYSTEM_CONFIG_EVENT_TYPES = Configuration events. Possible values are as follows: 0x12070206 = Configuration import complete 0x12070207 = Configuration import error 0x12070208 = Configuration export complete 0x12070209 = Configuration export error AGENT_SYSTEM_HI_EVENT_TYPES = Host Integrity events. Possible values are as follows: 0x12070210 = Host Integrity disabled 0x12070211 = Host Integrity enabled AGENT_SYSTEM_IMPORT_EVENT_TYPES = Import events. Possible values are as follows: 0x12070214 = Successfully imported advanced rule 0x12070215 = Failed to import advanced rule 0x12070216 = Successfully exported advanced rule 0x12070217 = Failed to export advanced rule AGENT_SYSTEM_CLIENT_EVENT_TYPES = Client events. Possible values are as follows:	

Table 1-78 System Report schema *(continued)*

Database Field	Comment	Data Type
	0x12070218 = Client Engine enabled 0x12070219 = Client Engine disabled 0x12071046 = Proactive Threat Scanning is not supported on this platform 0x12071047 = Proactive Threat Scanning Load Error AGENT_SYSTEM_SERVER_EVENT_TYPES = Server events. Possible values are as follows: 0x12070301 = Server connected 0x12070302 = No server response 0x12070303 = Server connection failed 0x12070304 = Server disconnected 0x120B0001 = Cannot reach server 0x120B0002 = Reconnected server AGENT_SYSTEM_PROFILE_EVENT_TYPES = Policy events. Possible values are as follows: 0x12070306 = New policy received 0x12070307 = New policy applied 0x12070308 = New policy failed 0x12070309 = Cannot download policy 0x120B0005 = Cannot download policy 0x1207030A = Have latest policy 0x120B0004 = Have latest policy AGENT_SYSTEM_AV_EVENT_TYPES = Antivirus engine events. Possible values are as follows: 0x12071006 = Scan omission 0x1207100B = Virus behavior detected 0x1207100C = Configuration changed 0x12071010 = Definition file download 0x12071012 = Sent to Quarantine Server 0x12071013 = Delivered to Symantec	

Table 1-78 System Report schema *(continued)*

Database Field	Comment	Data Type
	0x12071014 = Security Response backup 0x12071015 = Scan aborted 0x12071016 = Symantec AntiVirus Auto-Protect load error 0x12071017 = Symantec AntiVirus Auto-Protect enabled 0x12071018 = Symantec AntiVirus Auto-Protect disabled 0x1207101A = Scan delayed 0x1207101B = Scan restarted 0x12071027 = Symantec AntiVirus is using old virus definitions 0x12071041 = Scan suspended 0x12071042 = Scan resumed 0x12071043 = Scan duration too short 0x12071045 = Scan enhancements failed AGENT_SYSTEM_LICENSE_EVENT_TYPES = License events. Possible values are as follows: 0x1207101E = License warning 0x1207101F = License error 0x12071020 = License in grace period 0x12071023 = License installed 0x12071025 = License up-to-date AGENT_SYSTEM_SECURITY_EVENT_TYPES = Security events. Possible values are as follows: 0x1207102B = Computer not compliant with security policy 0x1207102C = Computer compliant with security policy 0x1207102D = Tamper attempt AGENT_SYSTEM_OTHER_EVENT_TYPES = Other events. Possible values are as follows: 0x1207020A = Email post OK 0x1207020B = Email post failure 0x1207020C = Update complete 0x1207020D = Update failure	

Table 1-78 System Report schema *(continued)*

Database Field	Comment	Data Type
	0x1207020E = Manual location change 0x1207020F = Location changed 0x12070212 = Old Rasdll detected 0x12070213 = Auto-update postponed 0x12070305 = Mode changed 0x1207030B = Cannot apply HI script 0x12070500 = System message from device control 0x12070600 = System message from anti-buffer overflow driver 0x12071021 = Access denied warning 0x12071022 = Log forwarding error 0x12071044 = Client moved For the Enforcer Activity System log. For this log, this field stores the hard-coded English string key that is located to the left of the = sign. To the right are listed the events that are queried by the group. The event IDs are in hex. ENFORCER_POLICY_MANAGER_EVENT_TY = Management events. Possible values are as follows: 0x101 = Connected to >0x102 = Lost connection to Symantec Endpoint Protection Manager 0x103 = Applied policy downloaded from the management server 0x104 = Failed to apply policy downloaded from the management server 0x107 = Applied management server configuration 0x108 = Failed to apply management server configuration ENFORCER_ENFORCER_EVENT_TYPES = Enforcer events. Possible values are as follows: 0x201 = Enforcer started 0x202 = Enforcer stopped 0x203 = Enforcer paused 0x204 = Enforcer resumed 0x205 = Enforcer disconnected from server	

