Global Information Assurance Certification Paper

SANS GSEC Practical Assignment
Ugo Corti
Seattle SANS - March 4, 2002 – March 9, 2002
Version 1.3
Title: Consolidating NT Event Logs from servers within DMZ using DCOM and NetIQ Security Manager

Abstract

NT event consolidation is a means of collecting and centrally storing events from NT systems scattered throughout an organization. This paper was written from real world experience in deploying NetIQ's Security Manager and addresses another approach to managing servers within a DMZ by using DCOM (Distributed Component Object Model) tunneling and a dedicated Windows 2000 server that has the Security Manager CAM (Consolidator/Agent Manager) component installed.

Security Manager will increase the productivity of IT Operations and Security teams. By decreasing exposure and reaction times, while increasing protection time, Security Manager can assist organizations in achieving higher server availability.

This paper will not go into detail on how to install each component of Security Manager but will review the architecture of Security Manager and how it can be utilized to monitor and consolidate NT events from servers within a DMZ.

Introduction - NT Event Consolidation

When intruders attack a computer system they will likely leave signs of their intrusion written as events to the logs on the computers to which they've compromised. Hackers or any individual within an organization or outside can deliberately clear NT Event logs to cover their tracks. The logs themselves are vulnerable to such attacks if built in mechanisms or customized scripts are not in place to alert administrators to the status of the logs or their entries. If the decision is made to write in-house scripts that will be used to backup and clear NT logs from each server, finding a means to centrally locate these logs can be a nightmare to administer. All current versions of NT unfortunately do not have a built in mechanism to alert individuals when logs are being cleared from critical systems. Every Microsoft Windows NT machine has built in logging capabilities. There are three types of event logs: Application, Security, and System. The sort of

information that is logged in each specific event log can range from the reporting on the status of applications and NT services, auditing access attempts over the network, and possible hardware and security issues.

Administrators especially security groups within an organization would like to see events as they are occurring (i.e. NT security log cleared). When someone clears the NT Security log on an NT system event id 517 is generated. If a company does not have a security policy in place that permits clearing of event logs, event ID 517 could potentially signify that someone is being mischievous and doesn't want anyone to know about it.

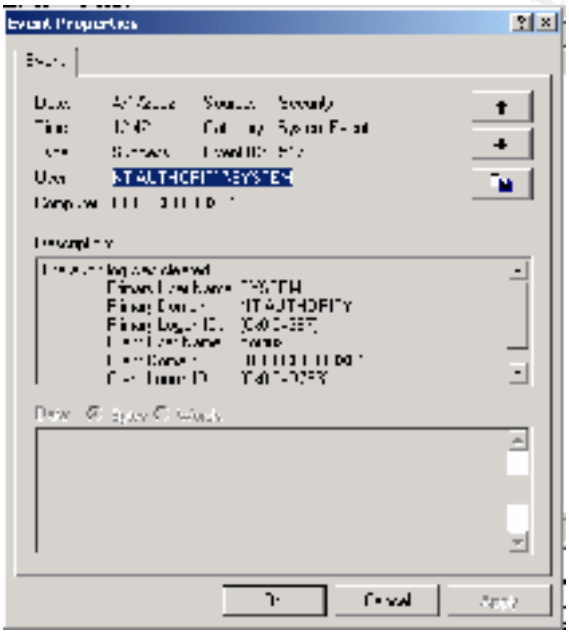

Figure 1: Security Event

Unless an administrator was viewing the security log in real-time this event could have easily been overlooked. It is very unrealistic for an organization to demand that the administrators monitor each and every NT server log for critical security breaches. This manual process can lead to human error and is very time consuming. To equip administrators of this task consolidating NT event log files to a central site is the best course of action. A central site is critical especially when organizations have decided to have fixed event log sizes. This will cause events to be overwritten when the log size has reached its maximum. Having a central site will provide the ability to review all recorded events from multiple sources in a single view.

If an intruder is successful in penetrating a computer environment particularly a Windows 2000 system running SQL within a DMZ and decides to conceal his or her tracks by clearing the NT logs; it would be possible to trace the steps of the intruder by reviewing the logs located at the central site. Of course the logs will be reviewed long after the intruder has vanished. The mechanism to which the logs are consolidated at a central site will require some sort of built in intelligence, which would allow alerts to be generated and automatically respond to key critical NT events. These specific NT events could potentially correspond to security policies established within an organization.

