CA DataMinder

Network Integration Guide

Release 14.6

CA Technologies Product References

This document references the following CA Technologies products:

- CA DataMinder™

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At http://ca.com/support, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at http://ca.com/docs.

Contents

Chapter 4: Linux Server Platform Software Installation 51

Chapter 5: Configuring the NBA 65

Chapter 1: About Network Integration

CA DataMinder Network operates at the boundary between your organization and the Internet. It reconstructs complete objects, including emails, files and IM conversations, from individual data packets transmitted across your corporate network to or from the Internet.

The primary function of CA DataMinder Network is to ensure that sensitive or confidential information does not leave your corporate network. Specifically, it is designed to monitor SMTP and POP3 emails, Webmails (such as Hotmail or Yahoo!), IM conversations, FTP file transfers, files sent as attachments to Webmails or IM conversations, and documents uploaded to or downloaded from websites. Monitoring includes files and emails sent over SSL-encrypted connections.

In technical terms, CA DataMinder Network allows CA DataMinder to cover additional communication channels so that you can monitor all messages, web and file activity using a unified policy framework.

Note: This guide uses the terms CA DataMinder Network and Network Boundary Agent (NBA) interchangeably. In particular, this guide uses the term NBA to refer to both the CA DataMinder agent, and the Bivio appliance or Linux Server running CA DataMinder Network.

This section contains the following topics:

Which Versions of CA DataMinder Network are Covered by this Guide?

This guide covers version r14.5 of CA DataMinder Network.

CA DataMinder Network 14.5 is supported on Linux Servers and Bivio 7000 appliances:

- Instructions for installing CA DataMinder Network 14.5 software on Bivio 7000 appliances are included in Bivio Platform Software Installation (see page 33).
- Instructions for installing CA DataMinder Network 14.5 software on Linux servers are included in Linux Server Platform Software Installation (see page 51).

Most other sections in this guide implicitly apply to both supported platforms. If a section only applies to a specific platform, this is stated explicitly.

Passive and Active Mode

The Stream Blocking setting in the NBA console controls whether the NBA is in Active or Passive mode. The NBA must be connected in-line with the monitored network when it is configured in Active mode. The following sections describe the two modes in detail.

Passive Mode

In passive mode (that is, stream blocking is off), the NBA is supplied with a copy of the data being sent over the Internet boundary. This is normally achieved by using a data inspection port on an Ethernet switch (other names for such a port are mirror port or SPAN port). Alternatively, you can wire the NBA so that it is inline with the internet connection with the Stream Blocking setting turned off.

In passive mode, the NBA cannot actively block data because the data has already been sent to the applications either side of the Internet boundary by the time the NBA sees a copy of the data. In passive mode the NBA cannot decode SSL sessions. Policy is applied retrospectively to analyzed files and emails.

This configuration helps ensure that the NBA can have no impact on network performance, but it also means that if the data rate is higher than the NBA can accept, it will not be able to analyze all traffic. In passive mode, the NBA is a ‘best-effort’ approach.

Active (Inline) Mode

In active mode (that is, stream blocking is on), the NBA must be physically inline between the corporate LAN and the internet. In this mode, the Data Inspection ports on the NBA connect it to the LAN and the internet, and all data packets transit through the NBA.

In active mode, the NBA can actively block network events simply by not passing packets across the Internet boundary, closing network sessions, or communicating at a protocol level with applications either side of the Internet boundary. In active mode, the NBA can also be configured to decode SSL sessions and detect files and emails in them.

To allow real-time analysis of network events, the NBA must be connected via the Socket API to policy engines. This allows CA DataMinder to apply policy to data streams to determine whether they need to be blocked.

Terminology

This guide uses the following terms:

Active mode

In this mode, the NBA is physically inline between the corporate LAN and the Internet. In Active Mode, the NBA can be configured to block network packets containing files or emails and decode encrypted communications

Blocking template

See template.

Data inspection port

A port on an Ethernet switch that is used to copy network traffic from any or all data ports to a single unused port for data monitoring purposes.

In an NBA deployment, the data inspection port on the switch replicates packets to a 'receiving' data inspection port on the NBA for analysis.

Filter

A filter is a rule that determines how the NBA handles data packets meeting a specific set of criteria. The NBA supports network filters and application filters. Filters are part of the NBA policy.

FTP folder

This folder provides remote access to files on the NBA, such as the NBA policy file, nbapolicy.xml, log files, plus files and emails reassembled from data packets analyzed by the NBA.

In particular, it allows users to access NBA files from their workstation using Windows Explorer.

Management port

On the Bivio platform, this is the Ethernet port labeled 'Mgmt' on the front of the NBA Network Processor Card. Use this port to manage NBA operations and connect to CA DataMinder policy engines.

On the Linux Server platform, you can configure any conventional Ethernet port with an IP address and use it to manage NBA operations and connect to CA DataMinder policy engines.

Mirroring port

See data inspection port.

NBA console

A Web-based console that enables you to manage key NBA operations (including stopping and restarting data captures) and monitor the status of NBA components.

NBA console user

This is the user account that you use to log onto the NBA console.

On Bivio 7000 appliances or a Linux Server platform, this account is also known as the NBA webadmin user.

NBA appliance

The dedicated hardware on which the NBA resides. These are also referred to as NBA boxes. The image shows the Bivio 7000.

Note: The NBA can also be installed on a Linux Server platform which is a particular specification of a 2U rack server. Details are described under Linux Server Hardware Specification (see page 17).

nbaconfig.xml

A configuration file on the NBA that contains various parameters to control NBA operations.

For example, you can edit these parameters to specify targeted web browsing file types and disk usage thresholds.

nbapolicy.xml

A configuration file on the NBA that contains various parameters to control NBA operations.

For example, you can edit these parameters to specify targeted IP addresses, protocols and logging levels.

Notification template

See template.

Output mode

This describes how the NBA outputs captured items: to the local hard disk; via a socket connection to policy engines and/or a remote PE connector (a type of policy engine hub); to all of these; or to none.

Packet processing

The NBA's ability to process data packets passing through the NBA. You can configure NBA filters to allow, block or analyze specific data packets.

Note: A data packet on the wire is often also referred to as a frame.

Passive mode

In this mode, the NBA can read files and emails from data packets being sent over the Internet boundary. You can use the data inspection port to supply the NBA with copies of data packets, or connect the NBA in-line with the network. In Passive Mode the NBA cannot block files or emails or decode encrypted communications.

SPAN port

See data inspection port.

SSL

The Secure Sockets Layer is a cryptographic protocol that encrypts the segments of network connections above the Transport Layer, using symmetric cryptography for privacy and a keyed message authentication code for message reliability.

Stream blocking

The NBA can identify and block individual data streams passing through the NBA.

Template

This is an HTML file that contains the notification message shown to users when a Web page or file upload is blocked.

Hardware Specifications

CA DataMinder Network can run on two different types of hardware platform. It can run on a Bivio Networks Deep Packet Inspection Platform (namely, a Bivio 7000). Alternatively, it can run on a Linux server that conforms to the specifications below. The following sections describe the hardware requirements for each platform.

Note: CA DataMinder Network must be installed on hardware conforming to the specifications below. It cannot run in a virtual machine (VM).

More information:

Bivio Hardware Specification (see page 16)
Linux Server Hardware Specification (see page 17)

Bivio Hardware Specification

The Bivio 7000 uses a combination of general-purpose PowerPC processors running the Linux operating system and a powerful network processor to provide a high-speed network packet analysis platform.

The main components are the chassis, the Network Processor Card, an Applications Processor Card (optional on Bivio 7000 appliances) and a Network Interface Module.

Chassis

This contains dual redundant hot swap load-sharing power supplies and two sets of monitored fans linked to software alarms. The twin hard-disks are in a RAID-1 configuration (each store the same data, so that data is not lost if one disk fails).

Network Interface Module (NIM)

CA DataMinder Network appliances contain a custom multi-port Gigabit Ethernet (10/100/1000BASE-T) module with a configurable hardware bypass. This hardware bypass is triggered by power failure events or system failure events to guarantee that when the NBA is connected in series with the network, connectivity is assured.

- Bivio 7000 appliances contain an 8-port Ethernet module.

Network Processor Card (NPC)

The Network Processor Card combines hardware and software to deliver accelerated packet processing and load sharing of CPUs for application processing.

- On Bivio 7000 appliances, the NPC has a controller CPU and 4 Application Processor CPUs. Each Application CPU subsystem has 2.0 GB of memory for applications and packet buffering. Optional SSL decode acceleration cards can be fitted, allocating one SSL accelerator chip to each CPU.

Application Processor Card (APC)

The Application Processor Card is connected to the Network Processor Card via the NBA's high speed full duplex bus. The stack bus technology also enables linear scaling of application processing power and internal failover communication.

- The APC hardware consists of multiple high-end server-class PowerPC CPUs: On Bivio 7000 appliances, the APC is an optional feature that contains 8 CPUs, each rated at 3750 MIPS, increasing the performance available from 15,000 MIPS to 45,000 MIPS.
- Optional SSL decode acceleration cards can be fitted, allocating one SSL accelerator chip to each CPU.

- Traffic enters the CA DataMinder Network appliance through an industry standard network interface located on the NIM.
- The CA DataMinder Network appliance is a 2RU box. It is delivered with mounting brackets for 19-inch racks.

Linux Server Hardware Specification

When CA DataMinder Network runs on a Linux server, it requires a specific hardware configuration plus additional components. The hardware specification is designed to support CA DataMinder Network fully and perform the same functions as other CA DataMinder Network platforms.

Chassis

The specified 2U chassis contains dual redundant hot-swap power supplies and a pair of RAID-1 hard disks that replicate each other for redundancy.

CPUs

Dual x86 Xeon X5650 processors, each with six cores running at 2.66 GHz.

The memory requirement is at least 12 GB. For heavily loaded systems or systems that have two quad-port Napatech high speed network cards, at least 16 GB is recommended.

These processors provide enough processing power with CPU capacity allocated for network frame handling, protocol decode, and SSL decrypt and re-encrypt.

Network Interfaces

On-board gigabit network interfaces provide management capability, but additional network cards are necessary for network data capture. The required network card has been designed for high speed packet capture and generation.

Hardware Bypass

If CA DataMinder Network is connected inline with the network but is powered off or not running, network traffic cannot flow unless you fit a separate hardware bypass unit. CA DataMinder Network communicates with the bypass unit to set up the software and instruct it to either pass network traffic through the Linux server for analysis or bypass it straight through.

Specifications

CA DataMinder Network requires a Linux server from DELL or HP. Higher specification machines may work but have not been tested by CA Technologies.

Dell Server (www.dell.com)

The configuration includes:

- Dell PowerEdge R510 Chassis
- CPU1: Intel Xeon X5650, 6C, 2.66GHz, 12M Cache, 6.40GT/s, 95W TDP, Turbo, HT, DDR3-1333MHz
- CPU2: Intel Xeon X5650, 6C, 2.66GHz, 12M Cache, 6.40GT/s, 95W TDP, Turbo, HT, DDR3-1333MHz
- 16 GB Memory for 2 CPUs, DDR3, 1333 MHz (4x4GB Dual Ranked UDIMMs)
- C23 Hot-Swap 8HD - R1 for SAS 6iR/PERC 6i/H200/H700, 2 HDDs
- SAS 6/iR Internal Controller for 8x HDD Chassis, SAS/SATA Support
- 2 x 250GB, SATA, 3.5-in, 7.2K RPM Hard Drive (Hot Plug)
- 750 Watt Redundant Power Supply for 8x and 12x Hot Plug HDD Chassis
- 2x Rack Power Cords
- SATA Cable for Optical Drive for 8HDD chassis
- 16X DVD-ROM Drive SATA

HP Server (www.hp.com)

HP ProLiant DL380 G7 X5650 2P 12GB-R P410i/1GB FBWC 8 SFF 750W RPS Perf IC Server (Code: 583966-421) includes:

- 2 x Intel® Xeon® X5650 (6 core, 2.66 GHz, 12MB L3, 95W)
- 12 GB RAM
- (2) 1GbE NC382i Multifunction 2 Ports
- (1) Smart Array P410i/1GB FBWC
- Slim SATA DVD-RW
- PSU (2) 750 Watt hot plug
- 2x Code: (507750-B21)
- HP 500GB 3G SATA 7.2K rpm SFF (2.5”) HDD

High Speed Network Card

Product

Napatech Quad port Gigabit inline PCIe card NT4E-4T-INL (order code “801-0084-03-02 Inline”) and Memory module DDR2, 2GB. Order code “802-0034-00-01”.

Note: This inline card has four ports so it can monitor two independent network connections. An additional card and time sync cable are required to monitor four independent network connections.

(Optional) Napatech Time Sync cable TSC-20-A to link two cards. Order code “802-0018”.

Available from Napatech Distributor:

www.networkallies.com

1600 Osgood Street, Andover, MA 01845, USA

Tel: 978-486-0300; Fax: 978-486-0195

External Gigabit Copper Bypass Unit

Choose a bypass unit with either one or two ports for protecting one or two monitored inline network connections.

Product

- Shore Microsystems Multiport Programmable Bypass Switch (Dual Redundant Power Supplies)

  Order code: SM-2402-C-CA (Two bypass ports for monitoring two network connections).

- Shore Microsystems Programmable Bypass Switch

  Order code: SM-2400GB-CA (One bypass port for monitoring one network connection).

Available from:

Shore Microsystems

www.shoremicro.com

45 Memorial Parkway, Long Branch, NJ 07740, USA

Tel: 732-870-0800; Fax: 732-870-1912

Product

Niagara External Active Bypass Switch – Copper (External Power Supply)

Order code: 2292-TX (One bypass port for monitoring one network connection).

Available from:

Interface Masters

www.interfacemasters.com

227 Devcon Drive, San Jose, CA 95112, USA

Tel (Sales): 408.441.9341 ext. 100; Fax: 815.364.0888

Product

Silicom BSEM Bypass Switch – Copper (External Power Supply)

Order code: BSST-CE (One bypass port for monitoring one network

connection) BS1U-CET (Rack mountable with multiple port capability).

Available from:

Silicom Connectivity Solutions Inc.

http://www.silicom-usa.com

6 Forest Ave, Paramus, New Jersey 07652, USA

Tel: (201) 843-1175; Fax: (201) 843-1457

Choosing a Deployment Architecture

You must correctly deploy the NBA in terms of its network location and the data inspection port:

Important: Do not connect the NBA between your corporate firewall and the Internet. The NBA needs to listen to traffic *before* the firewall or other devices using NAT (Network Address Translation) hide local addresses from the Internet. If you deploy the NBA incorrectly, machine IP addresses may not match the actual IP addresses of the source machines.

The following sections summarize the supported deployment architectures.

More information:

Connected to Network Tap/SPAN Port: Passive Mode Only (see page 21)
Connected Inline with a Network: Active and Passive Modes (see page 23)

Connected to Network Tap/SPAN Port: Passive Mode Only

The diagram below summarizes the NBA deployment architecture when the NBA is connected to a network tap or SPAN port, and also when data is output to the local hard disk.

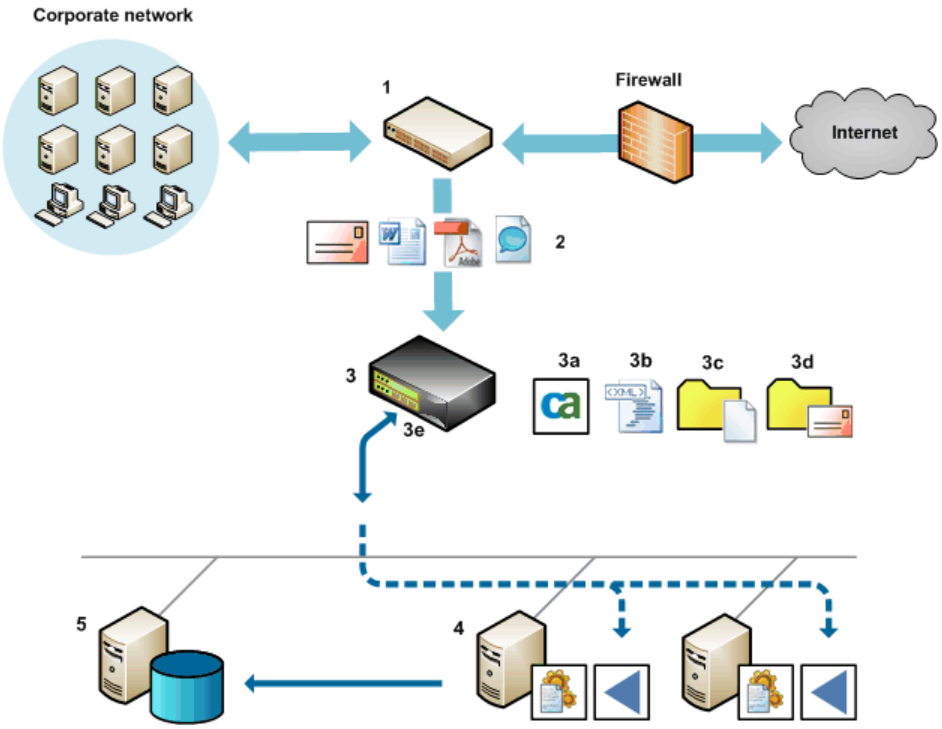
In this configuration, the NBA can only run in passive mode. In passive mode:

- The NBA cannot actively block data because the data has already been sent to the applications either side of the Internet boundary by the time the NBA sees a copy of the data.
- The NBA cannot decode SSL sessions.

Note: The passive-only configuration helps ensure that the NBA has no impact on network performance. However if the data rate is higher than the NBA can accept, the NBA is not able to analyze all traffic. If the NBA output mode is set to 'Disk' or 'Socket and Disk', the NBA is limited to a capture rate of 8 Mbyte/sec. You can connect the NBA to a network segment with traffic rates higher than this, but the sustained rate of data capture is limited to the speed that captured data files can be written to hard disk. Other data is ignored by the NBA and passes through the appliance without being analyzed or captured.

Example Architecture: Passive Mode

In this example, data packets destined for the Internet pass through switch **1**. From here, copies of these packets are replicated to the NBA, reassembled into files and emails, and stored in the NBA FTP folder. They can then be imported onto the CMS, either directly or as part of an Import Policy job.



Example architecture: Output to disk, Passive mode

1. **Switch:** Data packets passing through the switch from your corporate network to the Internet are replicated to a Data Inspection port on the NBA.
2. **Network traffic:** Replicated data packets containing emails, Webmails, files and IM conversations are passed to a *receiving* data port on the back of the appliance (typically port s0.e0 on a Bivio appliance and port 1 on a Linux server).
3. **NBA:** This hosts the Web console (**3a**) as well as the nbapolicy.xml policy file and nbaconfig.xml configuration file (**3b**).

   The NBA reassembles the incoming data packets into emails and files and stores them in \files and \mails subfolders (**3c** and **3d**) of the NBA FTP folder.

   You connect to the NBA via the management port (**3e**) to manage NBA operations and subsequently when importing captured data.

4. **Import Policy:** We recommend you run two Import Policy operations to separately import and apply policy to files and emails (imported from **3c** and **3d** respectively). In this example, both Import Policy servers are running in direct mode using local policy engines.
5. **CMS:** The resulting events are replicated up to the CMS and stored for subsequent retrieval and reviewing.

More information:

Passive Mode (see page 12)
Connecting the NBA Ports (see page 29)

Connected Inline with a Network: Active and Passive Modes

The diagram below summarizes the NBA deployment architecture when the NBA is connected inline with the monitored network, and also when data is output via a socket connection.

Using this configuration, the NBA can run in either active or passive modes.

In passive mode:

- The NBA cannot actively block network events.
- The NBA cannot decode SSL sessions.

In active mode:

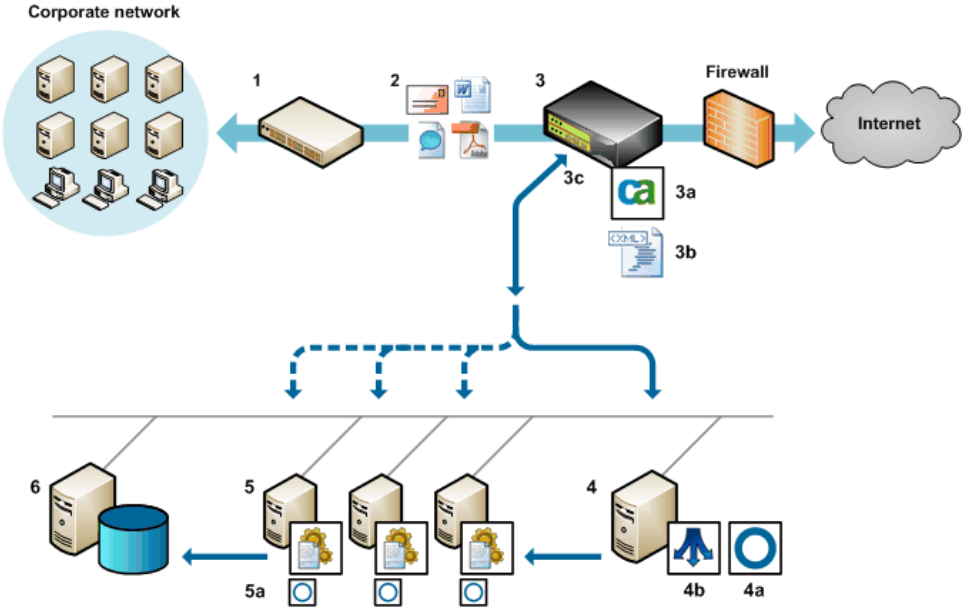
- The NBA can actively block network events. The NBA blocks events by not passing packets across the Internet boundary, closing network sessions, or communicating at a protocol level with applications either side of the Internet boundary.
- The NBA can decode SSL sessions and detect files and emails in them.