Table 1-78 System Report schema *(continued)*

Database Field	Comment	Data Type
	0x301 = Enforcer failover enabled 0x302 = Enforcer failover disabled 0x303 = Enforcer in standby mode 0x304 = Enforcer in primary mode 0x305 = Enforcer short 0x306 = Enforcer loop ENFORCER_ENABLE_EVENT_TYPES = Enable events. Possible values are as follows: 0x401 = Forward engine pause 0x402 = Forward engine start 0x403 = DNS enforcer enabled 0x404 = DNS enforcer disabled 0x405 = DHCP enforcer enabled 0x406 = DHCP enforcer disabled 0x407 = Allow all enabled 0x408 = Allow all disabled ENFORCER_PROFILE_EVENT_TYPES = Policy events. Possible values are as follows: 0x501 = Seat number change 0x601 = Failed to create policy parser 0x602 = Failed to import policy downloaded from Symantec Endpoint Protection Manager 0x603 = Failed to export policy downloaded from 0x701 = Incorrect customized attribute	
EVENT_DESC		nvarchar(255), varchar(255), not null

Table 1-78 System Report schema *(continued)*

Database Field	Comment	Data Type
MSG_ID		varchar(255), not null

Table 1-78 System Report schema *(continued)*

Database Field	Comment	Data Type
	This field stores the hard-coded English string key that is found to the left of the = sign. To the right is a description of the kinds of error messages that are queried. % or blank in this field means no filtering (all records). For the Administrative System log. Possible values are as follows: ERR_SERVER = Server error messages ERR_INVALID_PARAMETER = Invalid parameter error messages ERR_GENERAL = General error messages ERR_ROOT = Root error messages ERR_AUTHENTICATION = Login-related error messages ERR_METADATA = Metadata error messages ERR_TRANSACTION = Transaction error messages ERR_DATASTORE = Datastore error messages ERR_LICENSE = License error messages ERR_CERTIFICATE = Certificate error messages ERR_GROUP = Group error messages ERR_FILE = File related error messages ERR_LIVEUPDATE = LiveUpdate error messages ERR_OTHER = Other error messages ERR_NONE = None For the Server Activity System log: ERR_SERVER = Server error messages ERR_INVALID_PARAMETER = Invalid parameter error messages ERR_GENERAL = General error messages ERR_ROOT = Root error messages ERR_AUTHENTICATION = Login-related error messages ERR_METADATA = Metadata error messages ERR_TRANSACTION = Transaction error messages ERR_DATASTORE = Datastore error messages ERR_LICENSE = License error messages	

Table 1-78 System Report schema *(continued)*

Database Field	Comment	Data Type
	ERR_CERTIFICATE = Certificate error messages ERR_GROUP = Group error messages ERR_FILE = File related error messages ERR_LIVEUPDATE = LiveUpdate error messages ERR_OTHER = Other error messages ERR_NONE = None	
ENFORCERLIST	Comma-separated Enforcer names by which to filter.	nvarchar(255), varchar(255), not null
ENFORCER_TYPE	Possible values are as follows: 0 = Gateway Enforcer 1 = LAN Enforcer 2 = DHCP Enforcer 3 = Integrated Enforcer 4 = NAP Enforcer 5 = Peer-to-Peer Enforcer	int, null
SERVERGROUPLIST	Comma-separated domain names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
CLIENTGROUPLIST	Comma-separated group names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
SITELIST	Comma-separated site names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
PARENTSERVERLIST	Comma-separated server names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
COMPUTERLIST	Comma-separated computer names by which to filter. These names can contain wildcard characters.	nvarchar(512), varchar(512), not null
IPADDRESSLIST	Comma-separated IP addresses by which to filter. These names can contain wildcard characters.	nvarchar(512), varchar(512), not null
USERLIST	Comma-separated user names by which to filter	nvarchar(512), varchar(512), not null
POLICYNAMELIST	Comma-separated policy names by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null