Having an event correlation system in place will remove the burden off of the administrator(s) and will make intrusion detection more manageable. A key principle emphasized during SANS (System Administration, Networking and Security) Institute training is that "prevention is ideal, but detection is a must" [1]. Detection must be achieved in real-time. When an intruder is detected what steps will the administrator(s) take when they are alerted via an automated generated alert? If an organization has established a process that clearly define the steps that are to be followed when a security breach occurs, the correlation system must have a mechanism to add company knowledge that provide administrators with proper procedures to handle such events. This may include steps to stop the intrusion and repair any damage done.

The correlation system must be scalable and handle large quantities of data from multiple sources, eliminate any redundancy in the data, and react to security breaches out-of-the box with little or no tuning. In the case of NetIQ's Security Manager this is accomplished via ActiveKnowledge modules. ActiveKnowledge modules are ready-to-use processing rules for monitoring and managing specific applications and environments. An example is the default Security Event Collection ActiveKnowledge module. This module is responsible to collect security related events from Windows NT and Windows 2000 security event logs. This ActiveKnowledge module also contains rules you can enable to filter some non-essential security events. The information provided by this ActiveKnowledge module allows an organization to better understand its security issues.

NetIQ's Security Manager

NetIQ Security Manager is a three-tier security product that provides the following out-of-the box functionality.

- Host-based intrusion detection
- Vulnerability assessment
- Application security
- Security event log consolidation
- Enterprise correlation

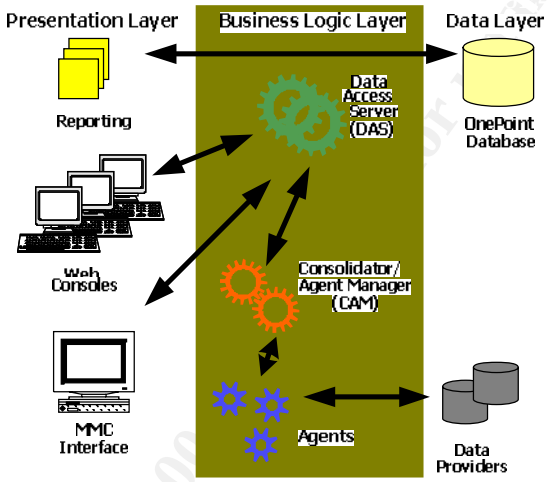

Figure 2: Three Tier-architecture

Architecture

Security Manager's architecture is designed to provide administrators with a solution that is scalable and easy to deploy. The core components of Security Manager is the **Agent**, **Consolidator/Agent Manager (CAM)**, **Database Access Server (DAS)**, and SQL **Database**. A Security Manager's Configuration Group consists of one SQL database server, one or more DCAMs (DAS/Consolidator/Agent Manager), and multiple agent machines. Security Manager provides a scalable distributed architecture because of the underlying DCOM technology that it is built on.

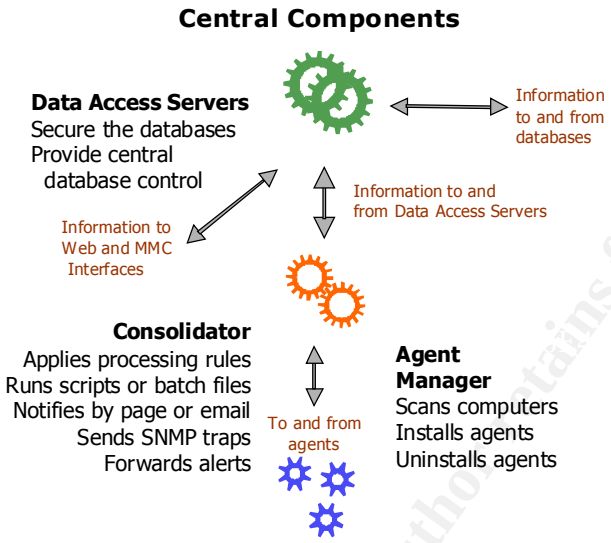

Figure 3: Security Manager Central Components

Agent

The Agent resides on each NT system as a single service that has a very small footprint. The agent currently supports the following platforms Windows NT 4, 2000, or XP. The Agent service receives a list of security rules from the Consolidator and maintains this list. If there is a matching rule for a specific event then an alert is generated and both the alert and event are forwarded to the Consolidator. If there is communication breakdown between the Agent and Consolidator alerts and events are stored locally on the Agent machine in persistent queues. Once the communication is reestablished the information stored in the persistent queues are forwarded to the Consolidator immediately. The source of incoming events are considered as Data Providers and can be anything from NT event logs, flat text files, application logs, SNMP traps, and Unix Syslogs.