To allow real-time analysis of network events in active mode, connect the NBA via the Socket API to policy engines. This allows CA DataMinder to apply policy to data streams to determine whether they need to be blocked.

Example Architecture: Active Mode

In this example, data packets destined for the Internet pass through switch **1**. From here, these packets pass through the NBA, are reassembled into files and emails, and passed to policy engines for processing.

When policy processing is complete, any resulting 'block' or 'allow' actions are returned to the NBA. If permitted, the NBA then forwards data packets from paired ports on the back of the device, via the firewall, to their intended Internet destination.



Example architecture: Output to socket connection, Active mode

1. **Switch:** Data packets passing through the switch from your corporate network to the Internet are directed via the NBA Data Inspection ports.
2. **Network traffic:** Replicated data packets containing emails, Webmails, files and IM conversations are passed to a receiving data port on the back of the appliance (for example port s0.e0 on a Bivio appliance and port 1 on a Linux server).
3. **NBA:** This hosts the NBA console (**3a**) and the nbapolicy.xml policy file (**3b**). The NBA reassembles the incoming data packets into emails and files and passes them to policy engines (**6**) for processing.

   You connect to the NBA via the management port (**3c**) to manage NBA operations and to pass captured data to policy engines for analysis.
4. **PE hub and Socket API:** In this example, the NBA uses the Socket API (**4a**) to pass captured items from the management port (**3c**) to a policy engine hub (**4b**). But see the alternative deployment below (**5a**).

5. **Policy engines:** The hub then distributes items to policy engines for processing. The results of any policy processing are returned via the Socket API to the NBA.

   Alternatively, the NBA can pass captured items direct to policy engines, using a Socket API (**5a**) on each PE host machine.

6. **CMS:** The resulting events are replicated up to the CMS and stored for subsequent retrieval and reviewing.

More information:

Passive Mode (see page 12)
Active (Inline) Mode (see page 13)
Connecting the NBA Ports (see page 29)

Socket Connections to a Hub versus Directly to Multiple Policy Engines

The NBA can output data through a socket connection to a CA DataMinder hub or direct to policy engines. But which method is better?

Output to a hub

This method can provide more effective fault tolerance. That is, if a policy engine becomes unavailable, the hub is able to distribute load more efficiently across the remaining policy engines.

The disadvantage is that this method requires an additional network hop, with captured data passing to the hub and then onto policy engines. Also, the hub captures whole objects before forwarding them to the policy engines, so communication between the NBA and policy engines is slower.

Output to policy engines

This method is more direct and can improve network performance. However, this method provides less effective fault tolerance: if a policy engine becomes unavailable, captured items are redirected to the specified standby policy engine (if defined in nbapolicy.xml), but without any attempt to balance load.

By default, this method only uses one policy engine per NBA CPU, although you can change this by setting the 'Connections per Network Analyzer' value on the CA DataMinder Network Console's 'Policy' page.

NBA Quickstart

This section provides a quick overview of the steps needed to get the NBA up and running. Each step refers to sections of this manual where you find detailed explanations.

1. Follow the instructions in Deployment Architecture (see page 20) and NBA Ports (see page 29).

   Briefly, you must set up the hardware, and connect the power and network cables.

   - Connect power, keyboard, and video.
   - Connect the network cable for management to the 'mgt0' port (on the front of a Bivio appliance) or an ethernet port (on the motherboard of the Linux Server platform).
   - Plug the network monitor cable into the network capture card's left-most port.
   - Plug the inline network monitor cable into the network capture card's next port.

2. (Linux Server Platform only) Follow the instructions in Linux Server Platform Software and Hardware Installation (see page 51).

   Briefly, you must set up the hardware, and install the CentOS Operating System and Napatech high-speed capture card drivers.

   a. Download the CentOS operating system as an ISO image from the CA support site and burn it to a CD. Boot an unconfigured server from the Operating System CD and follow the instructions.

   b. The CentOS installer also prompts you to install the Napatech high-speed capture card. Insert the CD with Napatech driver software that you received with the inline capture card.

3. Assign an IP address to the network port that is used for NBA management.

   - (Bivio 7000 platform only) Follow the instructions in Bivio Platform Software Installation (see page 33).
   - (Linux server platform only) The Linux server uses DHCP to configure the management network port automatically. See 'Troubleshooting' for information on how to manually set an IP address (see page 226) if your site does not use DHCP.

4. Install the NBA software package.

   a. Download the NBA software from the CA support site. You can choose an RPM file or an ISO image that you burn to an installation CD.

   b. (Bivio 7000 platform only) Follow the instructions in Bivio Platform Software Installation (see page 33).

   c. (Linux server platform only) The CentOS installer prompts you to insert the NBA installation CD.

   d. (Linux server platform only) If you downloaded the RPM file, copy the RPM file to your Linux server and then enter this command:

      ```
      rpm -i <rpm_file>
      ```

      Where <rpm_file> specifies the path and name of the rpm file.

   e. Wait while the installer runs an install check at the end of the installation process.

5. Configure the NBA with the IP addresses of the CA DataMinder policy engines that process reassembled files and emails.

   In the NBA Console, open the Policy page and type the IP addresses into the Policy Analyzer IP Addresses setting. Separate IP addresses with commas. The NBA appends the default port number of 8539 automatically.

   You can also set up a Policy Engine Hub which distributes files and emails to connected Policy Engines. To install a hub, you must install the External Agent API, the Socket API, and the Remote PE Connector.

   Note: See the CA DataMinder *Platform Deployment Guide* for details about installing and configuring a Policy Engine Hub or Policy Engines with a Socket API.

6. Customize the network traffic filter (see page 77)s to only analyze 'items of interest' (files or emails).

   The easiest way to edit NBA filter policy is by using the NBA console (see page 65):

   - On Bivio 7000 appliances or the Linux Server platform, you can set most policy settings in the Filters screen and the Policy screen of the NBA console.

   Alternatively, you can edit nbapolicy.xml (see page 153) in the \config folder on the NBA.

7. Follow the instructions in Decoding SSL Communications (see page 105) to enable SSL sessions to be decoded.

   Two essential setup steps are required:

   - Use the NBA Console SSL/Master Certificates page to export the Trusted Root Certificate and then import this certificate on each client PC that connects to the internet via the NBA.
   - Enable the default SSL Decode filter on the NBA console's Filters page. This sends network traffic on ports 80 and 443 to the SSL decoder. If the NBA sees an SSL session starting on these ports, it will be intercepted.

   Note: The NBA must be in active mode to decode SSL sessions, so enable stream blocking using the NBA console.

8. (Required only If the NBA is connected inline with the monitored network) Turn active mode on.

   To turn on active mode, you must enable stream blocking. You do this in the Administration screen of the NBA console.

   For more configuration options, see Configuring the NBA (see page 65).

Chapter 2: Connecting the NBA Ports

The NBA software can be installed on the Bivio 7000 Platform or a Linux server platform. Before installing the NBA software, configure your hardware correctly with the required components and connectors.

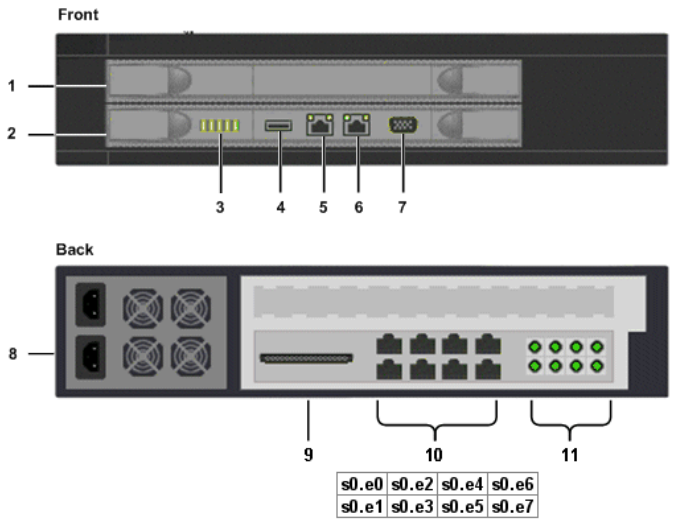
This section contains the following topics:

Connect Bivio 7000 Ports

Of the back ports, you use the s0.e0, s0.e2, s0.e4 and s0.6 data inspection ports (**8**) to receive replicated data packets from a network switch.

In passive mode, you do not use any other back ports; in active mode, you also use the s0.e1, s0.e3, s0.e5 and s0.e7 ports to forward processed packets to the internet.

These diagrams show the front and back of a Bivio 7000 appliance.



5 Management Port

You connect to this Ethernet port to manage NBA operations and when connecting to CA DataMinder Event Import machines or policy engines.

7 Console Port

This DB9 port is used when initializing the NBA.

10 Data Inspection Ports

You use these Ethernet ports to capture data on your corporate network.

The ports are paired (so.e0 with s0.e1 and so on). Each pair acts as a transparent link when inserted into a network connection. The NBA is configured such that if a failure occurs, even a loss of power, it automatically interconnects the port pairs so network traffic continues to flow

s0.e0, s0.e2, s0.e4, s0.e6

Use these ports in both active mode and passive mode. Connect these ports to your corporate network or, in passive mode, to a mirroring port on a network switch.

s0.e1, s0.e3, s0.e5, s0.e7

Use these ports in active mode. Connect these ports to the internet.

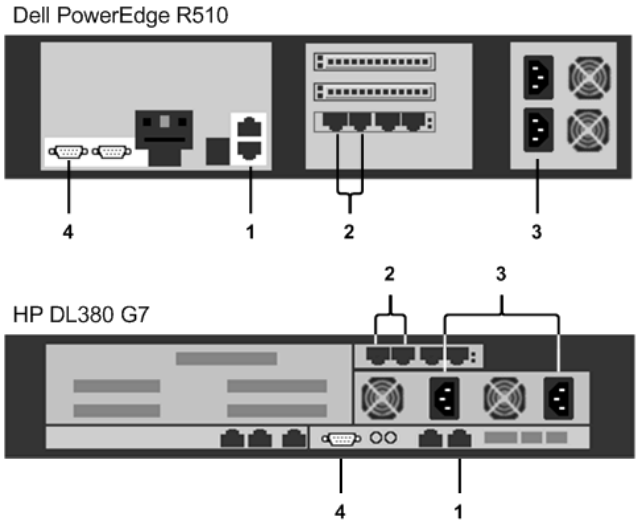
You can also use these ports in passive mode where they link directly with their paired port.

Connect Linux Server Ports

Connect the ethernet port you configured for server management to the appropriate management network switch or router.

- To support active mode, the ports on the high speed network card are connected in adjacent pairs, inline with network connections to the internet.
- Two SPAN/network mirroring ports may be connected per high speed network card. These should be connected to alternate ports.

These diagrams show the ports on a Dell PowerEdge R510 and HP DL380 G7 rack servers.



1 Device Management Ports

Use any configured Ethernet port to manage NBA operations and connect to CA DataMinder Event Import machines or policy engines.

2 Data Inspection Ports

Plug the Napatech High-Speed Network Card into PCIe slot 3 (Dell server) or slot 1 (HP server), respectively. Use the Ethernet ports on the card to capture and inspect data inline on your corporate network.

Important: The Napatech High Speed Network Card only supports Full-Duplex network connections (either 100MBit/s or 1GBit/s). If you connect Half Duplex equipment to the network card, network traffic is not detected or decoded. If you connect the card inline using Half-Duplex equipment, network traffic does not pass through the NBA.

The ports are paired (1 with 2, and 3 with 4). Each pair acts as a transparent link when inserted into a network connection.

Ports 1 and 3

Use these ports in both active mode and passive mode. Connect these ports to your corporate network or, in passive mode, to a mirroring port on a network switch.

Ports 2 and 4

Use these ports in active mode. Connect these ports to the internet.

You can also use these ports in passive mode where they link directly with their paired port.

3 Dual Power Supply Connectors

The chassis supports dual redundant hot-swap power supplies.

4 Serial Port

(Optional) Use the serial port for Bypass Unit communications.

Note: Some Bypass Units require a USB connection instead of a serial connection.

Chapter 3: Bivio Platform Software Installation

This section describes how to install NBA r14.5 software on the Bivio 7000 appliance.

After following these instructions, you will have a standard NBA system containing all the hardware and software ready for customization according to the requirements of your organization.

Installation Overview

To perform this installation, you will need a PC with suitable software installed. Actual details about PC configuration and the application user interface will vary according to the operating system and applications used. This chapter therefore only provides a general outline of the commands to use on the PC.

The installation is in two stages. First, you must check the equipment for the correct hardware configuration. Then you can install the NBA software package.

Note: The management port **must** be on a different network to the internal NBA expresslane network (which by default is 10.10.10.0). The NBA will prevent you from trying to configure both interfaces on the same network, but if existing network traffic uses 10.10.10.0 you need to reconfigure the expresslane network (see page 224). For instructions, see Troubleshooting (see page 215) in this manual or contact CA Technical Support at http://ca.com/support.

Requirements

The NBA requires the following hardware and software:

Hardware

- Bivio appliance. See the hardware specification (see page 16).
- Power cables
- RS-232 cable: DB-9 Female<->Female null-modem cable
- RJ-45 network cables
- PC with DB-9 RS232 serial connection and RJ-45 LAN connection

Software

- CA DataMinder NBA application bundle (RPM bundle)
- Terminal emulation PC application capable of connecting using SSH protocol.

Note: If you do not already have a suitable alternative on your PC, we suggest using the PuTTY application. PuTTY is available as freeware from:
`www.chiark.greenend.org.uk/~sgtatham/putty`

NBA console

When the NBA is running on a Bivio 7000 appliance, the NBA console is supported on the following browsers:

- Google Chrome
- Mozilla Firefox
- Microsoft Internet Explorer 8, 9, or 10

More information:

Distribution Packages (see page 34)
Bivio Hardware Specification (see page 16)

Distribution Packages

The CA DataMinder Network distribution packages contain the software, the CA DataMinder Network console and an SNMP agent.

To download the distribution packages:

1. Browse to the Download Center page of support.ca.com.
2. Download the software:
   - **Bivio 7000 package:** nba.B7000-14.5.3200.0.ppc.zip (6.0 MB)
   - **Cavium package:** cavium-3.00-2.ppc.zip (463 KB)

     (Bivio 7000 appliances only) If a Bivio 7000 appliance is fitted with Cavium CN1615 Security Processor cards, the NBA offloads the computationally expensive cryptographic operations to the Cavium device, freeing the Bivio CPUs from these operations and thereby maintaining high performance.

Note: Package version numbers and download sizes may differ from those listed above.

Prepare for Software Installation

Before installing the NBA software, verify that the hardware is correctly configured with the required components and connectors.

Go through the following hardware checklist:

- Note the chassis serial number in your log.

  Find this on a silver sticker in the center of the rear panel.
- Set up your PC physically close to the NBA, that is, within reach of the RS-232 cable.
- Verify that the Network Interface Module (NIM) is in the lower slot at the back of the NBA.
- On a new NBA, the only port that works is the Console (RS-232) port. Connect the Console port on the NBA to a COM port on the PC using a DB-9 RS-232 serial cable.
- On the PC you can use either Hilgraeve's HyperTerminal or a similar terminal emulation application such as PuTTY.
- Set the PC COM port to the following parameters:

  Bits Per Second: 115200

  Data Bits: 8

  Stop Bits: 1

  Parity: None

  Flow Control: None

More information:

Connect Bivio 7000 Ports (see page 29)
Connected Inline with a Network: Active and Passive Modes (see page 23)

Install the Software

When you install the NBA software on a Bivio appliance, you must perform the following tasks:

1. (Optional) Set up logging for the installation process.
2. Turn on the Bivio appliance.
3. Verify the required connections to your Bivio appliance.
4. Download the NBA distribution package from your network onto your Bivio appliance.
5. Remove any existing NBA distribution package from the NBA appliance.
6. Install the NBA software package.

   Important! For some prompts during the installation, you cannot edit your replies using the Backspace or Delete keys. The prompts always take the first character you enter as your response. Enter values with care to avoid having to repeat the process!
7. (Optional) Verify that the Cavium driver is up to date if you want to use the SSL decoding functionality.

More information:

Log the Installation Process (see page 36)
Turn On the Bivio Appliance (see page 37)
Verify the Required Cnnections (see page 41)
Download the NBA Software Package onto the Bivio Appliance (see page 42)
Remove an Existing NBA Software Package (see page 44)
Install the NBA Software Package (see page 44)
Verify that the Cavium Driver is up to Date (see page 45)

Log the Installation Process

We recommend that you log all NBA installation communications. This log is useful for checking that the installation is correct. You may be asked for it if you need to contact CA Technical Support.

To log the installation process with:

- HyperTerminal, choose Transfer > Capture Text.
- PuTTY, choose, Session > Logging > All session output

Start logging now. See the following sections for detailed installation instructions.

Turn On the Bivio Appliance

After powering up, the NBA outputs text on the Console port. When the ROM password prompt appears, type N, or let the prompt time out; you do not need a ROM password.

To finalize the installation, you have to configure the following settings:

MCCP subnet

Specifies the internal network address for the system. You can use the default unless it matches a subnet already in use by the system elsewhere. If the address is taken, change the default.

IP address of the Management Port and Network Mask

Defines the management port network address. This address gives you access to the CA DataMinder NBA console and FTP server.

Configure an IPv4 address during installation. On Bivio 7000 appliances and the Linux Server platform you can access the NBA using IPv6 addressing after installation.

Do one of the following:

- If you have not powered up the system before, finalize the installation (First Time) (see page 37)
- If you have powered up the system before, and you have previously logged in as 'admin', finalize the installation (As Admin) (see page 40)

More information:

Verify the Required Cnnections (see page 41)

Finalize the Installation (First Time)

If you have **not** powered up the system before, the system prompts you to log in as 'admin', and leads you through the initialization steps. The most important step is entering the IP address of the Management Network Port.

To finalize the installation

1. Log in as 'admin'. There is no password configured at this stage, and it does not ask you for one:

```
================================================================================
==
  Please log in as user 'admin' to perform the initial setup routine
================================================================================
==
CPU-X login: admin
```

2. Define the internal network address to use for this system. You can use the default unless it matches a subnet already in use by the system elsewhere.

   Note: If the address is taken, change the default. Changing the default entails that you also update the routing script that controls network connectivity to Policy Engines. You update the internal network routing (see page 48) after the NBA software installation.

```
To finalize your installation, please answer the following questions. When you
are done, the machine will be rebooted and then ready to use.

The MCCP subnet needs to be a class C subnet that is only used internally. This
cannot conflict with any traffic forwarded through this device, and also cannot
conflict with any interfaces of this device.

------------------------------------------------------------------------
MCCP subnet address    ? [10.10.10.0]
------------------------------------------------------------------------
Please verify your settings:

  MCCP subnet address  : 10.10.10.0
------------------------------------------------------------------------
Is this correct? (y/n) [n] y
```

   The internal network address is configured.

3. Define the management port network address. This address gives you access to the CA DataMinder NBA software package on the FTP server:

```
IP address for management port?  [10.0.12.17]
Network mask for management port?  [255.255.0.0]
Broadcast address for management port?  [10.0.255.255]
------------------------------------------------------------------------
Please verify your settings:

IP address for management port: 10.0.12.17
Network mask for management port: 255.255.0.0
Broadcast address for management port: 10.0.255.255
------------------------------------------------------------------------
Is this correct? (y/n) [n] y
```

4. Enter 'n'. You do not specify a default gateway.

```
You may enter a default gateway now for your management port, or choose N
and configure all routes later using 'configure staticroutes'.
------------------------------------------------------------------------
Do you want to specify a default gateway now? (y/n) [n] n
```

5. Enter a hostname for your NBA installation. The actual name is not important, but it must follow the format of hostname.domain.com (for example, nba17.ca.com). When asked to confirm, enter 'y'.

```
Hostname (incl. domain name)? [] nba17.ca.com
------------------------------------------------------------------------
Please verify your settings:

 Hostname (incl. domain name): nba17.ca.com
------------------------------------------------------------------------
Is this correct? (y/n) [n] y
```

6. Enter 'n'. You do not specify a name server.

```
Do you want to specify a name server now? (y/n) [n] n
```

7. Enter the default passwords when prompted: The passwords are 'root' for the root user and 'admin' for the admin user. You can change these later if required.

```
Next, please provide passwords for the standard system users:

root-> Privileged account for use by experts, in emergencies only
admin-> System administrator
------------------------------------------------------------------------
```

   You will see a warning about password strength.

   Enter the same password again at the "Re-enter..." prompt to override the warning:

```
Password for user: root
New password:
Bad password: too short
Re-enter new password:

Password for user: admin
New password:
Bad password: too simple
Re-enter new password:
```

8. Set the timezone at the next prompt. You can change this later if required. When asked to confirm, type 'y'.

```
|          Please select your timezone from the list below          |
------------------------------------------------------------------------
| 00 - Hawaii                    | 01 - Alaska                     |
| 02 - Pacific Time              | 03 - Arizona                    |
| 04 - Mountain Time             | 05 - Central Time               |
| 06 - Mexico City               | 07 - Eastern Time               |
| 08 - Atlantic Time             | 09 - Newfoundland               |
| 10 - Brazil/East               | 11 - Buenos Aires               |
```

```
| 12 - UTC                        | 13 - Western Europe            |
| 14 - Central Europe             | 15 - Eastern Europe            |
| 16 - Moscow                     | 17 - Calcutta                  |
| 18 - China                      | 19 - Singapore                 |
| 20 - Western Australia          | 21 - Taipei                    |
| 22 - Tokyo                      | 23 - Seoul                     |
| 24 - Eastern Australia          |                                |
-------------------------------------------------------------------
| nn -> select  |  M -> wider selection (tzselect)   |  C -> Cancel|
-------------------------------------------------------------------
Your selection > 12
Setting timezone to UTC
Setting time/date to 05/27/2008 10:50.
Is this correct? (y/n) [n] y
-------------------------------------------------------------------
Tue May 27 10:50:00 UTC 2008
```

The system restarts.