Table 1-78 System Report schema *(continued)*

Database Field	Comment	Data Type
EVENTSOURCELIST	Comma-separated event names by which to filter.	nvarchar(255), varchar(255), not null
SORTORDER	The column on which to sort for log views.	varchar(32), not null
SORTDIR	The sort direction. Possible values are as follows: Desc = Descending Asc = Ascending	varchar(5), not null
LIMITROWS	The number of rows to use for pagination.	int, not null
USERELATIVE	Use relative dates ('on') or absolute dates.	char(2), not null
REPORT_IDX	Not used.	int, not null
REPORTINPUTS	Special parameters if a report needs them.	nvarchar(64), varchar(64), not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	A USN-based serial number; this ID is not unique.	bigint, not null
DELETED	The deleted flag of the schema object. Possible values are as follows: 0 = Deleted 1 = Not Deleted	tinyint, not null

System State schema (SYSTEM_STATE table)

Table 1-79 describes the database schema for system state information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_SYSTEM_STATE.

Table 1-79 System State schema

Database Field Name	Comment	Data Type
CHECKSUM	The checksum of XML content.	char(32), not null
CONTENT	The XML content of the schema object.	image, not null
DELETED		tinyint, not null
ID*	The GUID of the schema object.	char(32), not null
OWNER	The GUID of the corresponding schema object.	char(32), null
TIME_STAMP	The time that the database record was modified; used to resolve the merge conflicts.	bigint, not null
TYPE	The type name of the schema object.	varchar(256), not null
USN	The update serial number; used by replication.	bigint, not null
DOMAIN_ID	The GUID of the domain that contains the state object.	char(32), null
RESERVED_INT1		int, null
RESERVED_INT2		int, null
RESERVED_BIGINT1		bigint, null
RESERVED_BIGINT2		bigint, null
RESERVED_CHAR1		char(32), null
RESERVED_CHAR2		char(32), null
RESERVED_varchar1		nvarchar(260), varchar(260), null
RESERVED_BINARY		varbinary(2000), null

Threat Report schema (THREATREPORT table)

Table 1-80 describes the database schema for threat report information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first

value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_THREATREPORT.

Table 1-80 Threat Report schema

Database Field Name	Comment	Data Type
THREATFILTER_IDX*	Primary Key.	char(32), not null
USER_ID*	The administrator GUID.	char(32), not null
FILTERNAME*	The user-specified name for this saved 'report'.	nvarchar(510), not null
STARTDATEFROM	The starting date.	datetime, not null
STARTDATETO	The ending date.	datetime, not null
RELATIVEDATETYPE	Possible values are as follows: 0 = past week 1 = past month 2 = past three months 3 = past year 4 = past 24 hours 5 = current month	int, not null
FILTER_TYPE	Possible values are as follows: 1 = Risk 2 = Proactive Threat Protection	tinyint, not null
PRODUCT	Not used.	varchar(32), not null
EVENTTYPE	The possibilities here are in the ALERTMSG table.	varchar(32), not null
ACTUALACTION	The possibilities here are in the ACTUALACTION table.	varchar(32), not null

Table 1-80 Threat Report schema *(continued)*

Database Field Name	Comment	Data Type
SOURCE	A hard-coded English lookup key. Possible values are as follows: Scheduled Scan Manual Scan Real Time Scan Heuristic Scan Console Definition downloader System Startup Scan Idle Scan Manual Quarantine	varchar(255), not null
SORTORDER	The column to use for the log view sort.	varchar(32), not null
SORTDIR	Either 'asc' or 'desc'.	varchar(5), not null
TIMEBASE	Deprecated.	varchar(32), not null
TREATCOMPRESSED	Deprecated.	varchar(32), not null
SERVERGROUPLIST	A comma-separated list of domains by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
SERVERGROUPINCLUDE	Whether to include (1) or exclude (0) the domains in the list. Always set to 1.	int, not null
CLIENTGROUPLIST	A comma-separated list of client groups by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
CLIENTGROUPINCLUDE	Whether to include (1) or exclude (0) the client groups in the list. Always set to 1.	int, not null

Table 1-80 Threat Report schema *(continued)*

Database Field Name	Comment	Data Type
PARENTSERVERLIST	A comma-separated list of management servers by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
PARENTSERVERINCLUDE	Whether to include (1) or exclude (0) the servers in the list. (Always set to 1.)	int, not null
COMPUTERLIST	A comma-separated list of computers by which to filter. These names can contain wildcard characters.	nvarchar(512), varchar(512), not null
COMPUTERINCLUDE	Whether to include (1) or exclude (0) the computers in the list. (Always set to 1.)	int, not null
IPADDRESSLIST	A comma-separated list of IP addresses by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
IPADDRESSINCLUDE	Whether to include (1) or exclude (0) the IP addresses in the list. (Always set to 1.)	int, not null
CLIENTUSERLIST	A comma-separated list of users by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
CLIENTUSERINCLUDE	Whether to include (1) or exclude (0) the users in the list. (Always set to 1.)	int, not null
HPP_APP_LIST	A comma-separated list of heuristic risks by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
THREATLIST	A comma-separated list of risks by which to filter. These names can contain wildcard characters.	nvarchar(255), varchar(255), not null
THREATINCLUDE	Whether to include (1) or exclude (0) the risks in the list. (Always set to 1.)	int, not null