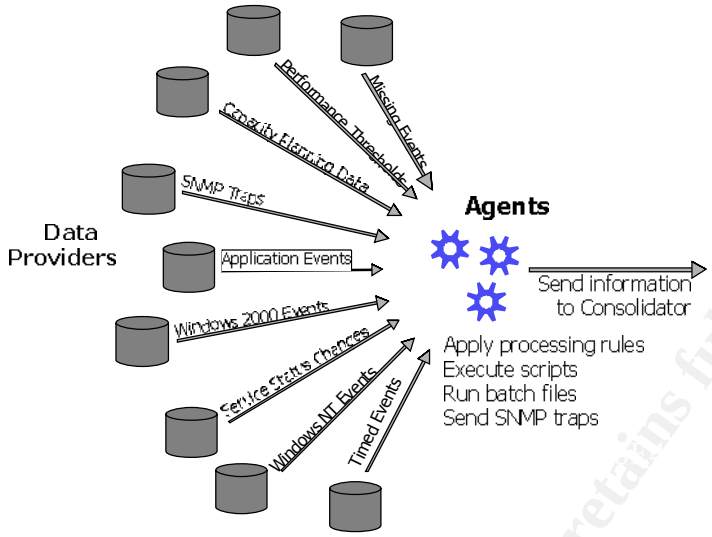

Figure 3: Data Providers

To use Security Manager through a firewall, a bi-directional (TCP and UDP) hole must be opened for either the encrypted (1270) or the unencrypted port (51515) for the Agent to communicate with its primary Consolidator. Both of these ports are configurable. A firewall rule should be created to further restrict traffic in that it should establish a source/destination hole in which only Agent traffic may pass to the Consolidator. Any other traffic coming into the defined Consolidator ports should be blocked. If a hacker was trying to cause a buffer overflow by passing invalid data to the Consolidator via port 1270 the Consolidator would drop the connection. The Consolidator has built in monitoring mechanism that checks the legitimacy of the incoming data packet for correct format. As for port 51515 an alert will be generated indicating a malformed packet has been received.

Figure 4: Alert Description

Agent deployment is accomplished through the use of RPC connections to the managed node; it is a fairly safe assumption that this functionality will be blocked by the firewall (TCP / UDP ports 135 – 139). With this restriction in place there is no communication mechanism in place for the Agent Manager to accomplish this task and therefore will require manual agent installation. Addition to the manual agent install one must manually populate the Security Manager's Computer Groups with the server names in the DMZ to allow rules to be pushed out by the Consolidator. Installing the Agent manually is a time consuming process especially in cases where there are hundreds or thousands of servers located in the DMZ.

Consolidator/Agent Manager

The Consolidator consists of two components, the Consolidator itself and the Agent Manager. Both components run under the same NT service called OnePoint. The Consolidator acts as an agent monitoring and alerting based on events that it collects. The main responsibilities of the Consolidator is collecting information from the Agents as well as sending security rule updates to the Agents. The Agent Manager has the responsibility of scanning the network for nodes (servers that potentially will have an agent installed) based on Machine Computer Rules (MCRs). The Agent Manager scans each node and queries the registry and based on what it discovers in the registry, places the computer in the appropriate Security Manager Computer Groups.