9. Wait for the system to restart. You do not need to set a ROM password. Do not press escape at the ROM Menu prompt. Wait until the ‘CPU-X login:’ prompt appears.

   When you have completed these tasks, continue with the NBA software installation procedure.

Finalize the Installation (As Admin)

If you **have** powered up the system before, and you have previously logged in as ‘admin’, note that the system does not prompt you to log in as ‘admin’.

To finalize the installation

1. Log in as ‘admin’ now.

```
System Name: system.ca.com

CPU-X login: admin
Password:
Last login: Tue Jan 30 08:01:00 on /dev/tts/0

You are now logged in to CPU-X of system.ca.com

BiviOS Version 3.2.5.9 (Build 200803051300) CLI

Loading ..............................
```

2. Enter the configuration commands as shown in the following example:

```
system[admin:]> configure interfaces
system[admin:interfaces]> set interface mgt0 address 10.0.12.17 netmask
255.255.0.0
Conflict checking interface address...Done!
system[admin:interfaces]> configure system

Do you want to perform a "commit boot" on your changes? (y/n) [y] y
Interface configuration committed.
system[admin:system]> set mccp subnet address 10.10.10.0
* To change the MCCP subnet to 10.10.10.0, the system must be rebooted.

Do you want to commit your changes and reboot now? (y/n) [y] y
```

   The system restarts.

3. Wait for the system to restart. You do not need to set a ROM password. Do not press escape at the ROM Menu prompt. Wait until the 'CPU-X login:' prompt appears.

   When you have completed these tasks, continue with the NBA software installation procedure.

Verify the Required Cnnections

Before you install the NBA distribution package:

- Your Bivio appliance must be connected to a network using the Ethernet management port.
- Your network must provide access to the computer containing the required NBA distribution package.

If the current NBA management port network address configuration does not permit this connection, you must reconfigure it.

To reconfigure the NBA management port

1. Connect the computer and Bivio appliance via an Ethernet switch or hub.
2. Configure your computer IP address and netmask to appropriate settings compatible with the NBA management port settings.
3. Use your web browser to connect to the NBA management port address.
   a. Log in as 'webadmin' and use the Interfaces screen.
   b. Change the IP address and netmask of the management port.

Perform the remaining installation instructions using the same serial connection that you used to configure the OS.

Alternatively, you can make a network connection with a PC terminal emulator connected to the NBA management port using an SSH connection. If you do not have a suitable application (capable of connecting using the SSH protocol) already installed on your computer , we recommend the PuTTY application. This is available free of charge from:

`www.chiark.greenend.org.uk/~sgtatham/putty`

Download the NBA Software Package onto the Bivio Appliance

Before you can install CA DataMinder Network, you must download the NBA software package from your network onto your Bivio appliance.

To download the NBA software package onto your Bivio appliance

1. Login as root:

   ```
   [nba17] CPU-X login: root
   ```

2. You can save the NBA software package anywhere on the NBA system, but for consistency we recommend the following directory:

   ```
   /bivio/shared/home/install
   ```

   If this directory does not already exist, create it with:

   ```
   mkdir -m 777 /bivio/shared/home/install
   ```

3. Go to the install directory.

   ```
   cd /bivio/shared/home/install
   ```

4. Download the required version of the NBA distribution package (see page 34). Use *one* of these methods:

   - Use FTP from the NBA command prompt. You must switch to binary mode before you get the software package.
   - Use an SCP-compatible application from a command prompt on the Windows host.

     The application must be compatible with the SCP protocol. For example, use the PuTTY application pscp.

   See the following sections for details about these download methods.

5. Download the package containing the Cavium driver.

   Note: If a Bivio 7000 appliance is fitted with Cavium CN1615 Security Processor cards, the NBA offloads the computationally expensive cryptographic operations to the Cavium device, freeing the Bivio CPUs from these operations and thereby maintaining high performance.

More information:

Using FTP from the NBA Command Prompt (see page 43)
Using pscp from a Windows Host Command Prompt (see page 43)
Distribution Packages (see page 34)
Verify that the Cavium Driver is up to Date (see page 45)

Using FTP from the NBA Command Prompt

To download the NBA distribution package

1. Continue from the CPU-X login in the previous section:

   ```
   ftp 10.0.1.96
   ```

2. Enter any required name and password.
3. List the available files and verify that the NBA RPM file exists:
   ```
   ls
   ```
4. Set the file transfer mode to binary and get the appropriate .rpm file. For example:

   ```
   binary
   ```

   ```
   get nba.B7000-14.5.3200.0-1.ppc.rpm
   ```

   Wait for the download to complete.
5. Quit the FTP client:

   ```
   quit
   ```

Using pscp from a Windows Host Command Prompt

To download the NBA distribution package

1. From a command prompt, navigate to the folder containing the pscp application. For example:

   ```
   cd "\Program Files\PuTTY"
   ```

2. Copy the appropriate .rpm file using the following command syntax on one line:

   ```
   .\pscp.exe -l root -pw root_password rpm_file
   nba_address:/bivio/shared/home/install
   ```

   Where root_password, rpm_file, and nba_address are as above. For example:

   ```
   .\pscp.exe -l root -pw root
     C:\Users\srimmel\Downloads\nba.B7000-14.5.3200.0-1.ppc.rpm
     10.201.12.10:/bivio/shared/home/install
   ```

Remove an Existing NBA Software Package

Before installing a new NBA software package, you may first need to uninstall any existing NBA package from the NBA appliance. This is necessary if the previously installed version does not support upgrades.

To remove an existing NBA package

Do the following to uninstall the NBA software from the system.

- Execute the following command from an NBA CPU-X root login:

  ```
  rpm -e nba
  ```

- If the above command generates 'dependency errors', use:

  ```
  rpm -e --nodeps nba
  ```

  Note the two hyphens before the 'nodeps' keyword.

Install the NBA Software Package

To install the NBA software

1. Execute the following command from an NBA CPU-X login in the directory where the .rpm is present. Note the two hyphens before the 'nodeps' keyword.

   ```
   rpm -U --nodeps <rpm file>
   ```

   This command installs the NBA software, configures it, and starts the application.

   Note: This command may fail if there is a previously installed NBA package that does not support upgrade. If so, remove the old package (see the previous section) before repeating this command.

2. Carefully monitor the output from this installation command.

   You must refer any errors back to CA Technical Support.

3. Log on as root administrator to be able to reboot the NBA:

   ```
   [ta] CPU-X:/bivio/shared/home/install # su admin
   ```

   This generates output similar to the following:

   ```
   BiviOS Version 3.2.5.9 (Build 200803051300) CLI

   *** There are [ 1 ] unacknowledged alarms.
       Use 'show alarm unacknowledged' to display.

   Loading ..............................
   ```

4. Reboot the NBA from your NBA CPU-X root login to complete the installation.

   ```
   ta[admin:]> reboot system
   WARNING: This will reboot the system and traffic will be impacted.
   Are you sure you want to do this? (y/n) [n] y
   ```

   Note: Do not use the NBA console until after this reboot.

Verify that the Cavium Driver is up to Date

If a Bivio 7000 appliance is fitted with Cavium CN1615 Security Processor cards, the NBA offloads the computationally expensive cryptographic operations to the Cavium device, freeing the Bivio CPUs from these operations and thereby maintaining high performance.

Therefore, if you want to use the SSL decoding functionality, verify that your Bivio appliance is using the latest Cavium driver.

To update the Cavium driver

1. Check which version of the Cavium driver is currently installed. To do this, execute the following command from an NBA CPU-X login:

   ```
   rpm -qa | grep cavium
   ```

   The output shows which version of the Cavium rpm is currently installed (for example, cavium-3.00-2)

2. If the installed version is older than the version on the CA Support Online download site ("Cavium 3.00 - 2"), delete the existing version.

   To delete the existing version, execute the following command:

   ```
   rpm -e cavium
   ```

3. Install the new version of the Cavium rpm. To do this, execute the following command in the directory where the .rpm is present:

   ```
   rpm -ivh cavium-3.00-2.ppc.rpm
   ```

   Note: The version number shown here is an example. Replace this example with the appropriate version number.

4. Reboot the Bivio appliance. To do this, execute the following command:

   ```
   nrsp reboot system
   ```

Verify the NBA Installation

At the end of the NBA package installation, the installer runs the "installcheck" script automatically. This script verifies that the basic installation steps have completed successfully. The script does not perform an extensive test of the entire system.

You can run the script manually at any time.

To check the installation

1. Execute the following command from an NBA CPU-X root login:

   ```
   ta] CPU-X:~ # installcheck
   ```

   On Bivio 7000 appliances, this generates output similar to this:

   ```
   ======================================================================
   =           NBA Installation Checker    Version 14.5.3200.0
   ======================================================================

   OS = 5.0.7-3 (Build 201008021214)                                   OK
   NBA version = 14.5.3200.0                                           OK
   NBA product branding                                                OK
   Ethernet port failover                                              OK
   NBA services                                                        OK
   ----------------------------------------------------------------------
   Overall result of all installation checks (All passed)              OK
   ----------------------------------------------------------------------
   ```

2. Manually check that the reported ‘NBA version’ is the required one.
3. If you see any Error results in the above check, wait briefly, then run the check again.

   Such errors are caused by a delay in some components starting after a reboot. If the error is still reported after five minutes, there is a problem with the installation.

Assign IP Addresses

Assign an IP Address to the NBA Management Port (Bivio Platforms)

To assign an IP address

1. Connect to the Management port on the front of the NBA.
2. Launch the NBA console for the first time. To do this, browse to its default IP address:
   `https://192.168.1.1`

   The NBA console displays automatically.

3. Log on using the default username and password for the NBA console user: 'webadmin' and 'webadmin' respectively.

   Note: You specify more secure passwords for the admin (see page 68) and for the root user (see page 68) later.

4. Go to the administration screen in the console, and go to the Interfaces screen.

5. Edit the following settings in the IP Address Configuration section:

   Management Port Address

   Defines an internal IP address that your administrators can use to access the NBA. This IP address must be on the same subnet as your CA DataMinder policy engines or it must have access to that subnet via a router or switch.

   Netmask

   Defines the subnet mask for the network segment on which the NBA is located.

   Example: If machines on your corporate network (including any policy engines) are on a 10.0.12.* subnet, configure the management port IP address to be on the same subnet as the policy engines, such as 10.0.12.1.

6. Click Apply to confirm the address change.

   As soon as you confirm the address change, you lose the existing connection to the NBA console.

7. Verify that the address change was successful. To do this, reconnect to the NBA using the address you supplied in the IP Address Configuration section.

   The NBA console displays.

More information:

Set a Password for the NBA Console User (see page 68)
Set a Password for the NBA Root User (see page 68)

Update the Internal Network Routing

If you have changed the internal network address (the MCCP subnet address) from its default 10.10.10.0 to another address, you must also specify the new address in the routing script that controls network connectivity to Policy Engines.

Follow these steps:

1. Edit the file /etc/bivio/init.d/apcnatd.
2. Change the following line to include the new address:

   ```
   CPUX_NR0_IP=10.10.10.1
   ```

3. Reboot the system

   ```
   nrsp reboot system
   ```

   The new IP address takes effect.

Example

If you changed the MCCP network address to 192.168.0.0, then edit the apcnatd script as follows:

```
CPUX_NR0_IP=192.168.0.1
```

Configuring the Bivio 7000 for Routed Mode

(Optional)

You can improve performance of the Bivio 7000 by using one network port for NBA management and a second for communicating with policy analyzers.

The default network configuration for the Bivio 7000 uses the Management network port at the front of the device for both NBA console and Policy Analyzer connectivity. Using this port makes configuring the device on the network much easier.

Routed mode uses separate network ports and IP subnets for the management connection and policy analyzer connection. This configuration can result in a 15 percent increase in the volume of network traffic that Policy Engines can analyze.

To activate routed mode:

1. Log in to the Bivio 7000 shell prompt using SSH and edit this file:
   `/etc/bivio/bvcig/conf.d/orch_bv_config.xml`
2. Change the line
   `<CIG:Interface name="s0.e7" type="transparent">s0.e6</CIG:Interface>`

   to read
   `<CIG:Interface name="s0.e7" type="routed"></CIG:Interface>`
3. Set the s0.e7 IP address (on an IPv4 subnet different from the management IP address) using this command:
   `ifconfig s0.e7 inet` *`IPAddress`* `netmask` *`Netmask`*
4. Plug the policy analyzer network connection into s0.e7 at the rear of the Bivio 7000.

Chapter 4: Linux Server Platform Software Installation

This section describes how to install NBA software on the Linux server platform.

After following these instructions, you have a standard NBA system containing all the required hardware and software and which you can customize to meet the needs of your organization.

Installation Overview

To perform this installation, you need a monitor and keyboard compatible with the connections provided on your Linux server platform.

After the initial installation, you can perform software updates using the same peripherals or using a PC with suitable software installed. The actual PC configuration and application user interface varies according to the operating system and applications used. This chapter therefore only provides a general outline of the commands to use on the PC.

Installation Process

The installation process has two stages.

1. Verify that the equipment has the correct hardware configuration.
2. Install the NBA software package.

How Long Does the Installation Take?

The longest step in the overall installation is the CentOS installation. This step typically completes in about 20 minutes.

Requirements

The NBA requires the following hardware and software:

Hardware

- Linux server. See the required hardware specification (see page 17).
- Power cables
- RJ-45 network cables

- Monitor with a suitable cable for connection to the server.
- Keyboard with a suitable cable for connection to the server.
- A PC with network access to the NBA management port. You use this PC to perform updates without a monitor and keyboard directly connected to the server.
- (Optional) Napatech High Speed Network Card. This card only supports Full-Duplex hubs and switches, either 100MBit/s or 1GBit/s.

Software

- CentOS (CA DataMinder NBA compatible) installation CD.
- Napatech (CA DataMinder NBA compatible) installation CD (if you use a Napatech card).
- CA DataMinder NBA installation CD (or application bundle, RPM bundle, for updates).
- A terminal emulation PC application that can connect using the SSH protocol (used for performing updates without a monitor and keyboard directly connected to the server).

  If you do not already have a suitable alternative on your PC, we recommend using the PuTTY application. PuTTY is available as freeware from:
  `www.chiark.greenend.org.uk/~sgtatham/putty`

NBA console

When the NBA is running on the Linux Server Platform the NBA console is supported on the following browsers:

- Google Chrome
- Mozilla Firefox
- Microsoft Internet Explorer 8, 9, or 10

More information:

Distribution Packages (see page 53)
Linux Server Hardware Specification (see page 17)

Distribution Packages

The CA DataMinder Network distribution packages contain the software needed to deploy CA DataMinder Network on a Linux Server Platform. The package includes the installation software, the Network console and an SNMP agent.

To locate these packages:

1. Browse to the Download Center page of support.ca.com.
2. Search for 'CA DataMinder Network'.

   The search returns the various product components.
3. Download the following components:
   - **CD - CentOS package:** CD*xxxxxxxxxx*.iso (411 MB)
   - **CD - CA DLP Network r14.5:** CD*xxxxxxxxxx*.iso (2.6 MB)

Note: Package version numbers and download sizes may differ from those listed above.

Install Dual Napatech Cards

(Only required if you use two Napatech cards.)

If you use two Napatech cards, you can monitor four networks. Before you install the NBA software, verify that the Napatech cards are correctly installed and synchronized.

To install and synchronize dual Napatech cards:

1. Install the Napatech NT4E-INL cards into the PCI-Express slots in the Linux Server Platform.

   Take anti-static precautions and adhere to the hardware installation instructions supplied by the hardware manufacturers.
2. If you use two Napatech cards, synchronize them using a Napatech time synchronization cable.

   Depending on the hardware layout, you may need to attach the time synchronization cable before installing one or both of the cards into the PCI-Express slots:

   a. Connect the time synchronization cable between the "Int1" MCX connectors (J16 - near the heat-sink and fan) on the Napatech cards.

   b. Verify that you pushed the connectors together securely without damaging the connectors or kinking the cable.

Important: If you connect a network inline with the Napatech ports before installing the NBA software, this can disrupt network traffic.

More information:

(Optional) Configure External Bypass Unit (see page 58)
Contact Your Napatech Distributor (see page 229)

Install the Software (Linux)

When you install the NBA software on hardware that has not previously been used as a CA DataMinder Network Appliance, you must perform the following tasks:

1. Install the CentOS operating system.
2. Verify the network port configuration.
3. Install the Napatech high-speed network card drivers.
4. Install the NBA software package.
5. Assign an IP address to the NBA management port.
6. (Optional) Configure the external bypass unit.
7. Verify the installation.

Note: When you upgrade an existing CA DataMinder Network Appliance, you may only need to upgrade (see page 62) some of these packages. See the NBA release notes for details of package compatibility.

More information:

Install CentOS (see page 55)
Verify the Network Port Configuration (see page 55)
Install the Napatech High-Speed Network Card Drivers (see page 57)
Install the NBA Software Package (see page 57)
Assign an IP Address to the NBA Management Port (Linux Server Platform) (see page 58)
(Optional) Configure External Bypass Unit (see page 58)
Verify the NBA Installation (see page 61)

Install CentOS

In this step, you install the CentOS operating system, and then log on to the Linux server to start the NBA package installer script.

The CentOS installation typically completes in about 20 minutes without any further user input.

To install CentOS:

1. Power on the Linux Server Platform
2. Insert the NBA "CentOS package" CD or DVD into the CD/DVD drive.
3. When the installation menu is displayed, select the first option "Install CentOS 6.0 for use with CA DataMinder Network" using the Enter key.
4. Check the output on the monitor for any error conditions that may cause this installation to fail and prompt for keyboard input. Details of the installation process are recorded in the /var/log/anaconda*log files.

   When the CentOS installation is complete you see a login prompt on the monitor.
5. Log on to CentOS using the following credentials:

   User name

   root

   Password

   rootroot

   When you log on as root, the 'nba-package-installer' script is automatically invoked. This script performs the rest of the NBA installation.

Now the installer reports on the port configuration. See the next section on how to verify that the network ports are correctly configured.

Verify the Network Port Configuration

After you install CentOS (see page 55) and log in as root, the nba-package-installer script automatically configures all standard network ports for DHCP operation and enables these ports. To prevent reconfiguration of these ports on future root logins, the script creates a lock file.

To verify the network port configuration:

1. Verify that the port that you have connected for management use has been allocated an IP address. If the address allocation succeeds, the script displays a message like the following:

   ```
   Bringing up interface ethX:
   Determining IP information for ethX... done.
                                                          [ OK ]
   ```

2. Verify that IP addresses have been allocated to the other network ports. The success or failure to allocate an IP address depends on the wiring and network configuration.

   For example, if the script fails to allocate an IP address to ports that are not wired to a network, the script displays messages like the following:

   ```
   Bringing up interface ethX:
   Determining IP information for ethX... failed; no link present.  Check cable?
                                                              [FAILED]
   ```

   In these examples, *ethX* indicates the Ethernet port where *X* represents the port number, such as *eth1*.

3. Investigate the port configuration and cabling for those ports where success or failure is not as expected (see "Troubleshooting" for more information).

4. When the network port configuration is complete, the following message displays:

   ```
   Creating lock file to prevent repeated network port reconfiguration.
   Network ports configuration OK
   ```

   The script creates the file /opt/ca/ca-dlp/nba/lock/nba-package-installer-network.lock.

Now the installer prompts you to install the Napatech driver. Continue with the next section.

Install the Napatech High-Speed Network Card Drivers

We recommend that you use a Napatech card. After the installer configured the network ports, you are prompted to install the Napatech package.

To install the Napatech high-speed network card drivers:

1. Wait for the following installer prompt:

```
Install "Napatech 5.1.1" package from CD
Please load the "Napatech 5.1.1" CD.
Is the "Napatech 5.1.1" CD loaded (y[es] or n[o]):
```

2. If you are using a Napatech card:
   a. Insert the Napatech package CD or DVD.
   b. Reply "y".

      The Napatech software typically installs in about one minute without requiring any keyboard input.

   Note: If you are not using a Napatech card, reply "n" to the "Is the Napatech CD loaded?" prompt. Then reply "y" at the subsequent 'Skip' prompt. No Napatech software gets installed.

Now the installer prompts you to install the CA DataMinder Network Package. Continue with the next section.

More information:

Contact Your Napatech Distributor (see page 229)

Install the NBA Software Package

After installing the high-speed network card or skipping that step, you are prompted to install the NBA package from CD.

For all configurations, proceed to install the NBA software.

To install the NBA software package:

1. Insert the "CA DataMinder Network" CD or DVD.
2. Reply "y".

3. Accept the End User License Agreement.

   The CA DataMinder Network software typically installs in about 2 minutes without requiring any further keyboard input.

4. Wait for the following message:

   ```
   Installation complete, creating lock file to prevent repeated installation
   Installation process for "CA DataMinder Network" packages complete
   ```

   The CA DataMinder Network software is installed.

Assign an IP Address to the NBA Management Port (Linux Server Platform)

On the Linux Server Platform, all Ethernet ports are enabled for DHCP during the Operating System installation phase. Connect one of the Ethernet ports to a network switch. This is usually the only step necessary to assign an IP address. You can now log on to the NBA Console using the machine's name that you set up during OS installation.