Table 1-80 Threat Report schema *(continued)*

Database Field Name	Comment	Data Type
THREATTYPELIST	The possibilities here are in the VIRUSCATEGORY table. It is no longer a list but a single item.	varchar(255), not null
THREATTYPEINCLUDE	Whether to include (1) or exclude (0) the risk types in the list Always set to 1.	int, not null
THREATCATEGORY	Possible values are as follows: = -1 = Unknown >= 1 = Very low risk >= 2 = Low risk >= 3 = Moderate risk >= 4 = Severe risk >= 5 = Very Severe	varchar(255), not null
LIMITROWS	The number of rows to use for pagination.	int, not null
USERELATIVE	Use relative dates ('on') or absolute dates.	char(2), not null
REPORT_IDX	Not used.	int, not null
REPORTINPUTS	Special parameters if a report needs them.	nvarchar(255), varchar(255), not null
FROMUSERLIST	Deprecated.	nvarchar(255), varchar(255), not null
FROMUSERINCLUDE	Deprecated.	int, not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row: 0 = Not deleted 1 = Deleted	tinyint, not null

Table 1-80 Threat Report schema *(continued)*

Database Field Name	Comment	Data Type
FULL_CHARTS	An administrator-specified list of charts to include in the Comprehensive Risk Report.	varchar(255), not null
R_OS_TYPE	Indicates the operating system for the client computer. Possible values are as follows: 0000 = All non-Windows 0001=All Windows 0002=All Mac 0004= Mac OS X 10.4 0005= Mac OS X 10.5 0006= Mac OS X 10.6 0601 = Windows 7 0600 = Windows Vista 0502 = Windows 2003 and Windows XP 64-bit 0501 = Windows XP 0500 = Windows 2000 0400 = Windows NT 9999 = Windows Server 2008 -1 = No filter (all)	int, null
RISK_LEVEL	The SONAR log filter field for the risk level, which includes the following values: >= -1 = All 0 = Unknown >=1 = Low 3 = Medium 4 = High	varchar(32), not null
WEB_DOMAIN	Risk report filter for Web domain name	nvarchar(126), not null

Table 1-80 Threat Report schema *(continued)*

Database Field Name	Comment	Data Type
WEB_DOMAIN_INCLUDE	Indicates whether or not the Web domain filter is in use for this particular saved filter. This field is not currently used.	int, not null

Version schema (VERSION table)

Table 1-81 describes the database schema for version information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_VERSION.

Table 1-81 Version schema

Database Field Name	Comment	Data Type
PRODUCT*	Primary Key.	char(20), not null
VERSION	The version of Reporting.	char(10), not null
DBSCHEMA	The schema version.	int, not null
SR_NONCE	For internal usage only.	char(64), null

Virus schema (VIRUS table)

Table 1-82 describes the database schema for virus information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_VIRUS.

Table 1-82 Virus schema

Database Field Name	Comment	Data Type
VIRUSNAME_IDX*	Primary Key, Index of virus / threat.	char(32), not null
VIRUSNAME	The name of the virus / threat	nvarchar(255), varchar(255), not null
CATEGORY	The current category (as downloaded from Symantec's Web site). Values are 1 through 5, where 1 is very low and 5 is very severe. A value of -1 means unknown or not applicable. This rating applies only to viral threats.	int, not null
MAXCATEGORY	The maximum category that the virus has reached. Values are 1 through 5. A value of -1 means unknown or not applicable. This rating applies only to viral threats.	int, not null
TYPE	The threat type. Possible values are as follows: 0 = Viral 1 = Non-Viral malicious 2 = Malicious 3 = Antivirus - Heuristic 4 = Security risk 5 = Hack tool 6 = Spyware 7 = Trackware 8 = Dialer 9 = Remote access 10 = Adware 11 = Jokeware 12 = Client compliancy 13 = Generic load point 14 = Proactive Threat Scan - Heuristic 15 = Cookie	int, null