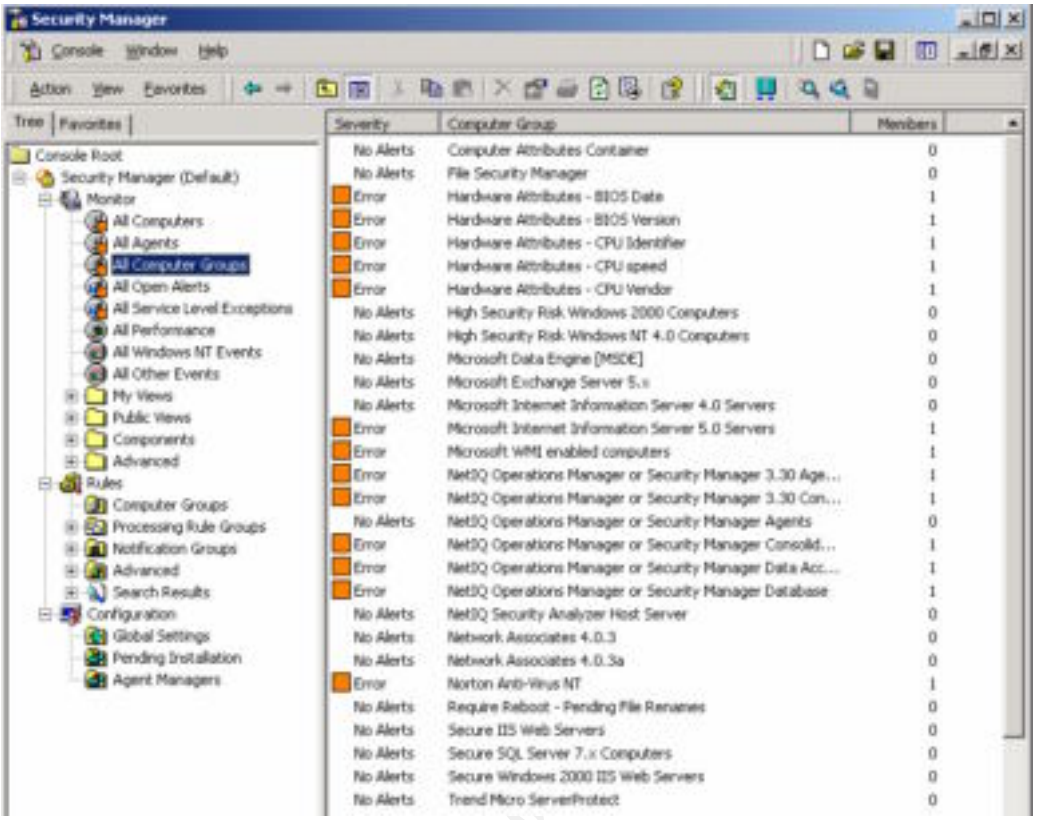

Figure 6: Security Manager's Computer Groups

The Agent Manager is also responsible to install or uninstall Agents. The Agent Manager installs an Agent by establishing a network connection to each of the node's administrative share (C$) and copies over the binary files. The Agent installation will execute scripts to create the "Mission Critical Software" registry key, the OnePoint NT service and the service will be set to "Automatic" and started.

Database Access Server

The Data Access Server (DAS) is responsible for the insertion of data within the OnePoint database residing on a SQL server. The DAS is also responsible for accessing this data via the user interface. The Consolidator's transactions to the database are controlled via the DAS. DAS is built on the Microsoft's Transaction Server (MTS) and in addition to being the access point to the database it is also the security administrator for Security Manager. User access to the database is defined within MTS (Component Services under Windows 2000) and is mapped to NT local groups residing on the DAS server.