Note: The default NBA console user name and password are 'webadmin' and 'webadmin' respectively.

(Optional) Configure External Bypass Unit

If the NBA installation includes an external bypass unit, configure it now so the NBA healthcheck process can correctly control it.

The NBA installation includes the following original scripts:

- /home/nba/bin/bypass_setup.sh.original
- /home/nba/bin/bypass_watchdog.sh.original

Because the original scripts can be overwritten by NBA upgrades, the installation creates the following working copies of the scripts:

- /home/nba/bin/bypass_watchdog.sh
- /home/nba/bin/bypass_setup.sh

These working copies of the original scripts are not overwritten by NBA upgrades. You modify these working copies to control your bypass unit individually.

The supplied scripts give examples for the control of Shore Micro SM2402/2404 and Interface Masters Niagara 2292 Bypass Units.

To use the supplied scripts to control the bypass unit:

1. Uncomment the appropriate lines to control these bypass units, or insert your own set of commands to control other bypass units.
2. Configure as much as possible in bypass_setup.sh. This script is only run *once* during start-up of the NBA "healthcheck" application.
3. Verify that bypass_watchdog.sh only contains commands that complete fast, because this script runs approximately every 2 to 3 seconds. Aim to keep execution of this script under about half a second, including any timeouts for commands that fail.

Some bypass units are controlled by simple serial commands (for example, the Shore Micro SM2402), while others are controlled by a software utility. The utility is provided by the manufacturer and you need to install it manually.

To install a software utility to control the bypass unit

1. Obtain the appropriate version of the software utility for use on CentOS 6 from the manufacturer.
2. Install it on the NBA server.
3. Verify that the utility has execute permissions:
   `chmod 755 utility-filename`

Set up the following connections to the bypass unit. For other bypass units, determine the equivalent sockets and connect cables accordingly.

To set up connections

1. Connect the supplied control cable between the bypass unit control port and the corresponding socket on the NBA. This connection is used for configuration of the bypass unit and to kick its watchdog to indicate that the NBA is alive.

   Shore Micro SM2402/2404

   USB B-type socket labeled "USB"

   Interface Masters Niagara 2292

   RJ45 socket labeled "Control"

   Silicom BSEM Bypass Switch

   USB B-type socket labeled "Console"

2. Connect two RJ45 patch leads between equipment on the network at the required monitoring point and the bypass unit. These connections carry the network traffic and are linked through the bypass unit relays when in "bypass" mode.

   Shore Micro SM2402/2404

   RJ45 sockets labeled "Router" and "Switch"

   Interface Masters Niagara 2292

   RJ45 sockets labeled "Network"

   Silicom BSEM Bypass Switch

   RJ45 sockets labeled "Network 0" and "Network 1"

3. Connect 2 RJ45 patch leads between the network monitoring ports on the NBA and the bypass unit. These connections carry the network traffic to the NBA through the bypass unit relays when in "online" mode.

   Shore Micro SM2402/2404

   RJ45 sockets labeled "Server1" and "Server2"

   Interface Masters Niagara 2292

   RJ45 sockets labeled "Monitor"

   Silicom BSEM Bypass Switch

   RJ45 sockets labeled "Port 0" and "Port 1"

4. Connect the power cord.

   Shore Micro SM2402/2404

   Connect to mains.

   Interface Masters Niagara 2292

   Connect to included external 12 V power supply unit.

   Silicom BSEM Bypass Switch

   Connect to included external power supply unit.

When the installation of the bypass unit is complete, you start its control from the NBA.

To start the bypass unit control

Do one of the following:

- Reboot the NBA with this command at the shell prompt:
  `reboot`
- Restart the NBA healthcheck application:
  `nbacontrol restart healthcheck`

Verify the NBA Installation

At the end of the NBA package installation, the installer runs the "installcheck" script automatically. This script verifies that the basic installation steps have completed successfully. The script does not perform an extensive test of the entire system.

You can run the script manually at any time with the following command at the shell prompt:

```
installcheck
```

To verify the NBA installation:

1. Run the installcheck script.
2. Verify that the reported NBA version is the required one.
3. If you see any error results in the check output during the first run, wait briefly, then run the check again. Often, errors are caused by a delay in some components starting after a reboot.
4. If an error is still reported after five minutes, there is a problem with the installation.

On the Linux Server Platform, the script generates output similar to the following:

```
================================================================================
=                                                                              =
=       CA DataMinder Network Installation Checker    Version 14.0.2800.0        =
=                                                                              =
================================================================================

OS = CentOS Linux release 6.0 (2.6.32-71.el6.x86_64)                        OK
NBA version = 14.0.2800.0                                                   OK
Napatech card = NT4E-4T-INL In-line Network Adapter PCIe 4x1Gb RJ45         OK
Napatech driver version = 1.3.0.14565                                       OK
NBA services                                                                OK
--------------------------------------------------------------------------------
Overall result of all installation checks (All passed)                      OK
--------------------------------------------------------------------------------
```

Upgrade the NBA

You can upgrade your existing NBA application package or Napatech driver package (see page 57) on Linux servers.

To upgrade the NBA

1. Get a new NBA application package or Napatech driver package.

   You can get the new distribution packages on a CD/DVD or download them to your existing NBA.

2. (Napatech driver packages only) Follow these steps:

   a. Unload the Napatech drivers. Run these commands:

   ```
   nbacontrol stop nba
   cd /opt/napatech3/bin
   ./ntstop.sh
   ./ntunload.sh
   ```

   b. Unpack the new driver. Run these commands:

   ```
   cd /home/install/napatech
   tar -xvzf ntinl_package_3gd_linux_5.1.1.tar.gz
   ```

   c. Install the new driver. Run these commands:

   ```
   cd ntinl_package_3gd_linux_5.1.1
   ./package_install_3gd.sh
   ```

3. Install the package updates using the nba-package-installer script with the following commands at the shell prompt:

   ```
   rm -f /opt/ca/ca-dlp/nba/lock/nba-package-installer.lock
   /opt/ca/ca-dlp/nba/sbin/nba-package-installer
   ```

4. When prompted to load the Napatech in-line driver CD, type 'n'. At the next prompt, type 'y' to skip the Napatech driver install. You have already completed this task in step 2.

5. (If installing a package from a CD/DVD) Follow the instructions in the respective 'How to Install the Software' sections.

6. (If installing a downloaded package) Follow these steps:

   a. Reply "n" when the nba-package-installer prompts you to insert a CD or DVD.

   b. Reply "y" when you are prompted for a preloaded package.

   c. Enter the full path of the directory containing the downloaded package.

      Existing packages are upgraded.

   When the upgrade is complete, any existing NBA policy and configuration settings are retained.

7. Verify the release information for new policy and configuration settings that may be available to add to your files.

More information:

Install the Software (Linux) (see page 54)
Install the NBA Software Package (see page 57)
Install the Napatech High-Speed Network Card Drivers (see page 57)

Chapter 5: Configuring the NBA

You use the NBA console to configure the NBA. First, you must set secure passwords for NBA accounts. You must also:

- Set the NBA to run in active or passive mode.
- Set the NBA output mode.
- Set the time zone, date and time.
- Bring the NBA online.

These essential configuration tasks are described in the following sections.

This chapter also includes advanced configuration tasks such as setting up FTP access, configuring NBA policy and filters, and setting up quarantining. This chapter also describes how to configure user notifications when the NBA blocks events such as emails, webmails, and uploads.

This section contains the following topics:

NBA Console

The NBA console is a Web-based interface that you use to manage NBA operations, monitor the hardware and collect diagnostic data.

Browser requirements

The NBA console runs in the following browsers:

- Google Chrome
- Mozilla Firefox
- Microsoft Internet Explorer 8, 9, or 10

Note: The NBA console may run in other browsers, but these have not been tested

Bivio 7000 Appliances or Linux Server Platform

The key screens are listed below:

Administration

Use this screen to:

- Switch NBA packet processing on or off.
- Run in active or passive mode (stream blocking).
- Specify whether the NBA outputs captured data to local folders or a remote policy engine.
- Set secure logon credentials for the NBA console user and NBA FTP user.

Analyzers

This screen shows status and statistics for each Network Analyzer process. Statistics are updated every ten seconds. Refresh your browser to update the table.

More information is available in Status Details for CPUs (see page 210).

Note: Each CA DataMinder Network contains multiple processing units that handle a portion of the network traffic.

Filters

This screen shows filter statistics and filter settings. You can add, edit and delete network level and application protocol filters.

Health

This screen shows status information, allowing you to assess the state of CA DataMinder Network.

For more information, see Health Screen (see page 206).

Interfaces

This screen includes configuration settings for data collection ports, interface ports (used for device management), and other networking settings.

Messages

Shows recent event messages and Linux system messages.

Newest messages are listed first.

Policy

Use this screen to configure some NBA policy settings. These settings control data capture operations.

For more information, see Policy Screen (see page 208).

SNMP

Use this screen to configure SNMP monitoring and write access, SNMP configuration details (such as system name), and traps. You can also download the SNMP MIB for your SNMP Manager.

For more information, see Configure SNMP (see page 201).

SSL

This screen shows SSL statistics for each Network Analyzer. See the console help for an explanations of these statistics.

For more information, see SSL Statistics (see page 126).

Time

Use this screen to set the time zone, date, and time for CA DataMinder Network. For more information, see Set the Time Zone, Date and Time (see page 72).

CA DataMinder Network uses the date and time to assign timestamps to events that do not have an inherent timestamp. For example, an email is already time-stamped by its sender but a Webmail requires a timestamp.

Set Secure NBA Account Passwords

The default passwords for user accounts are not secure. Before you use the NBA, you must change them.

Follow these steps:

Change the default passwords for the following user accounts:

- NBA console user
- NBA root user
- FTP user

Note: The NBA console uses secure HTTPS to transfer data from your web browser to the NBA.

Set a Password for the NBA Console User

You use the admin and webadmin user accounts to log onto the NBA console. You must choose a more secure password for these accounts before you start using the NBA. On Bivio 7000 appliances or the Linux Server Platform, the passwords are both ‘webadmin’.

Note: The ‘webadmin’ user (see step 4) is an alternative name for the NBA console user.

To set a new password

1. Connect to the NBA using the IP address you assigned to the management port (see the previous section).
2. In the NBA Logon screen, supply the default username and password.
3. Go to the Administration screen.
4. (Bivio 7000 Platform only) Go to the Password for **'webadmin'** User Account and FTP Server for **'admin'** User Account sections.
   a. Enter the new passwords for the ‘webadmin’ and 'admin' user accounts.

      The maximum length for this new password is 8 characters.

   b. Click Apply to confirm the change.
5. (Linux Server Platform only) Go to the Password for **‘webadmin’** User Account and FTP Server for **‘smb’** User Account sections.
   a. Enter the new passwords for the ‘webadmin’ and ‘smb’ user accounts.
   b. Click Apply to confirm the change.

Set a Password for the NBA Root User

Before you start using the NBA, you need to set a secure password for the NBA root user. This section describes how to set a new, more secure password.

Note: The NBA root user account is reserved for administering the NBA operating system; it is not used under normal circumstances and you cannot use this account to log on to the NBA console.

The following instructions require you to logon to the NBA using a terminal emulation application such as PuTTY (PuTTY is a free implementation of TELNET and SSH for Win32 and UNIX platforms).

To set a secure password for the NBA root user

1. Connect to the NBA using your terminal emulation application. To do this, use the IP address you assigned to the management port (see the previous section).

   This launches a connection to the NBA's operating system.

   Note: You must connect using the SSH protocol. You cannot connect using TELNET.

2. At each command prompt, enter the following details:

   a. At the 'login as' prompt, enter the default user name and password for the root user.

      - On the Bivio 7000 platform, the default password for the ‘root’ user is ‘root’.
      - On the Linux Server Platform, the default password for the ‘root’ user is ‘rootroot’.

   b. At the prompt, type the command ‘passwd’.

   c. When prompted, enter the new password for the root user.

      You are then prompted to re-enter it to confirm the change.

3. Log out of the NBA. To do this, do one of the following:

   - Type ‘exit'.
   - Press Ctrl+D.
   - Close your terminal emulation application.

Essential Configuration

The following sections describe essential NBA configuration tasks.

Choose Active or Passive Mode

To toggle the NBA between active and passive modes when the NBA is connected inline, change the Stream Blocking setting in the NBA console.

- When stream blocking is on, the NBA runs in active mode.
- When stream blocking is off, the NBA runs in passive mode.

By default, stream blocking is off. This state is equivalent to the <active value="false"> lines in the policy and configuration files, nbapolicy.xml and nbaconfig.xml.

To change the Stream Blocking settings

1. Log on to the NBA console and browse to the Administration screen.
2. Go to the Configuration Status and Mode of Operation section.
3. Click one of the Stream Blocking buttons.
   - Enable Stream Blocking to change from passive to active mode.
   - Disable Stream Blocking to change from active to passive mode.

   The <active> tags in nbapolicy.xml and nbaconfig.xml are updated automatically when you change the Stream Blocking setting in the NBA console.

Important! If you toggle the NBA software between active and passive modes, verify that the wiring to the NBA uses the correct ports to match the NBA mode. For port details, see Connecting the NBA Ports (see page 29).

More information:

Passive and Active Mode (see page 12)

How to Choose Active or Passive Mode?

- If you only want to capture and analyze files and emails, switch stream blocking off (passive mode).
- If you want the NBA to capture, analyze, and block files and emails, switch stream blocking on (active mode).

  Note: When stream blocking is switched on, the NBA can only block files and emails if it is connected inline (between the corporate LAN and the Internet). If stream blocking is enabled while not inline, the event log will incorrectly report that an action was performed.
- If you want to be able to decode SSL sessions, the NBA must be inline, and stream blocking must be switched on (active mode).

How Policy Processing Affects Objects Passing Through the NBA

When the NBA passes an object to a policy engine for analysis, it forwards all associated data packets onto the receiving computer, except for the final packet in the stream. Then, when policy processing is complete, it receives a 'block' or 'allow' action.

If the NBA receives an 'allow' action, it releases the final packet to its destination. The receiving computer can then complete the transaction, for example, an email send operation, or an HTTP Post operation.

Choose the Output Mode

You can configure the NBA to output captured files and emails either to folders on the local disk or via a socket connection to a CA DataMinder hub or policy engines. Or it can do both or neither. By default, the NBA outputs data to a socket connection. This output mode is equivalent to the following line in nbaconfig.xml:

```
<capture value="socket">
```

To set the output mode in the NBA console

1. Log on to the NBA console and browse to the Administration screen.
2. Go to the Configuration Status and Mode of Operation section.
3. Choose one of the Output Mode options:

   Disk

   The NBA saves copies of captured files and emails to the \files and \mails folders on the NBA.

   Socket

   The NBA sends captured items via a socket connection to CA DataMinder policy engines for processing. The socket connection can either be to a hub or direct to multiple policy engines.

   Socket and Disk

   The NBA simultaneously saves copies of captured files and emails to folders on the NBA and sends copies via a socket connection as described above.

   Off

   The NBA does not output any captured files and emails.

   However, stream blocking is still possible when the Output Mode is Off. If required, you can process data packets and block them using 'Prohibit' actions in the filtering tab.

Set the Time Zone, Date and Time

By default, the time-stamps for files and emails are set to the date and time when they were captured by the NBA. Therefore, before you start the NBA you must set the correct time zone, date and time to ensure that any captured files are correctly time-stamped.

To set the time zone, date and time

1. Log on to the NBA console and browse to the Time screen.
2. Set the time zone, date and time
3. Reboot the NBA. To do this:
   - On Bivio 7000 appliances, click the Reboot button in the CPUs screen.
   - On the Linux Server Platform, click the Reboot button in the Administration screen.

Set the NBA Online or Offline

To set the NBA online or offline, you change the Packet Processing setting in the NBA console. This is the recommended method for turning the NBA on or off.

- By default, packet processing is on. When packet processing is **on**, the NBA is online and data packets are processed as normal.
- When packet processing is **off**, the NBA is offline and packets pass through the NBA uninterrupted and without being processed.

To switch packet processing on or off

1. Log on to the NBA console and browse to the Administration screen.
2. Go to the Configuration Status and Mode of Operation section.
3. Click one of the Packet Processing buttons.
   - Enable Packet Processing to bring the NBA online.
   - Disable Packet Processing to take the NBA offline.

   The <online> tags in nbapolicy.xml and nbaconfig.xml are updated automatically when you change the Packet Processing setting in the NBA console.

FTP Folders and Files

Note: Before you can browse the folders on the NBA, see the FTP Folder Connection Requirements (see page 74).

The FTP folder structure is as follows:

config

Contains the NBA configuration files:

- **blocktemplate.html:** An HTML template for block actions.
- **filters.txt:** Details of currently active filters.
- **nbaconfig.xml:** Current NBA configuration settings.
- **nbaconfig.txt:** Easy-to-read version of current configuration settings.
- **nbapolicy.xml:** Current NBA policy.
- **nbapolicy.txt:** Easy-to-read version of current policy settings.
- **nbapolicy_last.xml:** When a policy file is successfully interpreted and used, a copy is made under this name.
- **nbaroot.p7b/nbaroot.crt** and **nbarevoked.p7b/nbarevoked.crt:** These files are the Trusted and Revoked master certificates for SSL Decode. For more information, see Distribute the Master Certificates (see page 116).

diag

Contains files of diagnostic data. You generate these by clicking the Diagnostics Collection button in the NBA console. This in turn runs the diag.sh utility.

Each diagnostic file is a zipped file containing various subfolders. These subfolders contain copies of the \config files, log files, and status files for individual NBA components. You can download these diagnostic files to your computer.

files

Contains files reassembled from data packets analyzed by the NBA.

log

Contains various NBA log files. Separate log files are maintained for each CPU.

The \err and \out subfolders contain log information created by NBA services at startup. These logs may be useful for troubleshooting and are packaged into diagnostic files when these are created.

mails

Contains EML emails reassembled from data packets analyzed by the NBA.

stagingfiles

A temporary storage area, used while the NBA assembles files from their constituent parts distributed across multiple data packets.

stagingmails

A temporary storage area, used while the NBA assembles emails from their constituent parts distributed across multiple data packets.

More information:

About nbaconfig.xml (see page 231)
Log Files (see page 205)

Switch on FTP Access

Before you can access the NBA FTP folder from your computer, you must enable FTP access to the NBA (by default, FTP access is disabled). You do this in the Administration screen of the NBA console.

1. In the Administration screen, go to the Access Control section.
2. Go to the FTP Server field.
3. Set up access to the FTP folder.
   - Click Enable to enable FTP access
   - Click Disable to disable FTP access.
4. If you have not already done so, choose a secure password for the FTP user.

Comply With FTP Folder Connection Requirements

Verify that you have enabled the FTP server and set a password in the Administration tab of the NBA console.

- (Bivio 7000 Platform) The FTP account name is 'admin'.
- (Linux Server Platform) The FTP account name is 'smb'.

The FTP server requires an Active Mode FTP client. For example, the Windows command line ftp.exe utility supports Active Mode. It is also possible to use Windows Explorer or other browsers in FTP mode.

Before you browse to the FTP folder in Windows Explorer for the first time, configure active mode:

Follow these steps:

1. Open the Internet Options dialog, and go to the Browsing section of the Advanced tab.
2. Verify that the following Internet Explorer settings are set correctly:
   - Enable the 'Enable FTP Folder View (Outside of Internet Explorer)' setting.
   - Disable the 'Use Passive FTP (for firewall and DSL Modem compatibility)' setting.

You can now log on to your FTP server.

Follow these steps:

1. Do one of the following
   - Start an FTP client.
   - Open the Windows Explorer.
2. Replace *name* by your FTP account name, and replace *NBA* by your NBA's URL or IP address, and connect to the FTP server:
   `ftp://name@NBA`

   Note: If you used the 'Open folder in Windows Explorer' option from Internet Explorer, you may need to re-enter this URL in the Windows Explorer address bar.

More Information:

Bivio 7000 FTP IPv6 Connections (see page 215)

NBA Policy

The NBA policy is analogous to conventional CA DataMinder machine policies. An NBA policy includes the rules (or **filters**) that jointly determine how the NBA handles data packets passing through the NBA. The policy also specifies the policy engines or PE hub that the NBA will use when running in active mode. Finally, the policy determines how the NBA manages its log files (for example, the default level of logging, and the maximum number and size of log files).

NBA policies can operate in tandem with CA DataMinder user policies. For example, an NBA policy may dictate that specific types of communication are passed (as reassembled files or emails) to a CA DataMinder policy engine to apply a CA DataMinder user policy to that communication, but that other communications are permitted to pass through the NBA uninterrupted.

On Bivio 7000 appliances, you can define most policy settings in the NBA web console. You can fully define an NBA policy by editing nbapolicy.xml. Find this file in the \config folder on the NBA.

Because complex XML files can be hard for people to read and comprehend, a summary of the NBA policy is written to a text file, nbapolicy.txt.

More information:

NBA Filters (see page 77)
Specifying NBA Policy in XML (see page 153)
Policy Screen (see page 208)

Example Policy

You may, for example, want to set up the NBA to analyze all outbound traffic to prevent IP loss.

To implement this policy, you need a combination of filters:

1. Focus on TCP data and exempt data sent using UDP.
2. Analyze Webmails but ignore outbound SMTP emails (because your organization has already deployed CA DataMinder agents to control emails sent through your Exchange servers).
3. Block all file uploads or posts to newsgroups except those uploaded from specific authorized machines.

NBA Filters

The NBA uses filters to check every IP packet it sees. These packets are then analyzed, decrypted, monitored, prohibited or ignored, depending on how the filters are configured. The NBA supports network filters and application filters. Both are defined in nbapolicy.xml.