Table 1-82 Virus schema *(continued)*

Database Field Name	Comment	Data Type
TYPE2	The threat location. Possible values are as follows: 0 = Boot virus 1 = File virus 2 = Mutation virus 3 = Macro virus 4 = File virus 5 = File virus 6 = Memory virus 7 = Memory OS virus 8 = Memory mcb virus 9 = Memory highest virus 11 = Virus behavior 12 = Virus behavior 13 = Compressed file 14 = Heuristic	int, null
DISCOVERED	When Symantec first discovered the threat (as downloaded from Symantec's Web site).	datetime, not null
VID	The unique identifier for a virus that Security Response sets.	bigint, not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The time when this database record was entered or modified in the database, in milliseconds since 1970.	bigint, not null
DELETED	Deleted row: 0 = Not deleted 1 = deleted	tinyint, not null
PATTERN_IDX	Pointer to the Pattern table that protects against this threat.	char(32), not null

Table 1-82 Virus schema *(continued)*

Database Field Name	Comment	Data Type
TOP_THREAT	Possible values are as follows: 0 = Not a top threat 1 = top threat	tinyint, not null
LATEST_THREAT	0 = not a latest threat 1 = latest threat	tinyint, not null
STEALTH	Assesses how easy it is to determine if a security risk is present on a computer. Possible values are as follows: 0 = No rating 1,2 = Low 3 = Medium 4> = High -1 means not applicable. This rating applies only to non-viral threats.	int, not null
REMOVAL	Skill level that is required to remove the threat from a given computer. Possible values are as follows: 0 = No rating 1, 2 = Low 3 = Medium 4 >= High -1 means not applicable. This rating applies only to non-viral threats.	int, not null

Table 1-82 Virus schema *(continued)*

Database Field Name	Comment	Data Type
PERFORMANCE	Measures the negative impact that the presence of a security risk has on the computer's performance. Possible values are as follows: 0= No rating 1,2= Low 3= Medium 4>= High -1 means not applicable. This rating applies only to non-viral threats.	int, not null
PRIVACY	The level of privacy that is lost due to the presence of a security risk on a computer. Possible values are as follows: 0= No rating 1, 2 = Low 3 = Medium 4 >= High -1 means not applicable. This rating applies only to non-viral threats.	int, not null
DEPENDENCY	The number of dependent components that the risk installs. Possible values are as follows: 0 = No rating 1, 2 = Low 3 = Medium 4 >= High -1 means not applicable. This rating applies only to non-viral threats.	int, not null
OVERALL	An average of all the security risk ratings. This rating applies only to non-viral threats.	int, not null

Table 1-82 Virus schema *(continued)*

Database Field Name	Comment	Data Type
DYNAUBER	The main category for the risk threat. Links to the VIRUSCATEGORY table. See "Virus Category schema (VIRUSCATEGORY table)" on page 238.	int, null
DYNACAT	The subcategory for the risk threat. Links to the VIRUSCATEGORY table. See "Virus Category schema (VIRUSCATEGORY table)" on page 238.	int, null
DETECTION_TYPE	The detection type.	int, not null

Virus Category schema (VIRUSCATEGORY table)

Table 1-83 describes the database schema for virus category information.

If a Data Type cell contains one data type value, the value applies to both the MS SQL Server and the embedded database. If there are two data type values, the first value applies to MS SQL Server and the second value applies to the embedded database.

An asterisk (*) by a database field name indicates that the field acts as a Primary Key, PK_VIRUSCATEGORY.

Table 1-83 Virus Category schema

Database Field Name	Comment	Data Type
DYNAUBER	Main category ID	int, not null
DYNACAT	Sub-category ID	int, not null
LOCALE	Locale integer	int, not null
TRANSLATION	Translated name	nvarchar(510), not null

Table 1-83 Virus Category schema *(continued)*

Database Field Name	Comment	Data Type
CATEGORY_DESC	The category description, a string key that is used for a lookup. Possible values are as follows: 0 = Viral 1 = Non-Viral malicious 2 = Malicious 3 = Heuristic 4 = Not used 5 = Hack tool 6 = Spyware 7 = Trackware 8 = Dialer 9 = Remote access 10 = Adware 11 = Jokeware 12 = Client compliancy 13 = Generic load point 14 = ApplicationHeuristic 15 = Cookie	varchar(255), not null
USN	A USN-based serial number; this ID is not unique.	bigint, not null
TIME_STAMP	The server-side time (GMT) when an event is logged on the client computer.	bigint, not null
DELETED	Deleted row: 0 = Not deleted 1 = Deleted	tinyint, not null