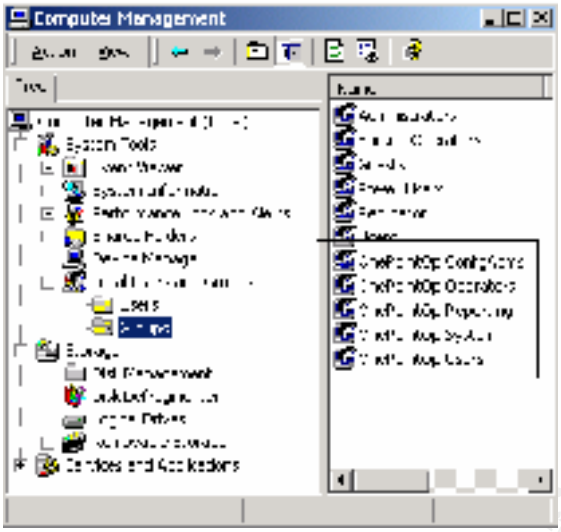

Figure 7: Computer Groups located on the DAS

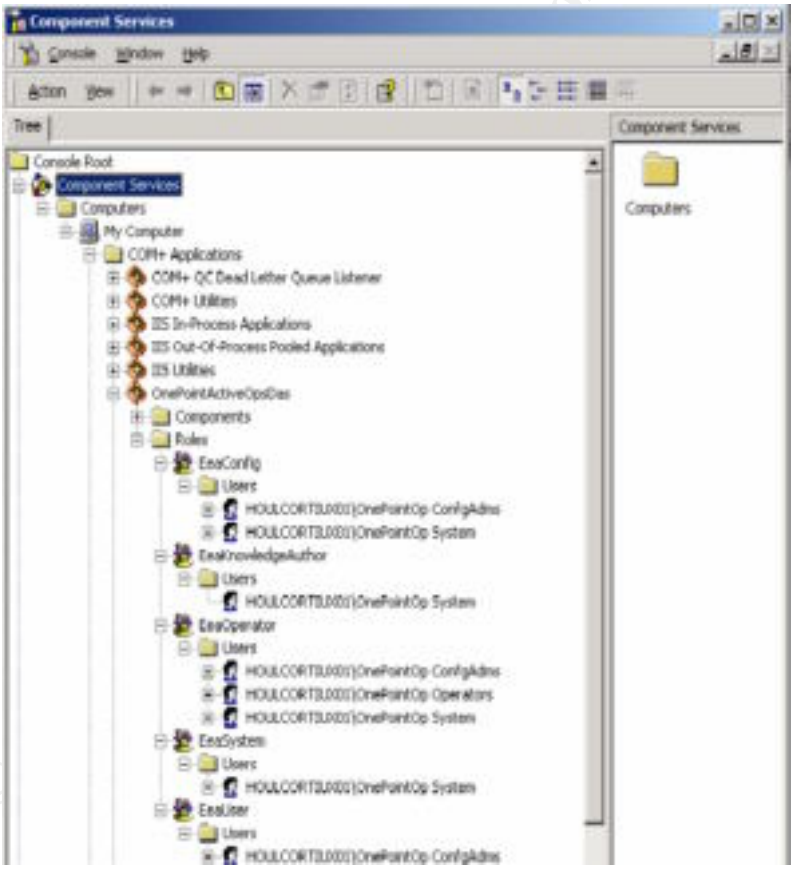

Figure 8: Component Services under Window 2000

Database

The OnePoint database for Security Manager resides on a SQL server that is running either SQL version 7.0 or 2000. The database was designed for optimal data insertion and can handle roughly 10 million events daily. This allows large amounts of data to be centrally stored quickly. There are built in database grooming jobs that are based on predefined data retention times. This allows grooming of the database for optimal performance. By utilizing Microsoft SQL server as the repository to store collected data from Security Manager agent(s) an administrator can easily create T-SQL commands to extract information from the OnePoint database.

DCOM and Security Manager

Security Manager is a distributed application that fully utilizes DCOM. DCOM is an extension of the Component Object Model (COM) technology. CIS (COM Internet Services) was first introduced in Service Pack 4 for Windows NT 4. DCOM supports the ability to communicate between COM application objects on different servers within a network by handling low-level details of network protocols. This enables Security Manager which consists of multiple processes working together to accomplish the task of collecting, alerting, and inserting NT events into a SQL repository.

DCOM has a built in dynamic port allocation feature which offers greater flexibility so that software developers are not required to configure or hard code applications to specific ports. This resolves conflicts between multiple applications attempting to use the same port(s). Unfortunately DCOM by default can utilize any TCP port between 1024 and 65535 when it dynamically selects a port for an application and therefore is firewall unfriendly.

DCOM and Firewalls

You cannot use DCOM through firewalls that do address translation (i.e. client connects to a virtual address that the firewall maps transparently to the server's actual address) because DCOM stores raw IP addresses in the packets and if the client cannot connect to the specified IP address in the packet then it will fail.

In the diagram provided you have a standard Security Manager Configuration group located within the internal network and managed servers within the DMZ. The Configuration group consists of one

Security Manager Database (NetIQ02) and one Security Manager DCAM (DAS/Consolidator/Agent Manager – NetIQ01). The Security Manager Agent was deployed to the managed servers via the dedicated CAM (NetIQ04) located within the DMZ. HOULCORTIUX01 is a Windows 2000 server running SQL 2000 and it has an Agent installed and is being managed by NetIQ04. NetIQ01 can be used to deploy Agents and manage serves within the internal network. For DCAM redundancy there is the option to add a third server (NetIQ03) to the Security Manager Configuration group.