- **Network filters** can optionally check for TCP or UDP data packets, the IP addresses and the port numbers they use.
- **Application filters** can check a packet's IP addresses and port numbers. They can also identify the protocol (which indicates the message type). For example, they can identify email or IM packets, or file uploads.

An NBA policy typically contains multiple application and network filters. To simplify filter management, you can organize your filters into groups. You can also assign descriptive filter names to help administrators understand the purpose of each filter.

You can assign multiple IPv4, or IPv6 addresses, or both, to network and application filters. You add and edit filters in the Filters screen of the NBA console. You can also directly edit the NBA policy file, nbapolicy.xml.

More information:

Filtering Methods (see page 78)
Default Filters (see page 78)
Add or Delete Filters (see page 78)
Specifying IPv6 Adresses (see page 80)
Appending Port Numbers to IP Addresses (see page 80)
Filter Names (see page 81)
Filter Groups (see page 81)
Multiple Filters Are Applied Successively (see page 81)
Which Filter Takes Precedence? (see page 83)
Application Protocols (see page 84)

Filtering Methods

An NBA policy can use the following methods to detect filter data packets:

Filtering by protocol

Network filters can detect TCP or UDP packets, or both. Application filters can detect packets sent using a range of IM, Webmail, email or file transfer protocols, such as POP3, YAHOOIM, or HTTPPOST; see Application protocols (see page 84).

Filtering by IP address and TCP port

All filters can detect packets sent from specified IP addresses or transmitted to specified TCP ports. For example, you can configure filters to ignore SMTP data from corporate email servers if that data is already being captured by CA DataMinder.

More information:

IP Address and Port Filters (see page 180)

Default Filters

Default filters are automatically set up in nbapolicy.xml. The default filters enable the NBA to analyze (but not decrypt) all network traffic. They also enable the NBA to send files and emails to policy engines for analysis. If you remove these default filters, the NBA ignores all network traffic.

Set Default Filters to 'Ignore' If You Add New Filters

If you add filters for specific protocols or IP address ranges, we recommend that you set the default filters to have an Ignore action. This configuration ensures that all incoming packets are processed by a specific filter and are accounted for in the statistics logs.

Add or Delete Filters

You add and delete filters in the Filters screen of the NBA console.

To add a filter in the NBA console

1. Go to the Filters screen of the NBA console.
2. Go to the filter group where you want to add the new filter.

3. Click the  Add button in the Add/Delete column.
   - Enter a name for the new filter.
   - Specify the IP address range and the protocols.

     In the IP Addresses/Protocols column, enter IP addresses in the upper field and enter the protocols in the lower field.

     Note: CA DataMinder 14.5 supports both IPv4 and IPv6 address formats (see page 80). You can assign multiple IPv4, or IPv6 addresses, or both, to network and application filters. Specify each address, address range, port, or port range on a separate line.
   - Specify the filter action and the logging level.

     In the Action/Logging column, enter the filter action in the upper field and the logging level in the lower field.
   - Click the checkbox in the Enable column.
4. Click the Apply button.

To delete a filter in the NBA console

1. Go to the Filters screen of the NBA console.
2. Go to the filter group containing the filter that you want to delete.
3. Click the  Delete button the Add/Delete column.
4. Click the Apply button.

To add or delete filters by editing nbapolicy.xml directly

You can also add or delete filters by editing nbapolicy.xml directly. For details about this file, see Specifying NBA Policy in XML (see page 153).

Specifying IPv6 Adresses

Use the following formats:

- Dotted format for IPv4:
  `192.168.0.3`
- Colon-separated format for IPv6:
  `fe80::214:c2ff:fec8:c920`
- All addresses in the range including the lower and upper address:
  `10.0.1.53-10.0.1.80`
- All addresses from 10.0.0.0 to 10.0.255.255:
  `10.0`
- All addresses from 10.0.0.0 to 11.255.255.255:
  `10/7`
- All addresses from "2001:0db8:85a3::" to "2001:0db8:85a3:ffff:ffff:ffff:ffff:ffff":
  `2001:0db8:85a3/48`
- All addresses and ports:
  `*`

  Note: You cannot combine the asterisk wildcard with any other characters.

Appending Port Numbers to IP Addresses

Enclose IPv6 addresses in brackets when you specify a port number or port range.

Use the following formats:

- IPv4 address and a port:
  `192.168.0.5:10`
- IPv6 address and a port:
  `[fe80::e828:209d:20e:c0ae]:375`
- All addresses where the port number ranges from 137 through 139:
  `:137-139`
- All addresses from 192.168.0.0 to 192.168.255.255 where the port number is from 1024 through 65535:
  `192.168:1024-65535`
- All addresses from fe80:: to fe81:: where the port number is 80:
  `[fe80::]-[fe81::]:80`

More information:

IP Address and Port Filters (see page 180)

Filter Names

When you define an NBA policy, you can add descriptive names for your NBA filters. When you edit the policy, these names are automatically copied to nbapolicy.txt along with the other policy settings. This enables administrators or other technical staff to more easily understand what each filter does.

You can assign filter names in the Filters screen of the NBA console.

More information:

XML Syntax: nbapolicy.xml (see page 155)
NBA Policy Copied to a Text File (see page 204)

Filter Groups

To simplify filter management, you can add any combination of network filters, or any combination of application filters, to a named group. You can then quickly disable or re-enable the filter group instead of having to individually disable or re-enable each filter in the group.

But why organize filters into groups? You may want to do this if you have to use multiple filters to implement a particular NBA policy. For example, consider an NBA policy to 'Manage traffic from IT department'. Such a policy may require the NBA to ignore traffic from some IP addresses on specified protocols, and to capture traffic from other IP addresses on other protocols. To quickly enable or disable the entire policy, you only need to enable or disable the filter group.

You can create filter groups in the Filters screen of the NBA console.

More information:

XML Syntax: nbapolicy.xml (see page 155)

Multiple Filters Are Applied Successively

In both passive and active modes, the NBA uses filters to check every IP packet it sees. These are filters that define which communications are captured, blocked or sent to a policy engine for processing.

The NBA supports network filters and application filters. These support the following actions.

Network filter actions

- **Ignore:** Packets are exempted from further NBA processing and permitted to continue;
- **Prohibit:** Packets are blocked;
- **Analyze:** Packets are passed from a network filter to an application filter.
- **Decrypt:** SSL session packets are decoded and then passed to an application filter. Matching packets that do not need decryption are also passed to an application filter.

Application filter actions

- **Ignore:** Packets are exempted from further NBA processing and permitted to continue;
- **Prohibit:** Packets are blocked;
- **Analyze:** Packets are passed from an application filter to a policy engine.
- **Monitor:** Packets are passed to a policy engine but cannot be blocked, even if the policy engine requests that they are blocked as a result of processing.

How Are Filters Applied?

Data packets are filtered as they pass through the NBA. The filters are defined in nbapolicy.xml. In this example, one network filter and two application filters are active. The network and application filters operate to successively narrow down the communications that must be sent to a policy engine for processing. This is the slowest part of the process.

Network filters are always applied first, followed by application filters. For best performance, configure the network filters to decrypt and/or analyze the smallest amount of network data possible for the application filters. In turn, configure the application filters to analyze or monitor the smallest amount of network data for reassembly into files and emails that are sent to a policy engine.

The following steps show how the NBA applies filters to data packets.

1. A network filter checks data packets for their protocol (TCP or UDP). In this example, the filter action is set to analyze TCP packets. Any UDP packets are ignored and permitted to continue through the NBA without further intervention.
2. When the NBA detects any TCP packets, it analyzes them to identify the application protocol.

3. The NBA then applies the appropriate application filter to the packets.
   - In this example, an application filter for SMTP data sent from specific IP addresses is set to 'ignore'. These packets are permitted to continue.
   - At the same time, an application filter for all other protocols is set to ‘analyze’. These non-SMTP packets are passed to a policy engine for processing.

   In this example, the ‘monitor’ and 'prohibit' actions for application filters are not used.
4. The policy engine, after analyzing the non-SMTP communication, either blocks it or allows it to continue.

Which Filter Takes Precedence?

Important! The order in which filters are listed in the nbapolicy.xml configuration file is irrelevant!

Filter precedence and optimization in the NBA means that only one network filter and one application filter are applied to any data stream.

When the NBA examines a packet, any network filters are always applied first, followed by any application filters. But if multiple network filters are defined, or multiple application filters, with potentially conflicting criteria, how does the NBA determine filter precedence?

Filter precedence is based on IP address. Specifically, the filter with the narrowest IP address range takes precedence over all other filters. That is, this filter’s action (Analyze, Prohibit, or Ignore) gets applied to data packets arriving from a specified IP address, even if these data packets also meet the criteria of other filters, each of which may specify a different action.

Filter precedence is applied successively, so after the ‘narrowest range’ filter has been implemented, precedence passes to the filter with the next narrowest address range, and so on. This is best illustrated with an example.

Example

An NBA policy comprises three application filters to target machines on the 10.0.*.* network. In the nbapolicy.xml, the filters are listed in this order:

- **Filter 1** analyzes packets sent from machines with a 10.0.1.* IP address.
- **Filter 2** ignores any packets sent from machines with a 10.0.*.* IP address.
- **Filter 3** prohibits packets sent from a specific machine, whose IP address is 10.0.1.53.

In this example, filter 3 takes precedence; all packets from the 10.0.1.53 machine are prohibited (that is, blocked), regardless of any other filters.

Next, the NBA implements filter 1; any packets from a machine on the 10.0.1.* subnet are passed to a policy engine for analysis *except for packets from* 10.0.1.53*, which are prohibited as described above.*

Finally, any remaining packets from machines on the 10.0.*.* network are ignored. That is, they are permitted to pass through the NBA without interruption.

XML Order	IP Address	Filter Action	Precedence
Filter 1	10.0.1.*	Analyze	2nd
Filter 2	10.0.*.*	Ignore	3rd
Filter 3	10.0.1.53	Prohibit	1st

Application Protocols

NBA application filters can detect data packets transmitted using the following protocols:

IM protocols

IM_ALL

All recognized instant message formats.

AOLIM, ICQIM

AOL and ICQ instant messages. Both these protocol options detect the same protocol in the NBA which is called OSCAR. However, AOL instant messages are usually encrypted and the AOL client cannot be configured to trust the NBA master certificate.

JABBERIM

Jabber (XMPP) instant messages. Can be decrypted from SSL sessions if the Jabber client is configured to trust the NBA master certificate.

MSNIM

Windows Live Messenger (formerly MSN) instant messages.

SIPIM

Instant messages sent using an application that uses the Session Initiation Protocol (SIP).

YAHOOIM

Yahoo! instant messages.

Note: Many of these protocols may be encrypted and some cannot currently be detected by the NBA. However, the presence or absence of encryption varies from one IM client version to another and may also depend on account preference settings.

Email and Webmail protocols

AOLMAIL

Outbound messages sent using AOL Mail.

DELTASYNC

Outbound messages sent using Windows LiveMail.

To capture all Windows LiveMail messages, you must also use the HOTMAIL protocol.

Note: The NBA cannot block messages *received* by a Windows LiveMail client. It can only block messages *sent* from a Windows LiveMail client.

GMAIL

Outbound messages sent using Gmail (or Google Mail). These messages can be decrypted from SSL sessions.

HOTMAIL

Outbound messages sent using Microsoft Hotmail or Windows LiveMail. These messages can be decrypted from SSL sessions.

To capture all Windows LiveMail messages, you must also use the DELTASYNC protocol.

POP3

Messages received using an email client that uses the POP3 protocol (most commonly, Outlook Express).

Note: The NBA cannot block POP3 emails.

SMTP

Messages sent using SMTP (Simple Mail Transfer Protocol). This typically includes messages sent over the Internet from an Exchange or Domino server, or sent from an email client such as Outlook Express. These messages can be decrypted from SSL sessions.

SMTPDEST

Messages sent to specified destinations using SMTP. These messages can be decrypted from SSL sessions.

You must list the destination IP addresses in the filter definition.

SMTPSRC

Messages received from specified destinations using SMTP. These messages can be decrypted from SSL sessions.

You must list the source IP addresses in the filter definition.

WEBMAIL

Messages sent using a Webmail protocol that is decoded by the NBA. These include AOL Mail, Gmail (or Google Mail), Hotmail, Windows Live Mail and Yahoo! Mail.

Note the NBA can only detect outbound (sent) messages. It cannot detect inbound messages, arriving in a user's Webmail inbox.

YAHOOMAIL

Outbound messages sent using Yahoo! Mail.

File protocols

FTP

Files transferred using FTP (File Transfer Protocol).

FTPGET

File transfers over FTP downloaded from a server.

FTPPUT

File uploads over FTP.

HTTPGET

Files downloaded from a web site. Add ".HTM" to the nbaconfig.xml file's httpgetfiletypes to include web page content. HTTPGET messages can be decrypted from SSL sessions.

HTTPPOST

Files uploaded to a Web site. HTTPPOST messages can be decrypted from SSL sessions.

HTTPURL

Web browsing. The NBA detects HTML Web pages requested from a Web site, including a URL plus page title and other HTML metadata (keywords, author, and description if available). It does *not* include the actual page content (see HTTPGET). HTTPURL messages can be decrypted from SSL sessions.

SMB

Files accessed on a remote server using the SMB protocol.

The NBA can detect and block attempts to browse remote files, but it cannot analyze the contents of those files.

Other protocols

ALL

You can configure the NBA to detect all known application protocols.

SKYPE

The NBA can detect the start of Skype sessions, but it cannot analyze their content or block the packets.

NNTPGET

Messages read from a news group using the Network News Transfer Protocol (NNTP).

NNTPPOST

Messages posted to a news group using the Network News Transfer Protocol (NNTP).

Blocked Emails: Notifying Users

If the NBA passes an email to a policy engine for analysis and that email is subsequently blocked, you can set up CA DataMinder to send a notification email to the sender, informing them that their email was not sent. The notification procedure depends on whether or not CA DataMinder recognizes the sender:

Scenario 1: Known Sender

In this situation, the sender uses Outlook or Notes to send an email to an external recipient. The recipient can be a Webmail account (such as MyFriend@Hotmail.com) or a corporate account in an external organization (such as MyFriend@Company.com).

Because the sender is known to CA DataMinder (their email address can be mapped to a CA DataMinder user account), CA DataMinder is able to send a notification email directly to them. This notification email is sent internally, via your corporate email server.

Scenario 2: Unrecognized Sender

In this situation, the sender uses a Web-based service such as Hotmail to send a Webmail to an external recipient. As for scenario 1, the recipient can be another Webmail account or a corporate account in an external organization.

In either case, the sender's Webmail address (such as mysteryman@hotmail.com) cannot be mapped to a CA DataMinder user account. The notification email is therefore sent externally through the NBA to the sender's Webmail address.

More information:

Setting Up Notification Emails (see page 89)
Scenario 1: Sender Is Known to CA DataMinder (see page 90)
Scenario 2: Unrecognized Sender (see page 91)

Setting Up Notification Emails

To enable notification emails, you need to set up:

Socket API

If a policy engine applies a 'Block with Notification' control action to an email, the Socket API sends a notification email to a designated SMTP server. In addition to the usual Socket API setup, you must edit the registry on the Socket API host machine to configure a connection to an SMTP server. You must also define the notification sender's address and add an 'exemption domain' to a custom email header:

Notification sender's address

This is the address shown in the From: field of the notification email (such as Compliance@Unipraxis.com). To define the notification sender's address, you must edit the NotificationFromAddress registry value.

Domain header for notification email

When the Socket API generates a notification email, it writes the name of a DNS domain to a custom header in the email. When the NBA detects this domain, it exempts the notification email from policy. To set this domain, you must edit the SMTPDNSHostName registry value.

For full details about configuring the Socket API, see the *Archive Integration Guide*; search for 'Socket API, configuring'.

NBA policy

In scenario 2, the 'blocking notification' email is routed through the NBA. To ensure it can pass through the NBA without being processed, you must edit the NBA policy so that it can recognize emails sent by the Socket API.

To do this, edit the <enterprisednslist> policy tag in nbapolicy.xml so it includes the domain specified by the Socket API SMTPDNSHostName registry value, (see above).

<enterprisednslist> is described in the section for Settings tags.

User policy triggers

You must edit the user policies for your ordinary users, setting up the Outgoing email control triggers and a control action to detect unauthorized emails and Webmails and apply a 'Block with Notification' control action. Use the Message To Users trigger setting to specify the body text of the notification email.

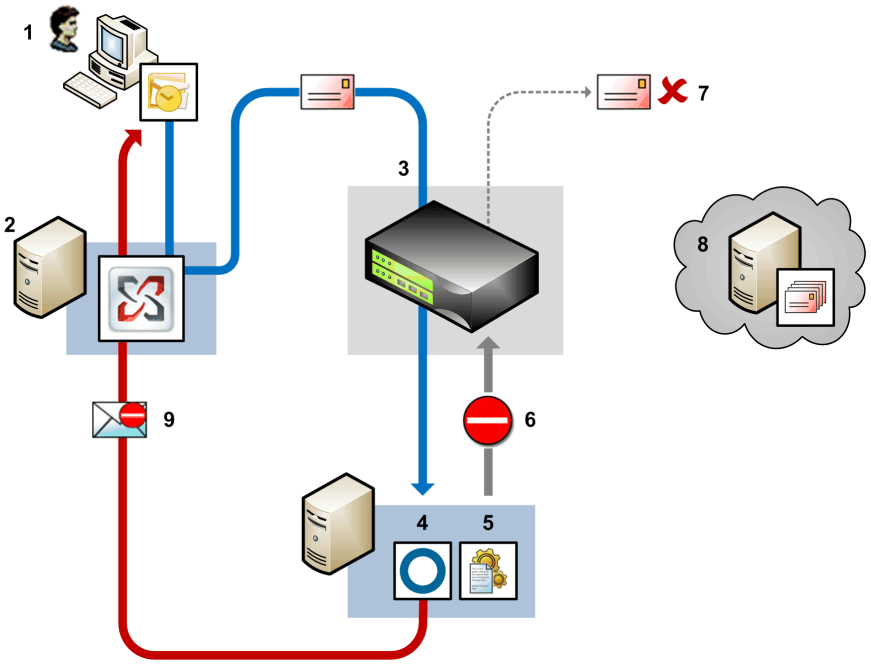
More information:

Settings Tags (see page 169)
User Policy Changes (see page 137)

Scenario 1: Sender Is Known to CA DataMinder

Here, the sender is recognized as a CA DataMinder user. That is, the From: address can be mapped onto an existing CA DataMinder user account. For example, this will be the case if the sender uses Microsoft Outlook and your CA DataMinder user accounts are synchronized with Active Directory.

The NBA passes the outbound email to a policy engine, which applies the sender's own policy. When the email is blocked, a notification email is sent internally to the sender's corporate email address.



Example NBA notification email: Known sender

Using Outlook, a user sends an email to an external recipient (**1**). The corporate Exchange server (**2**) routes this email to the Internet via the NBA (**3**). The NBA passes the email via the Socket API (**4**) to a policy engine for processing (**5**).

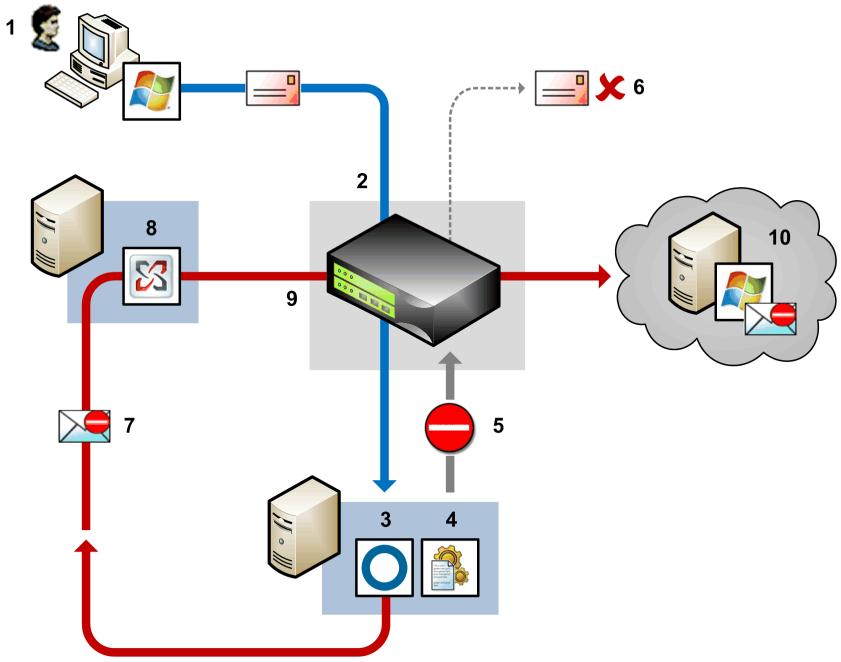
The policy engine maps the sender's email address to a CA DataMinder user account and applies that user's policy. After applying email triggers, the policy engine calls back to the NBA to block the email (**6**). The email is blocked (**7**) and never reaches the external email server (**8**).

The policy control action is set to 'Block with Notification', so the Socket API (**4**) generates a notification email (**9**). The sender address in the From: field is configurable; for example, it can be set to 'Compliance@Unipraxis.com'.

The notification email is delivered to the original sender (**1**) by the corporate Exchange server (**2**). If a CA DataMinder Exchange server agent is running, it automatically exempts the notification email from further policy processing.

Scenario 2: Unrecognized Sender

Here, a Webmail is sent from an unrecognized address (such as mysteryman@hotmail.com). The NBA detects the Webmail and passes it to a policy engine, which applies the External Sender's policy. When the Webmail is blocked, a notification email is sent externally, through the NBA, to the sender's Webmail address. The NBA detects that the notification email was generated by CA DataMinder and exempts it from policy processing.



Example NBA notification email: Unrecognized sender

Using Internet Explorer, a user sends a Webmail (**1**) to another Hotmail recipient. The NBA (**2**) detects the Webmail and passes it via the Socket API (**3**) to a policy engine for processing (**4**).

Because the sender cannot be mapped to a CA DataMinder user account, the policy engine applies the External Sender's policy to the Webmail. After applying email triggers, the policy engine calls back to the NBA to block the Webmail (**5**); the Webmail is duly blocked (**6**).

The policy control action is set to 'Block with Notification', so the Socket API (**3**) generates a notification email (**7**). The From: field is set to Compliance@Unipraxis.com. The corporate Exchange server (**8**) then forwards the notification email to the original sender's Hotmail address.

The notification email passes through the NBA (**9**) on its way to the Hotmail server (**10**). The NBA automatically exempts the notification email from policy processing.

The original sender (**1**) sees the notification email when they next view their Hotmail inbox.

Quarantined Emails

CA DataMinder is able to quarantine certain categories of documents addressed to external recipients until they have been approved by an appropriate representative.

If you want the NBA to quarantine emails and Webmails, you need to deploy the Quarantine Manager. You also need to edit the NBA policy to prevent emails released from quarantine being reprocessed (and quarantined again!).

CA DataMinder policy engines can quarantine emails without notifying the sender (using a 'Quarantine Quietly' control action) or they can send an email notifying the sender that their email has been quarantined. If your policy engines apply a 'Quarantine With Notification' control action, you will also need to configure the Socket API to send the notification emails.

Set Up NBA Quarantining

To enable the NBA to quarantine and release emails, set up the following configuration:

Quarantine Manager

First, set up a 'QM domain user' with rights to log in to your email server. Finally, after installing the Quarantine Manager, edit its registry to configure how it handles quarantined emails.

This setup procedure is the same as the general Quarantine Manager setup. For full details, see the *Deployment Guide*; search the index for 'Quarantine Manager'.

Socket API

If a policy engine applies a 'Quarantine with Notification' control action to an email, the Socket API generates the notification email and sends it to a designated SMTP server.

The Socket API setup for sending 'quarantine notification' emails is the same as the setup for sending 'blocking notification' emails. In particular, you need to edit the NotificationFromAddress and SMTPDNSHostName registry values.

For full details, see the *Archive Integration Guide*; search for 'Socket API, configuring'.

NBA policy

When the Quarantine Manager releases an email that was quarantined by the NBA, the released email is routed through the NBA a second time on its way to the intended recipient. Likewise, 'quarantine notification' emails may also be routed through the NBA if the original email was sent using a web-based service such as Hotmail.

To ensure that released emails and notification emails can pass through the NBA without being reprocessed (and quarantined again!), edit the NBA policy so that it can recognize emails that are sent by the Quarantine Manager and the Socket API. To do this, edit the <enterprisednslist> policy tag in nbapolicy.xml so it specifies the following domains:

- The Domain of the Quarantine Manager host machine. This ensures that emails released from quarantine are ignored when they pass through the NBA, and
- The Domain that is specified by the Socket API registry value, SMTPDNSHostName. This ensures that 'quarantine notification' emails are ignored when they pass through the NBA.
- <enterprisednslist> is described in the section for Settings tags.

For an overview of how the NBA is able to ignore notification emails, see NBA ignores already-processed emails.

User policy triggers

Edit the user policies for your ordinary users, setting up the Outgoing Email control triggers and a control action to detect unauthorized emails and webmails, and apply a 'Quarantine Quietly' or 'Quarantine With Notification' control action.

In the control trigger, edit the 'Message to Users' to indicate that the sent email has been quarantined. Otherwise users may attempt to send the email again, believing the email has been blocked.

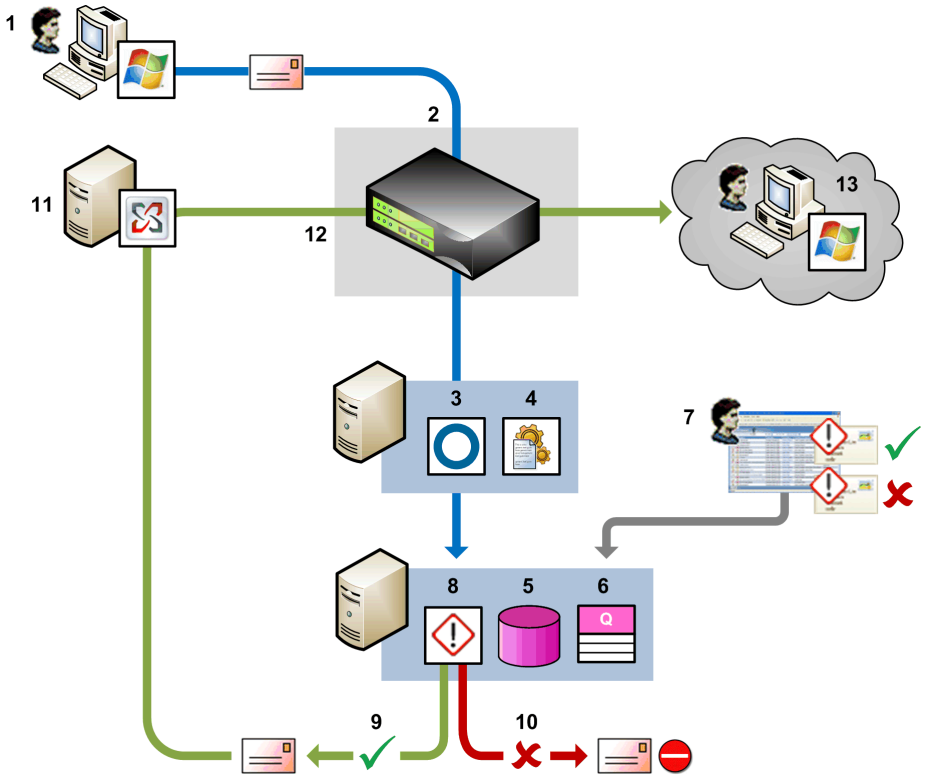
More information:

Settings Tags (see page 169)
NBA Ignores Already-Processed Emails (see page 95)
User Policy Changes (see page 137)

Deployment Architecture: NBA and Quarantine Manager

In this example, a Webmail is sent to an external Hotmail recipient. The NBA detects the Webmail and passes it to a policy engine, which applies the External Sender's policy. When the Webmail is quarantined, a reviewer can either release it or reject it. If released, the email is forwarded by the Quarantine Manager, via an SMTP server and the NBA, to its intended recipient.

Note: For simplicity, this diagram omits 'quarantine notification' emails sent by the Socket API.



Example: NBA and quarantined emails

A user sends an email (**1**) to a Hotmail recipient. The NBA (**2**) detects the email and passes it via the Socket API (**3**) to a policy engine (**4**). The policy engine applies a 'Quarantine Quietly control action and calls back to the NBA to block the associated data packets. It then replicates the email to the CMS.

The CMS (**5**) maintains a queue of quarantined emails (**6**). A reviewer checks this queue in the iConsole (**7**) and either releases or rejects the email.

The Quarantine Manager (**8**) regularly checks this queue and forwards released emails (**9**) to the intended recipients. Emails rejected by the reviewer are not forwarded (**10**).

Emails released from quarantine are sent via an SMTP server (**11**) through the NBA (**12**) on their way to the Hotmail server (**13**). The NBA detects that the email has already been processed by CA DataMinder and does not quarantine it again. The recipient sees the email when they next view their Hotmail inbox.

NBA Ignores Already-Processed Emails

The NBA does not reprocess emails that have already been processed by CA DataMinder. These include notification emails generated by the Socket API, emails released from quarantine, and emails processed by CA DataMinder email server agents.

How Does this Work?

1. **Domain written to custom header**

   When the NBA detects an email or Webmail, it checks for the presence of a custom header containing details about the originating domain and the 'policy processed' status. This header is present in:

   - 'Blocking notification' and 'quarantine notification' emails generated by the Socket API. The originating domain is specified by the SMTPDNSHostName registry value on the Socket API host machine.
   - Emails released from quarantine by the Quarantine Manager. The originating domain is hard-coded to be the domain of the Quarantine Manager; it cannot be changed.
   - Emails already processed by the Exchange or Domino server agents. The originating domain is specified by the SMTPDNSHostName registry value on the Policy Engine machine.
   - Emails already processed by the Milter MTA agent. The domain is set by the smtp-dns-hostname setting in the wgnmilter.conf configuration file.

2. **Domain checked against NBA domain list**

   If it detects the custom header in an email or Webmail, it compares the domain details with the domain list specified by the Enterprise Mail Server DNS List setting on the Policy tab.

   If the domain in the header matches an item in the domain list, the NBA then checks the 'policy processed' status. For the emails types listed in step 1 above, the status tells the NBA to ignore these emails.

More information:

Settings Tags (see page 169)

Example

The corporate network has a unipraxis.com domain.

1. The <enterprisednslist> tag in the NBA policy includes unipraxis.com. That is, the NBA policy is configured to ignore emails already processed in this domain.
2. A sender uses a Web-based service to send a Webmail to an external recipient.
3. The NBA quarantines the Webmail as it leaves the unipraxis.com corporate network.
4. The Socket API sends a 'quarantine notification' to the sender's Webmail address.

   Because SMTPDNSHostName is set to unipraxis.com, this domain is written to a custom header in the notification email. This allows the notification to pass through the NBA without intervention on its way to the sender's Webmail address.
5. Likewise, when the Webmail is finally released from quarantine, the Quarantine Manager writes unipraxis.com to a custom header in the released email. This allows it to pass through the NBA without intervention on its way to the intended recipient.

Blocked Web Pages and File Uploads: Notifying Users

When the NBA prohibits a Web page or file upload, or when these events are blocked following analysis by a policy engine, the NBA notifies the user.

This section refers to communications that use the HTTPURL and HTTPPOST protocols. You configure the NBA to detect these protocols by editing the <protocols> application filter tag and the <objtypes> configuration tag.

Default User Notifications

The NBA is designed to automatically notify users when a Web page or file upload is blocked. By default, it displays a pop-up dialog containing an explanatory message. An example dialog is shown below:

The only configuration required by you is to define the text content of the notification message. How you do this depends on the NBA application filter. If the filter is configured to:

- **Prohibit** the Web page or file upload, the message text is defined by the <prohibititle> and <prohibitmessage> tags in NBA policy; see <prohibitmessage>. The notification text is always the same, regardless of the item being blocked.
- **Analyze** the Web page or file upload, and subsequent processing by a policy engine results in a Block control action, the message is defined by the 'Dialog Title — Blockings' and 'Message To Users' settings in CA DataMinder user policy. The notification text depends on which Data In Motion trigger fired.

More information:

Notifications for Blocked File Uploads, Web Pages and Webmails (see page 140)

Customizing User Notifications

By default when a Web page or file upload is blocked, the NBA displays a pop-up notification dialog. But you can instead display an HTML file in the user's browser or you can redirect the user to an alternative URL such as a page on the corporate intranet. (The notification method is determined by the content of an HTML notification template.)

More information:

Display a Pop-up Notification Dialog (see page 98)
Display an HTML Page in Browser (see page 98)
Redirect Users to an Alternative URL (see page 98)
Notification Templates (see page 99)

Display a Pop-up Notification Dialog

To display a pop-up notification dialog, you add a JavaScript script to the notification template. This script also specifies the notification title and body text. When you edit the script, you can add the notification text directly, or you can use variables:

```
%scriptmessagetitle%
%scriptmessagetousers%
```

This is the recommended notification method; using JavaScript reduces the risk that the notification message fails to display.

More information:

Template Variables (see page 99)

Display an HTML Page in Browser

To display an explanatory message in an HTML page, you simply specify the <title> and <body> tags. The <title> displays in the browser title bar. The <body> can contain any standard HTML content. You can add the required text directly (be aware that there is a 1,000 character limit) or you can use variables:

```
%messagetitle%
%messagetousers%
```

More information:

Template Variables (see page 99)

Redirect Users to an Alternative URL

To redirect users to an alternative URL, such as a page on the corporate intranet, you add a <meta> tag to the <head> of the notification template.

Notification Templates

If a file upload or Web page is blocked, the NBA uses an HTML template to notify the user. The template content determines the type of notification (for example, a pop-up dialog or a redirection to an alternate URL) and, if necessary, the message content.

By default, the NBA uses blocktemplate.html to display notifications in a pop-up dialog. You can edit this template or use a custom HTML template. To specify a custom template, you must edit the <htmlblocktemplate> tag in the NBA policy.

When you edit a template, you can add the message content or alternative URL directly or you can use variables (see below).

Note: You can only use a single template; you cannot specify separate templates for each application filter.

More information:

XML Syntax: nbapolicy.xml (see page 155)
Template Variables (see page 99)

Template Variables

NBA notification templates support separate variables for the title and message text. Using variables can make it easier to manage your notification messages.

Notification Title Variables

Use these variables to specify the notification title:

%messagetitle%

Use this variable in the <title> tag of your notification template.

%scriptmessagetitle%

Only use this variable if your template includes a JavaScript script to display the notification in a pop-up dialog.

If the NBA application filter is set to:

- **prohibit**, both variables are replaced with the title defined for the <prohibittitle> tag in the NBA policy. <prohibittitle> is described in the section for Settings tags.
- **analyze**, both variables are replaced by the title defined in the 'Dialog Title — Blockings' setting in CA DataMinder user policy.

More information:

Settings Tags (see page 169)
Notifications for Blocked File Uploads, Web Pages and Webmails (see page 140)

Notification Message Variables

Use these to specify the message text:

%messagetousers%

Use this variable in the <body> tag of your notification template.

If you do use this variable, we recommend that you enclose it within percent tags. This ensures that any line breaks defined in the 'Message To Users' setting are preserved in the NBA notification message.

%scriptmessagetousers%

Only use this variable if your template includes a JavaScript script to display the notification in a pop-up dialog.

If the NBA application filter is set to:

- **prohibit**, both variables are replaced with the message defined for the <prohibitmessage> tag in NBA policy. <prohibitmessage> is described in the section for Settings tags.
- **analyze**, both variables are replaced with the message defined for the 'Message To Users' setting in a Data In Motion trigger.

More information:

Settings Tags (see page 169)

Redirection Variables

Although there is no variable to explicitly redirect users to an alternative URL, you can use %messagetousers% in a customized template to specify the target URL.

If the NBA application filter is set to **prohibit**, %messagetousers% is replaced with a URL defined for the <prohibitmessage> tag in NBA policy (see the previous section).

Example Notification Templates

The following examples show templates configured to display a notification message in a pop-up dialog or as a page in the user's browser. The final two examples show templates configured to redirect users to an alternative URL.

Pop-up dialog

JavaScript is used to display the notification message in a pop-up dialog. Note that the default template blocktemplate.html uses this method and is shown below.

```
<html>
        <head>
        <script type="text/javascript">
        window.alert("%scriptmessagetitle%\n\n%scriptmessagetousers%");
        </script>
        </head>
        <body></body>
</html>
```

HTML file

The notification displays as a page in the user's browser. In this example, the message content is defined directly in the template.

```
<html>
        <head>
        <title>Unipraxis Advisory</title>
        </head>
        <body>
        <p>This Web site breaches corporate Internet usage guidelines.</p>
        </body>
</html>
```

HTML file with variables

The notification displays as a page in the user's browser. In this example, variables are used to specify the message content. Note the use of <pre> tags in the message body to preserve line breaks.

```
<html>
        <head>
        <Title>%messagetitle%</Title>
        </head>
        <body>
        <pre>%messagetousers%</pre>
        </body>
</html>
```

Redirection to alternative URL

In this example, an alternative URL is defined directly in the template.

You **must** include the http:// prefix in the URL to ensure that it is interpreted correctly by the browser. No other <title> or <body> text is needed.

```
<html>
        <head>
        <Title></Title>
        <meta http-equiv="refresh" content="0;URL='http://www.hr.unipraxis.com'"/>
        </head>
        <body></body>
</html>
```

Redirection to alternative URL

In this example, a variable is to specify the alternative URL.

In this situation, the <prohibitmessage> tag in the NBA policy or the 'Message To Users' setting CA DataMinder user policy **must** contain a URL that includes the http:// prefix.

```
<html>
        <head>
        <Title></Title>
        <meta http-equiv="refresh" content="0;URL='%messagetousers%'"/>
        </head>
        <body></body>
</html>
```

Blocked Webmails: Notifying Users

When the NBA prohibits a Webmail, or when these events are blocked following analysis by a policy engine, the NBA notifies the user. The notification method is almost identical to that used when blocking file uploads or Web pages.

This section refers to communications that use a Webmail protocol. You configure the NBA to detect these protocols by editing the <protocols> application filter tag and the <objtypes> configuration tag.

More information:

Default User Notifications (see page 103)
Customizing User Notifications (see page 103)
Notification Templates (see page 104)
Template Variables (see page 104)
Blocked Web Pages and File Uploads: Notifying Users (see page 96)

Default User Notifications

The NBA is designed to automatically notify users when a Webmail is blocked. By default, it displays a pop-up dialog containing an explanatory message.

As with blocked file uploads and Web pages, the only configuration required by you is to define the text content of the notification message.

If the NBA application filter is configured to:

- **prohibit**, the message text is defined by the <prohibititle> and <prohibitmessage> tags in NBA policy; <prohibitmessage> is described in the section for Settings tags.
- **analyze** and subsequent processing by a policy engine results in a Block control action, the message is defined by the 'Dialog Title — Blockings' and 'Message To Users' settings in CA DataMinder user policy. The notification text depends on which Outgoing Email trigger fired.

More information:

Settings Tags (see page 169)
Notifications for Blocked File Uploads, Web Pages and Webmails (see page 140)

Customizing User Notifications

By default when a Webmail is blocked, the NBA displays a pop-up notification dialog. But you can instead display an HTML file in the user's browser or you can redirect the user to an alternative URL. The notification method is determined by the notification template.

Display a pop-up notification dialog

To display a pop-up notification dialog, you add a JavaScript script to the notification template. This script also specifies the notification title and body text.

Display an HTML page in browser

To display an explanatory message in an HTML page, you simply specify the <title> and <body> tags in the notification template.

Redirect users to an alternative URL

To redirect users to an alternative URL, such as a page on the corporate intranet, you add a <meta> tag to the <head> of the notification template.

In all cases, the method used is the same as that used for blocked file uploads and Web pages.

Notification Templates

If a Webmail is blocked, the NBA uses an HTML template to notify the user. The use of templates and template variables is generally the same as for blocked file uploads and Web pages.

However, there is one important difference. Because of the diverse methods used by different Webmail services, it is not always possible for the NBA to use the template specified by <htmlblocktemplate> in the NBA policy.

If this happens, the NBA defaults to an internal template that is identical in structure to the default template blocktemplate.html. This internal template displays a pop-up notification dialog when a Webmail is blocked and uses variables to populate the title and message text.

Template Variables

The text sources for template variables are exactly the same as for blocked file uploads and Web pages, but with one difference.

If the NBA application filter is set to analyze, both %messagetousers% and %scriptmessagetousers% are replaced with the message defined for the 'Message To Users' setting in an Outgoing Email trigger.

Chapter 6: Decoding SSL Communications

The NBA supports SSL decoding. SSL prevents transmitted data from being intercepted by a third party and ensures that network transactions (such as web requests) are serviced by the intended network host (such as a web site).

SSL decoding is available in CA DataMinder Network 14.0. It also available for r12.5 if you have deployed fix #RO37161 (see page 34).

This section contains the following topics:

What is SSL?

The Secure Sockets Layer protocol (SSL) helps ensure that a network transaction (such as a web request) is only serviced by the intended network host (such as a web site). SSL also prevents transmitted data from being intercepted by a third party. The connection does this by encrypting the traffic using public/private key encryption.

- You obtain a public key via a certificate which is validated against a trusted certificate authority.
- Each client holds a well-known public certificate of an organization that it trusts (the certificate authority). The client then requests the certificate of the server that it needs to connect to. If the server's certificate is correctly signed by a trusted certificate authority, the client proceeds with the connection and negotiates the encrypted communications channel.

Typical SSL applications include online purchasing and webmail, and an increasing number of web sites and applications (such as instant messaging). In particular, the widespread use of social networking sites is a major cause for concern regarding data loss. Your ability to analyze data transmitted from your company network to these external networks is increasingly important.

About Certificates

A certificate is a small file containing data about a website or network host. The certificate is signed to prevent falsification and contains a chain of responsibility (the certification path) that allows a browser or network client to verify the certificate even if the browser or client only has local access to the top-level (or root) certificate in the chain.

Web browsers provide the ability to view the certificate of a website and verify that the certificate is valid. Browsers ship with, and regularly update, a set of Certificate Authority certificates to help ensure that verification can be performed.