Once the firewall security policies are configured, standard DCOM over RPC will not be available for communication between NetIQ04 and NetIQ01. The steps below will guide you in setting up CIS so that DCOM can communicate over HTTP on port 80. CIS is also commonly called "DCOM Tunneling". A firewall rule will be created to allow bi-directional TCP communication over port 80 between NetIQ04, NetIQ01 and NetIQ02. Since NetIQ04 is located within the DMZ Security Manager agents can be deployed automatically without installing them manually. This is the benefit of having a CAM within the DMZ. In turn the Agent Manager will automatically place each DMZ server in the correct Security Manager Computer Groups.

DCOM must be enabled for communications to take place between NetIQ04, NetIQ01 and NetIQ02. Most traffic will travel over the NetIQ04-NetIQ01 link. The only exception is when the CAM located on NetIQ04 is first initiated it needs to know the name of a server that has the DAS component installed. It acquires the name directly from the database server NetIQ02 by using the DASLocator COM object. Once the CAM knows the identity of the DAS server it will initiate communication with it. DCOM Tunneling is implemented in part through an ISAPI filter and therefore requires that IIS be installed on each server (NetIQ04, NetIQ01, and NetIQ02). Please note that with IIS installed the necessary security patches need to be applied before connecting the servers to the network.

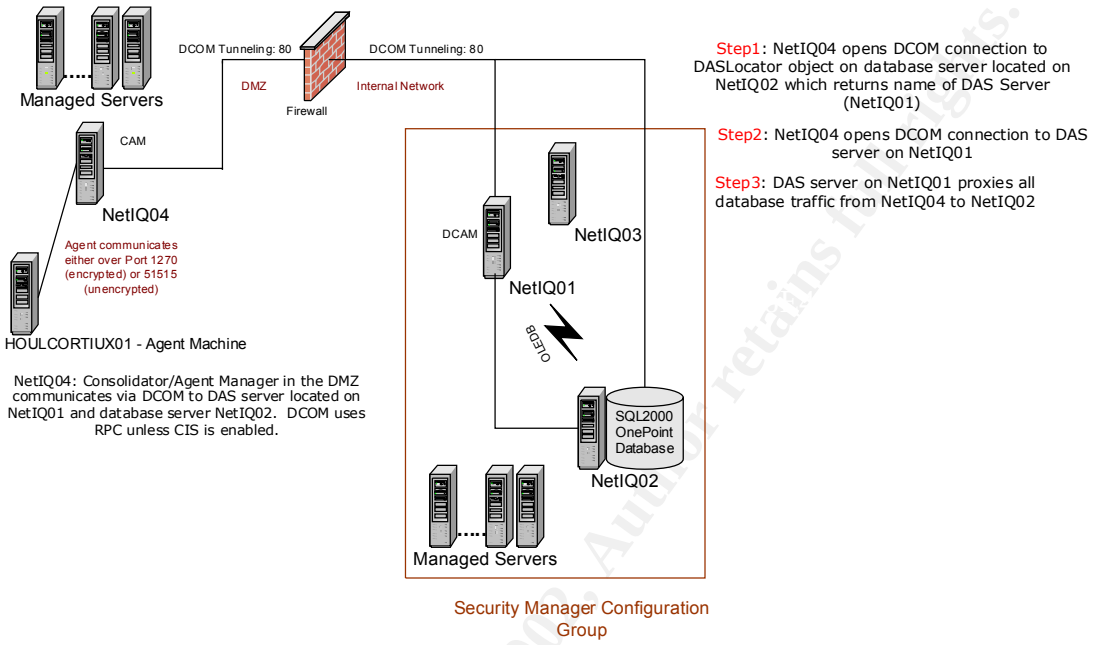

Figure 9: Security Manager - DMZ Monitoring

Steps to configure DCOM Tunneling

The following steps where taken from Microsoft Knowledge Base Article Q282261. This article is listed as a reference in the Reference section of this paper. This article describes the necessary steps to install DCOM on both NT4 and Windows 2000 systems. We are only interested in Window 2000 platform.