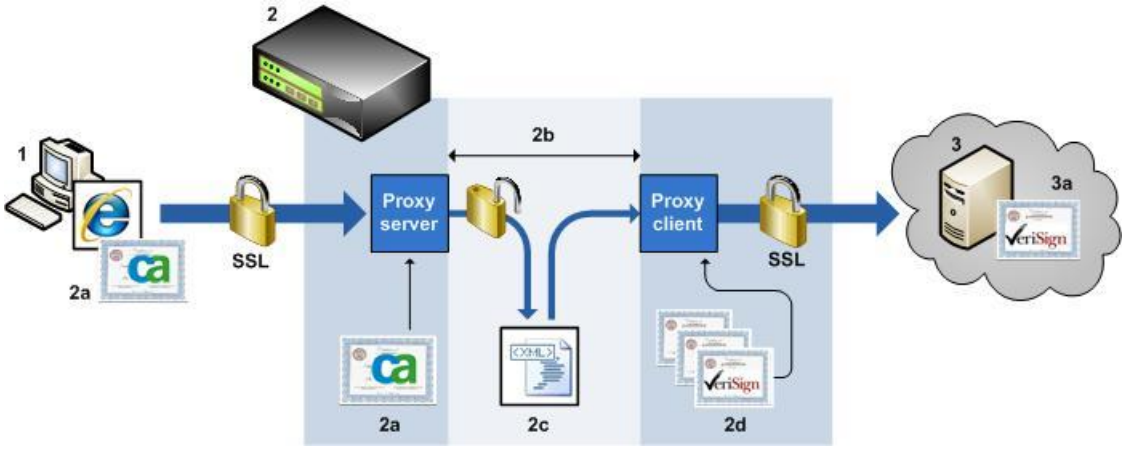
How Does the NBA Decode SSL Traffic?

Because SSL is designed to prevent third parties intercepting data, the only way to access the content of an SSL communication is to implement a 'man-in-the-middle' proxy. To do this, the NBA is connected between a client computer and the internet-based web server. A proxy server on the NBA intercepts and terminates client requests to open SSL sessions. A proxy client on the NBA then creates a second SSL session to the web server.

The trust relationship between the client and web server now becomes one trust relationship between the client and NBA, and a second trust relationship between the NBA and web server. To establish this trust, you generate a 'master' trusted root certificate on the NBA. You must then install this master certificate onto each client so that the client trusts the NBA. At the same time, the NBA holds a set of root certificates, similar to those installed in most browsers, enabling it to verify the connection between itself and the web server.

When the NBA intercepts an SSL communication, the NBA decodes the SSL communication and applies NBA policy. If NBA policy is configured to decrypt and analyze the communication, the NBA processes all data on that session just as if it were unencrypted, allowing CA DataMinder policy to be applied. This CA DataMinder policy processing returns an 'allow' or 'block' result. If the result is 'allow', the NBA re-encrypts the communication before forwarding it to the web server.

Decryption and re-encryption is performed by the decoder, a module within the NBA that incorporates a proxy server and proxy client.



NBA Uses Proxy Server and Client to Decode SSL Traffic

A client application (**1**), such as a web browser or email client, attempts to connect to a target server application, such as a web server or SMTPS server (**3**). The client holds a copy of the NBA master certificate (**2a**).

The proxy server on the NBA (**2**) intercepts the client request to open an SSL session. At the same time, the proxy client on the NBA creates an SSL session to the target web server (**3**). The target server issues a certificate signed by a certificate authority (**3a**). The NBA holds a store of common root certificates from certificate authorities (**2d**) and uses one of these root certificates to verify the connection to the target server.

The NBA then returns a certificate to the client using details from the target server's certificate (**3a**). The NBA signs this certificate with its own master certificate (**2a**). The client then uses its NBA master certificate to verify the connection to the NBA.

Having established an encrypted link between the client and the target server, the NBA proxy server decodes the SSL communication (**2b**). Decoded data is passed to NBA policy filters for decryption and analysis (**2c**). When analysis is complete, and policy processing returns an 'allow' result, the proxy client re-encrypts the communication and forwards it to the target server application.

Network Configuration

SSL traffic is intercepted transparently by the NBA.

The NBA therefore does *not* require proxy IP addresses to terminate SSL sessions, DNS entries, browser proxy configuration, or additional firewall configuration.

Decoded Network Protocols

The NBA can decode the following protocols:

HTTPS (GET and POST)

This includes Web forms and documents downloaded from web sites.

WEBMAIL over HTTPS.

GMail and Hotmail are SSL-enabled and the NBA can decode them even if the browser uses HTTPS.

SMTPS.

This includes secure email transmitted either using the STARTTLS protocol extension or without using STARTTLS.

SSL 3.0, SSL 3.1, TLS 1.0, TLS 1.1, TLS 1.2.

Note that the NBA does not decode the old SSL 2.x protocol. Verify that SSL 2.x is disabled by default in the client's browser.

Hardware Acceleration with Cavium Devices

(Bivio 7000 appliances only)

If a Bivio 7000 appliance is fitted with Cavium CN1615 Security Processor cards, the NBA offloads the computationally expensive cryptographic operations to the Cavium device, freeing the Bivio CPUs from these operations and thereby maintaining high performance.

The NBA console on Bivio 7000 appliances contains an SSL Decode Statistics page which shows whether the hardware or software decoder is being used.

More information:

Verify that the Cavium Driver is up to Date (see page 45)

What Does the User See?

When a user browses to a secure site, they see a change in the browser address bar. The protocol part of the URL changes to 'HTTPS' and a padlock or shield icon usually appears. If the trusted 'master' certificate of the NBA is installed on the client machine, the user sees no change in the look or behavior of the site. If the user clicks the padlock or shield icon, they can view the certificate of the web site. This certificate looks very similar to the original certificate from the web site, except that the certificate signer will be 'CA DataMinder' (you can customize this name).

Invalid Certificates

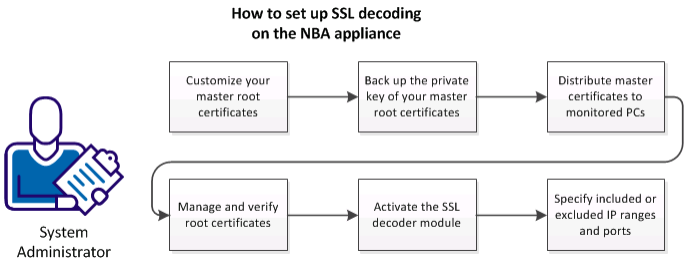
If the user browses to an HTTPS site that presents an invalid certificate, the address bar may turn red and the main content window shows a warning message. This message permits the user to ignore the warning and continue browsing to the site. The same happens if the NBA is decrypting the network traffic. The error in the site's certificate is replicated in the certificate supplied by the NBA to the client's browser.

Extended Validation (EV) or High Assurance certificates

Extended Validation (EV) or High Assurance certificates are issued to web sites by certain Certificate Authorities. They are issued to sites only after additional checks to verify the identity of the owner of the site. The browser contains a list of these Certification Authorities. If the root certificate of a particular web site is in the list, the address bar in the browser turns green and shows the Certificate Authority name. When using the NBA SSL decoder, such sites are decoded correctly but displayed with the standard browser address bar (with a white background).

How to Set up SSL Decode

External web sites and applications (such as instant messaging and social networks) using SSL are a major cause for concern regarding data loss. You as a system administrator want to set up CA DataMinder to be able to decode and monitor SSL encrypted data passing through your network.



To set up SSL decoding, you need to:

1. Customize your master root certificates (see page 111).
2. Back up the private key of your master root certificates (see page 115).
3. Distribute the master certificates (see page 116) to all computers in your organization for which you want the ability to decrypt SSL communications.
4. Manage the root certificates (see page 118) on your NBA appliance, and verify that the set of root certificates is up-to-date.
5. Activate the SSL decoder module (see page 120) within the NBA.
6. Specify the IP ranges (see page 121) and ports that you want to include or exclude from SSL decoding.

These tasks are described in the following sections.

Customize the Master Certificate

When CA DataMinder Network is installed, two master root certificates are created, a trusted certificate and an untrusted or revoked certificate. These are the NBA master certificates. Throughout this guide, the term 'master certificate' refers to these root certificates generated by the NBA.

To permit CA DataMinder Network to decode SSL communications sent by a client, the client *must* trust the master certificate. You establish this trust by distributing the NBA master certificate to all client machines in your organization where you want the ability to decrypt SSL communications. This process is essential for SSL decoding.

But first, you must customize the master root certificates with your preferred certificate details.

Why customize the master certificate details?

If the trusted master certificate of the NBA is installed on the client, the user sees no discernible difference when browsing to web sites over SSL connections. But if they examine the certificate behind the connection (by clicking the padlock icon in the browser address bar), they see the certificate signing authority. For example, if they browse to the GMail website they can see that the www.google.com website is verified by VeriSign.

By default, however, the NBA master certificates use 'CA DataMinder' as the common name. So if the NBA intercepts an SSL connection to the GMail web site, the user would see that www.google.com is verified by CA DataMinder. We recommend that you change this name to the name of your organization.

More information:

Trusted and Untrusted Master Certificates (see page 112)
Default Master Certificate Details (see page 113)
Generate Customized Master Certificates (see page 114)

Trusted and Untrusted Master Certificates

When you generate an NBA master certificate, two certificates are actually generated, one trusted and one untrusted.

The need for a trusted master certificate is self-evident; it ensures that clients will trust certificates that the NBA generates.

But why generate an untrusted certificate? Because the NBA must sometimes create a certificate that mimics the untrusted, or revoked, certificate provided by a real web site. That is, the NBA sometimes needs to create a certificate that a client's browser cannot trust. The NBA does this to force the browser to display a certificate error.

The NBA then uses the untrusted, or revoked, master certificate to sign certificates from SSL servers that the Network Appliance cannot trust. These include certificate revocation detection and other certificate signing errors. For example, the NBA uses the untrusted certificate to sign the certificate created when the NBA cannot determine a chain of trust from the certificate provided by a web site to a well known root certificate, or when a web site's certificate matches one in the list of revoked certificates on the NBA appliance.

If required, you can install the untrusted master certificate to your clients, adding it to an untrusted certificate list. This forces the client to display additional warnings to the user. However, note that such behavior depends on the client and browser.

Important: You must not install this untrusted or revoked master certificate in a browser's Trusted Root Certificates list.

Default Master Certificate Details

The default information included in the NBA master certificates is summarized below.

Common Name (or Subject)

Set to 'CA DataMinder Network'.

The Common Name is the most important certificate detail because this name is usually presented as the signing root authority when a user checks their SSL connection. The user does this by clicking the padlock icon in the browser address bar.

When you customize the NBA master certificates, set the common name to the name of your organization. You may also want to add a note explaining the purpose of the certificate and clarifying its origin.

Organization Name

Set to 'CA Technologies'.

Locality Name

Set to 'Islandia'.

Province Name

Set to 'NY'.

Country Name

Set to 'US'.

Validity Period (Days)

Defaults to 730 days (two years).

Generate Customized Master Certificates

Before you distribute the NBA master certificates to your clients, you must customize the certificate details. Do one of the following to customize and regenerate NBA master certificates.

To change the master certificate using the NBA console

1. Log on to the NBA console and go to the SSL tab.
2. Click the Master Certificates option.
3. Change the Common name, Organization, Locality, Province, Country and Validity Period settings as required.

   The Common Name is the most important because this name is usually presented as the signing root authority when a user checks their SSL connection. They do this by clicking the padlock icon in the address bar of their browser.

   Use your organization's name as the Common Name to make the origin of the certificate clear. You may also want to add a note explaining the purpose of the certificate.
4. Click Generate.
5. The console displays a warning that the new public certificate and private key pairs will become active immediately and overwrite the current public certificate and private key pairs.
6. Type ‘confirm’ in the input box and click Generate.

   The NBA generates the Trusted certificate and Revoked certificate and saves them on the Network appliance. (The Revoked certificate is optional.)

To change the master certificate using FTP

1. Using FTP, browse to the /config folder on the NBA appliance.
2. Edit nbaconfig.xml and change the following lines:

```
<commonname type="stringType" value="CA DataMinder Network"/>
<organizationname type="stringType" value="CA Technologies"/>
<localityname type="stringType" value="Islandia"/>
<provincename type="stringType" value="NY"/>
<countryname type="stringType" value="US"/>
<validityperioddays type="numberType" value="730"/>
```

   The <commonname> is the most important setting because this name is usually presented as the signing root authority when a user checks their SSL connection. They do this by clicking the padlock icon in the address bar of their browser.

   Use your organization's name as the Common Name to make the origin of the certificate clear. You may also want to add a note explaining the purpose of the certificate.

3. Log on to the NBA console using SSH.
4. Run this command to prepare the NBA command environment:

   ```
   . /usr/local/share/nba/nbarc
   ```

   Note: Do not omit the space between the period and the first slash.
5. Change into the NBA executable directory:

   ```
   cd /home/nba/bin
   ```
6. Run this command to generate the new master certificate:

   ```
   ./nbacmd SSL_GENERATE
   ```

   This generates the following output:
   ```
   2010/12/23 11:26:43.963997 CMD: SSL certificate regeneration completed.
   ```
7. Using FTP, browse to the /config folder on the NBA appliance.

   The nbaroot (trusted) and nbarevoked (untrusted) certificates are available for download in both .p7b and .crt formats.

Back up the Private Key

The private key of the NBA's master certificate (used by clients to verify each SSL connection) is stored on disk on the NBA appliance. You must copy the private key for backup purposes. You must also copy the private key if the same certificate details are needed on multiple NBA appliances in a failover or load sharing configuration.

Note: Backing up the /config folder only protects the NBA configuration settings. It does not back up the private key.

To copy the private key

1. Log on as root to the NBA console using SSH.
2. Go to the private key directory. To do this, run this command:

   ```
   cd /home/nba/bin/private
   ```

   This folder contains the following files:

   nbaroot.crt

   Public key in base64 X509 format

   root.pem

   Private key

   nbarevoked.crt

   Public key in base64 X509 format

   revoked.pem

   Private key
3. Copy the complete /private folder, including these files, from the configured CA DataMinder Network appliance to the unconfigured appliances.
4. Copy the complete /home/smb/config folder, including all subfolders and files, from the configured CA DataMinder Network appliance to the unconfigured appliances in order to fully replicate your configuration.
5. Reboot the unconfigured appliances.

Distribute the Master Certificates

The NBA master certificates that the NBA uses to sign server certificates must be available for the client application to use.

The client application is usually a browser running on a user's computer. You therefore need to download and distribute the master certificates to all computers in your organization for which you want the ability to decrypt SSL communications.

In most cases, the browser used is Internet Explorer. You can update the certificate store for Internet Explorer using Windows Group Policy. Other browsers and applications have their own certificate stores, so you must identify and update those stores before you enable SSL decoding.

Important: You must distribute the master certificate! If you do not, client applications may fail to make SSL connections because they will be unable to validate the certificate returned by CA DataMinder Network. Internet Explorer displays the generic message 'Internet Explorer cannot display the web page' and may not explicitly warn you about a certificate error.

More information:

Download the Master Certificates (see page 117)
Install the Master Certificates (see page 117)

Download the Master Certificates

To export the master certificate using the NBA console

1. Log on to the NBA console and go to the SSL tab.
2. Click the Master Certificates option.
3. Go to CA DataMinder Network Root Certificate Download section.
4. Click Export to download the certificate you want.

   You can separately download the Trusted and Revoked certificates.
5. Choose Save in the File Download dialog and specify the target folder for the downloaded certificate.

To export the master certificate using FTP

1. Using FTP, browse to the /config folder on the NBA appliance.

   The nbaroot (trusted) and nbarevoked (untrusted) certificates are available for download in both .p7b and .crt formats.
2. Copy the certificates to your preferred location.

Install the Master Certificates

To install a certificate for Internet Explorer or Chrome using Group Policy

The actual steps vary, depending on your operating system. In summary, you must:

1. Assign a new Public Key Policy to your domain.
2. Locate the nbaroot.p7b file that you downloaded from the NBA appliance.

   This file contains the trusted NBA master certificate.
3. Import the NBA master certificate into Trusted Root Certification Authorities.

To install a certificate for Firefox

1. Locate the nbaroot.crt file that you downloaded from the NBA appliance.

   This file contains the trusted NBA master certificate.
2. Click Tools, Options, Advanced, Encryption, View Certificates, Authorities, Import.

   This command imports the file into Firefox.
3. Choose 'Trust this CA to identify web sites.'

Manage the Root Certificates

The NBA holds a set of well-known root certificate authority certificates that permit the NBA to validate connections to target websites. However, certificate authorities sometimes withdraw certificates and issue new ones, so you must keep the set of root certificates up to date on the NBA appliance. You may need to add or remove certificates from this set and if any public certificates are revoked, you must add them to the NBA's revocation list.

Status information for all the certificate files is recorded in two log files on the NBA:

- sslcertstatus.txt records the status of certificates being used by the NBA's SSL decoder.
- console_sslcertstatus.txt records the status of certificates listed in the NBA console.

There are two methods for updating the certificate lists.

To manage root certificates using the NBA console

1. Log in to the NBA console and go to the SSL tab.
2. Click the Root Certificates option.
3. You can add, remove or download trusted root certificates and revoked root certificates. Do one of the following:

   Add new certificates.

   Click Import to add new certificates.

   Then browse to the file containing the certificates that you want the SSL decoder to use. A certificate file can contain multiple certificates.

   Finally, click Import to add the selected file.

   Remove one or more certificates.

   Click Delete.

   Then hold the Ctrl key down while selecting one or more certificates to remove.

   Finally, click Delete to remove the selected certificates.

   Download a certificate file

   Click Export to download a file containing *all* certificates in the list.

   You can import this file onto another NBA to keep the certificate sets identical on multiple NBA appliances.

   Reset the certificate list

   Click Reset to remove all current certificates and replace them with the certificates delivered on installation.

To manage root certificates using FTP

1. Using FTP, browse to the NBA /config/rootcerts folder.
2. Add, remove, or copy the certificate files to maintain the set.
3. To make the NBA use the modified set of certificates in this folder:
   a. Log on to the NBA console using SSH.
   b. Prepare the NBA command environment with this command:

      ```
      . /usr/local/share/nba/nbarc
      ```

      Note: Do not omit the space between the period and the first slash.
   c. Change into the NBA executable directory:

      ```
      cd /home/nba/bin
      ```
   d. Update the NBA SSL Decode configuration with this command:

      ```
      ./nbacmd SSL_UPDATE
      ```
   e. The following output confirms successful operation:

      ```
      2010/11/26 15:55:32.653788 nbaSendEvent: Event system connected
      2010/11/26 15:55:37.679308 CMD: SSL certificate regeneration completed.
      OK
      ```

Root Certificate Formats

Certificates are downloaded in .p7b format, which allows multiple certificates to be handled in one file using the Microsoft Windows MMC Certificates plug-in. When uploading certificates to the Network appliance, use this format or alternatively use .cer/.crt/.key/.pem files, which are either base-64 or DER encoded.

Certificate revocation lists (.crl files) are downloaded as a gzipped tar file (.tar.gz). When uploading certificate revocation lists to the Network appliance, you must upload each .crl file individually. The revocation lists may be either base-64 or DER encoded and contained in .crl or .pem files.

Activate the Decoder

The SSL decoder is a module within the NBA that decrypts intercepted SSL traffic and then re-encrypts the communication when policy processing is complete. For the SSL decoder to operate, the NBA must be online and in active mode, and network filters in the NBA policy must be set up for packet decryption.

Before you enable SSL decoding

1. Verify that master certificates from the NBA SSL decoder are distributed to all clients where you want to decode network traffic.

   Use Group Policy or your preferred client administration tool to install the master certificates.

2. Review the SSL network traffic that you expect to see on the network segment of the NBA.

   The NBA must be configured with details of sessions to include or exclude from decoding. For example, some instant messaging clients cannot be configured to accept the NBA master certificate, so they cannot be decoded and must be excluded.

Activating the SSL decoder

For the SSL decoder to operate, bring the NBA online and verify that it is in active mode:

- Install the NBA physically inline between the corporate LAN and the Internet.
- Switch on Packet Processing.
- Switch on Stream Blocking.
- Configure network filters in the NBA policy to enable packet decryption for selected IP addresses, or port numbers, or both. The NBA policy contains a preconfigured network filter to start the SSL decode of HTTPS traffic. The filter is named "Default SSL decryption".

  To enable SSL decode, do *one* of the following:

Enable SSL decode using the web UI

1. Select the 'Filters' tab and browse to the 'Network (packet) filters' section.
2. Select the Enable checkbox for the "Default SSL decryption" filter.
3. Click 'Apply'.

   The NBA reloads the policy and activates the network filter.

Enable SSL decode using FTP

1. Edit the file /config/nbapolicy.xml
2. Change enabled=false to true in the networkfilter element:

```
<networkfilter enabled="true">
    <filtername type="stringType" value="Default SSL decryption"/>
    <ipaddrlist type="stringListType">
        <element value=":80"/>
        <element value=":443"/>
    </ipaddrlist>
    <protocols type="stringListType">
        <element value="tcp"/>
    </protocols>
    <action type="simpleEnumStreamBlock" value="decrypt"/>
    <loglevel type="simpleEnumLogLevel" value="error"/>
</networkfilter>
```

3. Save the file.

   The NBA reloads the policy and activates the network filter.

More information:

Active (Inline) Mode (see page 13)

Include or Exclude IP Ranges from SSL Decoding

Included IP Ranges and Ports

You must configure the NBA to decode SSL traffic using specific IP ranges and ports. You specify these IP ranges and ports when you set up your network filters. Any SSL traffic using other IP addresses or ports is not decoded.

We recommend that you target SSL decoding at IP address ranges and ports where you expect to see SSL traffic that can be decrypted. For example, many client computers use port 80/443 for HTTPS and port 25/465/587 for SMTPS, so you need to target these address IP ranges and ports. Other applications may use different ports. For example, Forefront TMG, a Microsoft threat management product, uses ports in the range 25,000 to 50,000. Your network administrators can provide you with the IP ranges and ports that you must target when decoding encrypted SSL traffic.

You may also want to specify IP addresses or port numbers where SSL traffic is not typically expected but where you need to detect any SSL traffic that does occur.