1. Log in to NETIQ04 with an administrator account.
2. Run **Add/Remove Programs** from the Windows Control Panel and click **Add/Remove Windows Components**.
3. Select **Networking Services** and click the **Details** button.
4. Check **COM Internet Services Proxy** and click **OK**.
5. Click **Next** at the Windows Components dialog.
6. At the Terminal Services Setup dialog, leave **Remote administration mode** checked and click **Next**.
7. You may need to provide the path to the Windows i386 directory. Click **Finish** when done.
8. From a command line or Start→Run, start the DCOM Configuration Utility "dcomcnfg.exe"

9. Switch to the **Default Protocols** tab.
10. Click the **Add** button.
11. Select **Tunneling TCP/IP** from the drop-down list and click **OK**.
12. Click the **Move Up** button several times until **Tunneling TCP/IP** moves to the top of the list.
13. Click **OK**.
14. Reboot the machine.
15. Repeat steps 2-14 for NetIQ01, NetIQ02 and NetIQ03.

If DCOM tunneling was correctly installed Security Manager processing rules would have been pushed out to the agent machine HOULCORTIUX01 via the NetIQ04 CAM. The CAM would have been able to receive these rules via the NetIQ01 DCAM since the processing rules are stored on the database server. An example of a processing rule that would have been pushed out to the agent is the rule to generate an alert when the NT Security log is cleared. If an intruder was able to gain access to server HOULCORTIUX01 and decided to delete the contents of the security log two alerts will be generated. One alert is an Information type of an alert and the other is an Error alert. These particular alerts can be viewed via the Security Manager's MMC console.

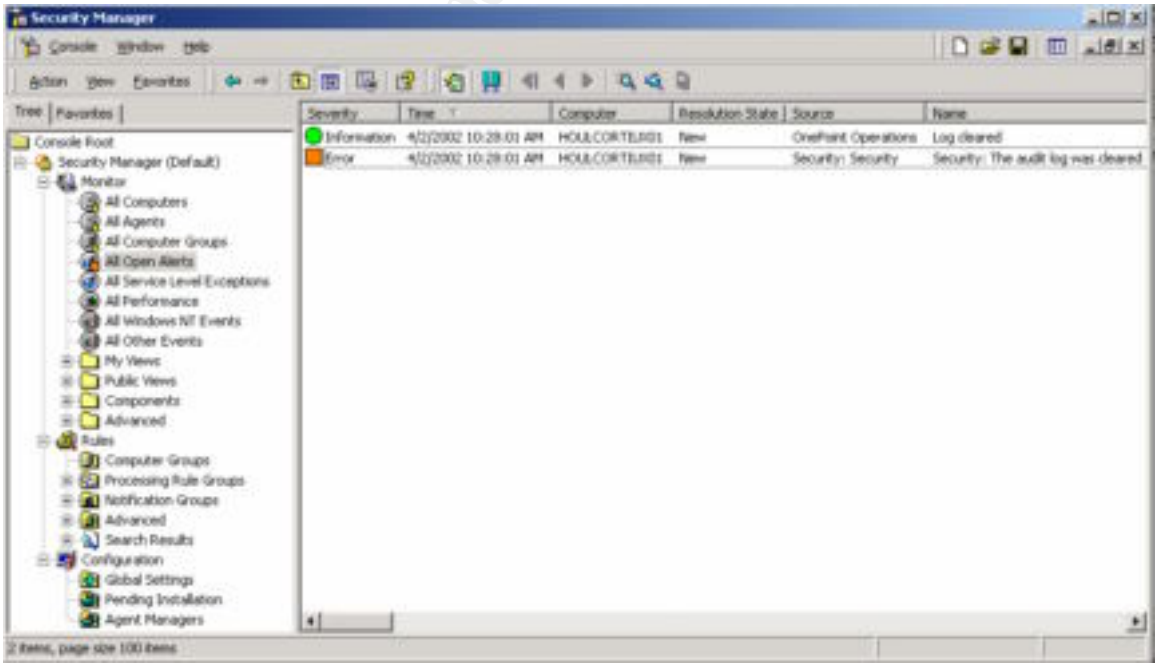

Figure 10: Security Manager MMC Console showing an Error and Information event

An administrator would be able to drill down into the Error alert and examine the detailed information. In the alert description it shows the user name (Client User Name: cortiux) that was responsible for clearing the security log.

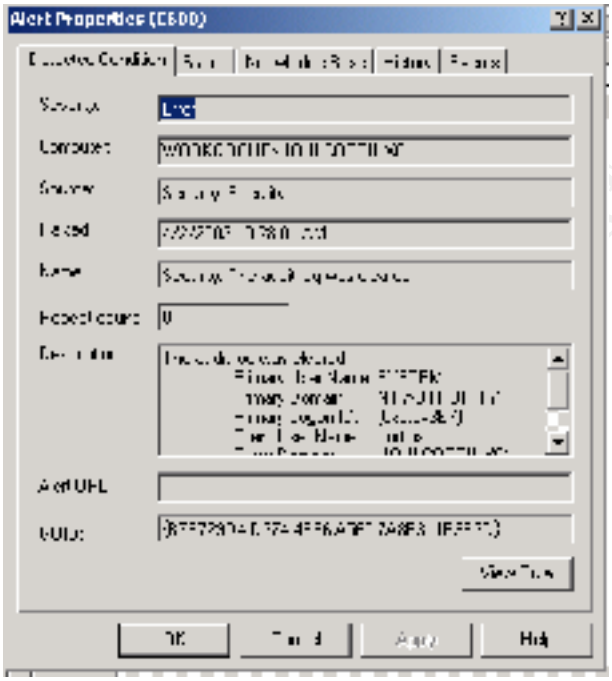

Figure 11: Alert Properties

Unknowingly to the intruder Security Manager Agent has collected all previous entries in the security log and stored them safely within the SQL database server and can be reviewed at a later time.

The administrator can easily check for additional instructions on how to handle this type of alert by viewing the contents of the Company Knowledge Base. It turns out that the clearing of the security log on this particular server is a weekly occurrence and therefore the alert can be set to resolve without any further action. In this scenario the alert was not generated do to an intruder's action but a scheduled weekly event.

Figure 12: Company Knowledge Base

Conclusion

Security event correlation is considered a huge data processing task within any organization and could possibility be well beyond the capabilities of its administrator(s) and is therefore error-prone. NetIQ Security Manager provides the means to consolidate information from various resources, reducing the amount of data by using filters, finding patterns in events, alerting and acting on these events. Security Manager captures events as they occur and makes it difficult for someone to hide their tracks by tampering with the logs.

This document provides steps in monitoring and consolidating NT events from servers within a DMZ. Security Manager provides the means for event log consolidation and stores the information into a protected, centralized Microsoft SQL repository. The administrator has the means to view alerts and events and default data that is collected via the Microsoft MMC or web console. The administrator can also use standard T-SQL commands to extract the necessary information from the SQL database since the data is not stored in proprietary format.

Security Manager was developed to fully utilize Microsoft's DCOM technology which is an extension of COM (Component Object Model).

DCOM provides developers the means to create distributed applications. Distributed applications are more scalable since the various application modules can be dispersed to various servers across your network.

References

1. The SANS Institute, "Incident Handling Foundations (SANs Security Essentials II: Network Security – Track 1)". pg 4-1.
   Many Authors

2. Establish policies and procedures for responding to intrusions
   URL: http://www.cert.org/security-improvement/practices/p044.html

3. Marc, Levy. "COM Internet Services." 23 April 1999.
   URL: http://msdn.microsoft.com/library/backgrnd/html/CIS.htm

4. Smith, Randy Franklin. "Protecting NT Security Log". Windows & .NET Magazine. July 2000.
   URL: http://www.winnetmag.com/Articles/Index.cfm?ArticleID=8785

5. DCOM Technical Overview
   URL:
   http://www.microsoft.com/technet/treeview/default.asp?url=/TechNet/prodtechnol/winntas/plan/dcomtowp.asp?frame=true

6. How To: Monitor for Unauthorized User Access in Windows 2000 (Q300958)
   URL: http://support.microsoft.com/default.aspx?scid=kb;EN-US;q300958

7. Windows NT Events
   URL: http://msdn.microsoft.com/library/default.asp?url=/library/en-us/comsrv2k/htm/cs_mmc_monitorsites_ycko.asp

8. Detecting Unauthorized Access
   URL:
   http://www.microsoft.com/windows2000/en/server/iis/default.asp?url=/windows2000/en/server/iis/htm/core/iidetsc.htm

9. Posey, Brien M. "Managing Windows 2000 Security Logs". 02 November 2000.
   URL: http://www.earthweb.com/article/0,,10456_624841,00.html

10. How To: Configure COM Internet Services (CIS) on the Server Side (Q282661)
    URL: http://support.microsoft.com/default.aspx?scid=kb;en-us;Q282261

11. Security Manager FAQ
    URL: http://www.netiq.com/products/sm/faq.asp