Excluded IP Ranges and Ports

After choosing which IP ranges and ports you want to monitor for SSL traffic, you can exclude certain addresses or ports from decoding. For example, if you have included SSL traffic from IP range 10.20.0.0/16, you can exclude SSL traffic from a specific address within this range, such as 10.20.0.12.

Exclusions from SSL decoding are necessary when:

- You do not need to analyze the network traffic as a source of potential data loss.
- The NBA cannot decode a particular protocol, so there is no benefit in flagging it for decryption.
- The SSL connection uses client certificates and server certificates. The NBA does not decrypt this traffic.
- The SSL connection uses no certificates at all. The NBA does not decrypt this traffic.
- The NBA master certificate cannot be installed in the client to allow it to trust SSL sessions terminating at the NBA.

Excluded Domains

You can also specify exclusions based on DNS names. If an SSL connection is made to a server with a matching domain name, the connection is not decoded.

More information:

Include an IP Range (see page 123)
Exclude an IP Range (see page 124)
Exclude a Domain (see page 124)
Define Exclusion Caching Controls (see page 125)

Include an IP Range

To decrypt SSL frames on specific IP address and port ranges for HTTPS web traffic, you set up network filters.

To include IP addresses

1. Log on to the NBA console and go to the Filters tab.
2. Create a network filter.
3. Specify the following parameters.

   IP Addresses

   Specify the IP range and port that you want to include. For example:

   10.20.0.0/16:443

   Protocols

   TCP

   Action

   decrypt

   This filter decrypts addresses in the range from 10.20.0.0 to 10.20.255.255, on port 443.
4. Configure the application (stream) filter.
5. Specify the following parameters.

   IP Addresses

   Set this value to '*'

   Protocols

   HTTP

   Action

   analyze

   This filter uses the asterisk wildcard to include all remaining addresses in the analysis after the network filter has prefiltered them.

Exclude an IP Range

To exclude IP ranges from decryption, add network filters for IP addresses.

To exclude servers by IP address:

1. Log on to the NBA console and go to the Filters tab.
2. Create a network filter.
3. Enter the following parameters.

   IP Addresses:

   Enter the IP range that you want to exclude, for example:

   10.20.0.12:443

   Protocols

   TCP

   Action

   ignore

   The excluded IP address in this example has a smaller range than the included IP addresses. The excluded IP range therefore takes precedence over the included IP range. Consequently, SSL traffic from this address is ignored and not decrypted.

Exclude a Domain

You can also specify exclusions based on DNS names. If an SSL connection is made to a server with a matching domain name, the connection is not decoded.

Default List of Excluded Domains

The NBA is prepopulated with a default list of excluded domains. These domains are the addresses of Windows Update and Activation servers. We recommend that you add antivirus and other infrastructure management connections to this list. Or add an IP address filter or port number filter (or both) to exclude these sessions from SSL decoding.

How Are Domains Verified?

If you have an RFC2817 HTTP CONNECT proxy that browsers use to connect to secure web sites, and your NBA appliance is between the clients and the proxy, the NBA appliance identifies the destination domain for each connection. If the domain is excluded, SSL connections to the domain are allowed to proceed without decoding.

For connections that do not go through a proxy, the NBA compares domains against the "Subject" or "Issued to" property of the SSL certificate. The first connection to the domain is closed and subsequent connections are allowed to proceed without decoding.

Subdomains

Subdomains of excluded domains are also excluded. For example, if the excluded domain is "company.com" but the website is "special.company.com", then the subdomain is still excluded.

To exclude domains using the NBA console

1. Log on to the NBA console and go to the SSL tab.
2. Click the General option.
3. Add the domains to the excluded domains list.

To exclude domains by editing nbapolicy.xml

1. Open nbapolicy.xml.
2. Locate and edit the following elements:

```
<domainexcludelist type="stringListType">
  <element value="update.microsoft.com"/>
  <element value="download.microsoftupdates.com"/>
  <element value="activation.sls.microsoft.com"/>
  <element value="windowsupdate.microsoft.com"/>
</domainexcludelist>
```

Define Exclusion Caching Controls

The NBA console supports client and server exclusion caching. If you are not using a web proxy, you can enable these caches to simplify the process of excluding certain traffic from SSL decoding. In particular, you can allow 'failed connection' SSL sessions to pass through the NBA without decoding them and without needing to specify a domain exclusion or an excluded IP range.

Server exclusion caching

When enabled, the server exclusion cache allows unmonitored sessions to SSL servers that do not accept connections from the SSL decoder.

This can happen if the decoder's SSL protocols are unacceptable to the server. A client must attempt to connect to the server before the decoder can determine this, so only subsequent connections are permitted. The IP address and port number of the server are cached so that future connections to this server and port are excluded from SSL decoding.

Note: If you have a web proxy or similar device between the decoder and the internet that hides the real server's IP address from your internal network, you cannot use the server exclusion cache and must disable it. This is because all web servers will appear to have the same IP address, so connections to all web servers will be excluded from SSL decoding.

Client exclusion caching

When enabled, the client exclusion cache allows unmonitored sessions from clients that fail to connect to the SSL decoder on the NBA.

This can happen if the client does not have the NBA master certificate installed. The IP addresses of both server and client as well as the port number of the server are cached, so that future connections from this client to the server are excluded from SSL decoding.

Note: Be aware that some client applications do not cause the SSL negotiation error needed to trigger the caching. Instead, they simply close the connection after it has been negotiated.

Note: If you have a web proxy or similar device between the clients and the decoder that hides the real client IP addresses from the NBA, you cannot use the client exclusion cache and must disable it. This is because all clients will appear to have the same IP address, so connections from all clients will be excluded from SSL decoding.

To control exclusion caches using the NBA console

1. Log on to the NBA console and go to the SSL tab.
2. Click the General option.
3. Click the Enable/Disable button for the required cache.

To exclude domains by editing nbapolicy.xml

1. Open nbapolicy.xml.
2. Edit the following elements:

```
<serverexclusioncache type="booleanType" value="false"/>
<clientexclusioncache type="booleanType" value="false"/>
```

Where

value="false" disables the cache.

value="true" enables the cache.

SSL Statistics

(Bivio 7000 appliances and the Linux Server Platform only)

Optionally, you can view the SSL statistics for each CPU on the SSL Statistics page.

To view SSL statistics

1. Log on to the NBA console and go to the SSL tab.
2. Click the Statistics option.

The following statistics are available.

Decoder State

Shows the state of the SSL Decoder. This can be Hardware, Software, or Disabled. The normal states are Hardware or Software.

- **Hardware:** Shown if SSL streams are being processed and the NBA appliance is fitted with SSL acceleration co-processors.
- **Software:** Shown if SSL streams are being processed but the NBA appliance does not have SSL acceleration co-processors or they have been manually disabled.

  Note: You can manually disable the co-processors. To do this, write a file named 'disablesslcoprocessor' to the /NBA /config folder (this file does not need any content) and then restart the NBA.

  Disabled: Shown if SSL streams are not being processed. Examine the log file to identify the relevant CPU in order to determine the cause of the problem.

Active Sessions

Shows the number of SSL sessions in progress.

If you take the Network appliance offline or change filter settings that affect SSL decoding, SSL sessions may be disconnected. Therefore, you must only perform these actions when the number of active sessions is zero.

Total Sessions

Shows the total number of SSL sessions that have been decoded.

This count includes attempts to decode a session that later fails due to a certificate negotiation problem.

Excluded Sessions

Shows the total number of SSL sessions that have been excluded from SSL decoding. Data on these sessions cannot be analyzed.

Sessions are excluded when the domain name of the connection matches an entry in the excluded domains list.

Sessions are also excluded if a decoder connection failure causes the connection to be excluded from decoding. Such exclusions are only permitted when client or server exclusion caching is enabled or if the domain in a certificate matched a domain in the exclusion list.

Dropped Frames

The dropped frames count shows the number of frames that have been dropped by the NBA in response to a flow-control event. The NBA uses the TCP window mechanism to try and limit the amount of data it needs to buffer for each SSL session.

If the client/server doesn't react to the reduced TCP window quickly enough, the NBA drops frames on that connection and expects the client/server to resend these frames when the TCP window is restored.

Decrypted Frames

Shows the total number of network frames containing data that has been decrypted.

Decrypted Records

Shows the total number of SSL records that have been decrypted.

A network packet can contain multiple SSL records and an SSL record can be spread across multiple network packets. The hardware-accelerated decoder is much more efficient when SSL records are large and spread across multiple network packets, but SSL record size is controlled by the client and server using the SSL connection.

Decrypted Bytes

Shows the total number of bytes decrypted.

Cached Trusted Certificates

Shows the number of SSL certificates in the trusted cache.

Creating a certificate for a connection is time-consuming for the SSL decoder, and fetching a single web page may create many connections. To make the decoding more efficient, the NBA caches trusted certificate details for up to one hour.

Cached Untrusted Certificates

Shows the number of SSL certificates in the untrusted cache.

The NBA creates an untrusted certificate to mimic the untrusted certificate provided by the web server. The comment field in the untrusted certificate gives the reason why the decoder does not trust the web server certificate. For example, there may be a validity date problem or a problem with the root certificate used by a web site's certificate.

To make the decoding more efficient, the NBA caches untrusted certificate details for up to one hour.

Cached Excluded Sessions

(Not displayed in the NBA console. This statistic is included in the statistics log file (see page 205) for diagnostic purposes.)

Shows the number of SSL sessions in the exclusion cache.

If a certificate negotiation failure causes the connection to be excluded from decoding, the NBA caches session details so that future SSL connections are permitted without decoding them. Such exclusions are only permitted when client or server exclusion caching is enabled or if the domain in a certificate matched a domain in the exclusion list.

Cached Included Sessions

(Not displayed in the NBA console. This statistic is included in the statistics log file (see page 205) for diagnostic purposes.)

Shows the number of cached SSL sessions that need decoding.

If HTTP web traffic is directed from client machines via the NBA and then through an RFC2817 web proxy, the NBA decodes the HTTP CONNECT requests to discover the requested domains. When the connection transitions to SSL, the NBA compares the requested domain to the list of excluded domains. If the requested domain is:

- Excluded, the NBA allows the session to go through without decoding it.
- Not excluded, the NBA decodes the session identified from the cached session details.

Chapter 7: Applying User Policy to NBA Events

This chapter describes how CA DataMinder identifies and assigns event participants to files and emails captured by or imported from the NBA. It also explains:

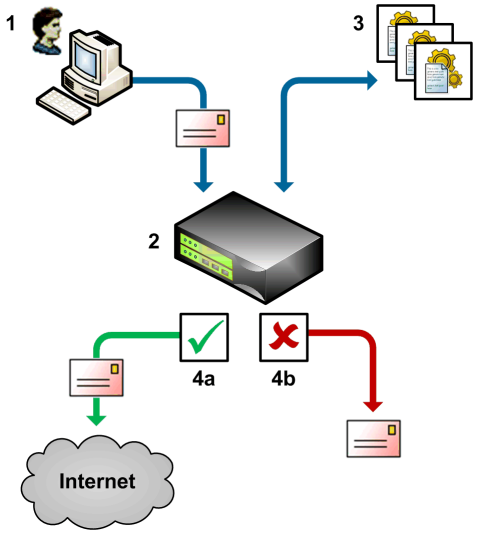
- How CA DataMinder determines which user policy to apply to those files and emails, and which policy triggers and control actions are applicable.
- How to set up user policies to identify external emails and save copies of captured files and emails on the CMS.
- How NBA blockings are perceived by end users and summarizes the underlying blocking mechanisms used by the NBA.
- The key settings in the machine policy for your policy engines. These settings determine which user policies are used to analyze files and emails.

This section contains the following topics:

Applying User Policy Overview

The NBA passes captured items to a CA DataMinder policy engine for analysis. The policy engine applies the appropriate user policy to the item. It then calls back to the NBA, instructing it block or allow the item.

In the example below, a user sends a Webmail (**1**). The NBA (**2**) analyzes the Webmail and passes it to a policy engine (**3**) for processing. After applying CA DataMinder policy to the Webmail, the policy engine instructs the NBA to either allow the Webmail to continue out to the Internet (**4a**) or to block it (**4b**).



Example CA DataMinder user policy applied to NBA events

More information:

Machine Policy Changes (see page 144)
How Does the NBA Block Events? (see page 142)

How Are Participants Assigned to NBA Events?

For each file, IM conversation or email captured by the NBA and output via socket connection to policy engines, or output to disk and subsequently imported onto the CMS, CA DataMinder identifies the event participants.

Note: Captured or imported files include downloads, uploads, FTP transfers, email attachments and IM conversations. Captured or imported emails include Webmails, SMTP and POP3.

Email Participants

For emails captured by or imported from the NBA, the event participants are actually the sender and recipient email addresses.

Output to socket

When the NBA outputs items via a socket connection to policy engines, the policy engine stores the sender and recipient email addresses as participants when it processes the email.

Output to disk

If the NBA runs in passive mode and outputs items to disk, participants are not immediately assigned to captured emails. Instead, Event Import assigns the sender and recipient email addresses as participants when the emails are imported from the NBA.

IM Participants

For IM conversations captured by or imported from the NBA, the event participants are derived from the participants' display names plus, where possible, their email addresses. If the IM protocol does not make these addresses available to the NBA, the NBA derives a 'pseudo address' from a participant's display name and the IM protocol:

Format

<displayname>@chat-<protocol>

For example

frank1234@chat-yahoo

Pseudo addresses may be generated for the following IM protocols: AIMICQ, JABBER, MSN, SIP, and YAHOO.

In addition, the source and destination machine IDs are also stored as event participants.

More information:

Machine ID as Stored File or IM Event Participants (see page 134)

File Participants

For files captured by or imported from the NBA, the event participants are derived from the IP addresses of the source and destination machines. For details about machine IDs as event participants, see the next section.

Output to socket

When the NBA outputs files via a socket connection to policy engines, it also passes the IP addresses of the source and destination machines. The policy engine then stores both addresses as event participants.

Output to disk

If the NBA runs in passive mode and outputs items to disk, when the files are subsequently imported Event Import can optionally assign the IP address of the source machine as the event participant. This is controlled by the ImpFile.ParticipantsFromNBAFilename import parameter.

More information:

Import Parameters (see page 195)

Machine ID as Stored File or IM Event Participants

When a policy engine processes an NBA file or IM conversation, the resulting event is automatically associated with the source machine. Specifically, an address matching the machine's IP address is associated with each processed event and stored in the CMS database.

This means each NBA file or IM event is associated with a 'host machine' address. This provides a mechanism for associating file uploads, downloads, FTP transfers and IM events with individual CA DataMinder users. In CA DataMinder terms, these machine addresses are referred to as 'pseudo user addresses'.

Important! This mechanism only works if your organization assigns static IP addresses to users' computers. This mechanism does not work if your organization uses DHCP and introduces CA DataMinder security risks due to reassigned IP addresses.

Specifically, if an IP address is reassigned to another user's workstation, this can compromise CA DataMinder 's security models based on management groups during subsequent event searches. (These security models prevent reviewers from seeing events associated with users outside of their management groups.)"

Notes

- This mechanism does not apply to files sent as email or IM attachments. These are mapped to CA DataMinder users by conventional email addresses.
- To help ensure that the resulting file events are searchable by user name in the iConsole or Data Management console, CA DataMinder administrators must first add the relevant machine IP addresses to users' address lists in the Administration console. For details, see the online help; search the index for 'email addresses: updating'.

  For example, the NBA captures a file being uploaded from machine 10.0.169.5. To help ensure that this file can be retrieved during an iConsole event search, add that IP address to the address list for an appropriate CA DataMinder user account. (You add new addresses in the User Properties dialog in the Administration console.)

Which User Policy Is Applied to NBA Events?

This section describes how policy engines determine which user policy to apply to NBA-captured IM conversations, files and emails. It also describes which policy triggers and control actions are applicable to these NBA events.

Note: You also need to configure the machine policy for your policy engines.

More information:

Machine Policy Changes (see page 144)

Applying Policy to Emails

CA DataMinder policy engines apply **outgoing email** triggers to all emails (or Webmails) received from the NBA. The mechanism for applying policy is always the same, regardless of whether the NBA is in active or passive mode. First, the policy engine tries to map the sender's email address to a CA DataMinder user account. If the sender is not recognized, the Unknown Internal Sender and External Sender machine policy settings on the policy engine determine which policy is applied.

You also need to set up a CA DataMinder account—with an appropriate user policy—for the 'notification sender'.

Warnings and Encryption Are Not Supported

The NBA does not support email or Webmail warnings or encryption. Although the NBA applies outgoing email triggers to these items, it cannot apply Warn or Encryption control actions. The NBA can only apply Block, Quarantine or Categorize control actions to emails and Webmails.

More information:

Machine Policy Changes (see page 144)
Setting Up Notification Emails (see page 89)

Applying Policy to Files

CA DataMinder policy engines apply **Data In Motion** triggers to all files received from the NBA. The mechanism to identify the policy participant (that is, to determine which user policy gets applied) depends on the NBA output mode:

Output to socket

When the NBA outputs files via a socket connection, policy engines always apply the Default Policy for Files. This setting is defined in policy engines' machine policy; for details, see Default Policy for Files.

By default, the Default Policy for Files setting specifies the 'DefaultFileUser' account (this account is created automatically when you install a CMS). However, you may prefer to specify a custom CA DataMinder user account for the NBA (for example, 'NBA Policy User'). You can then tailor this account's user policy to apply triggers to files processed by the NBA.

Output to disk

If the NBA runs in passive mode and outputs items to disk, when the files are subsequently imported as part of Import Policy job, the correct user policy is specified by the ImpFile.PolicyParticipant import parameter. This parameter specifies an SMTP email address that the policy engine can map to an existing CA DataMinder user account.

If the policy participant is not specified, or the user account does not exist, policy engines apply the Default Policy for Files (see above).

Warnings and Encryption Are Not Supported

The NBA does not support warnings or encryption. Although the NBA applies outgoing Data In Motion triggers to files entering or leaving the corporate network, it cannot apply Warn or Encryption control actions. The NBA can only apply Block, Quarantine or Categorize control actions to these files.

More information:

Import Parameters (see page 195)

Applying Policy to IM Conversations

CA DataMinder policy engines apply **Outgoing Email** or **Data In Motion** triggers to IM conversations received from the NBA. Briefly, Outgoing Email triggers are always applied **unless** the IM conversation was conducted using a web-based IM application (such as www.ebuddy.com). Because such Web-based IM applications use the HTTPPOST file protocol, policy engines apply Data In Motion triggers to these IM conversations. If the NBA applies:

- **Outgoing email triggers**, the mechanism for determining which policy to apply is the same as for emails captured by the NBA. That is, the policy engine first tries to map a participant's email address (or pseudo address) to a CA DataMinder user account. If the address is not recognized, the Unknown Internal Sender and External Sender machine policy settings on the policy engine determine which user policy is applied.
- **Data In Motion triggers**, the mechanism for determining which policy to apply is the same as for files captured by the NBA—see the previous section.

More information:

Machine Policy Changes (see page 144)
IM Participants (see page 133)

User Policy Changes

When applying user policy to emails and files captured by, or imported from the NBA, you edit the triggers described in the topics that follow, plus the control actions and other policy settings.

More information:

Available Capture Actions (see page 140)

Available Triggers

Policy engines can apply the following triggers to NBA events:

Emails

You use Outgoing Email capture and control triggers to apply policy to emails captured by or imported from the NBA. For example, you can set up triggers to detect emails with specific content or which have specific characteristics (such as the sender's CA DataMinder account attributes).

For each trigger, verify that the 'Network Boundary Agent for Email' option is selected in the Which Email Sources? setting. By default, all options are selected.

IM conversations

You use Outgoing Email capture and control triggers to apply policy to IM conversations captured by or imported from the NBA. For example, you can set up triggers to block comments that use specific words or phrases.

For each trigger, verify that the 'Network Boundary Agent for Email' option is selected in the Which Email Sources? setting. By default, all options are selected.

Files

You use Data In Motion capture and control triggers to apply policy to files captured by or imported from the NBA, including downloads, uploads, FTP transfers, email attachments and IM file transfers. Each trigger can be configured to detect files with specific names or text content or which match a particular classification.

For each trigger, verify that the 'Network Boundary Agent for File' option is selected in the Which File Sources? setting. By default, all options are selected.

Available Control Actions

The range of available control actions depends on the NBA output mode:

Output to Socket

When the NBA outputs items via a socket connection, you can block unauthorized emails and IM comments by applying Outgoing Email control actions. You can also block unauthorized files by applying Data In Motion control actions

If a trigger fires and the control action is set to ‘block’ but the NBA application filter is set to 'monitor' or the NBA is in passive mode, the item is allowed to continue through the NBA without interruption.

You can also configure control actions to quarantine emails, or to categorize or apply smart tags to the resulting email or file events stored on the CMS.

However, you cannot use the Warn or Inform options that are provided with these control actions. These options are not supported for NBA events. Warn or Inform options can only be applied to events captured by CA DataMinder email agents and CA DataMinder file agents for client machines.

If you want to save a copy of the email or file, you must also specify a capture action.

Output to Disk

If the NBA runs in passive mode and outputs items to disk, you can categorize or apply smart tags to imported events. If you want to save a copy of the associated email, IM conversation or file, you also need to specify a capture action.

Note: It is possible to apply Block, Warn and Inform to NBA events captured in passive mode. Obviously, however, these control actions are only applied retrospectively. Nevertheless, these options may be useful during the evaluation phase of a CA DataMinder deployment. They can provide insight into the level violations that occur across your organization.

More information:

Available Capture Actions (see page 140)
How Does the NBA Block Events? (see page 142)
Quarantined Emails (see page 92)

Available Capture Actions

If you want to view the emails or files captured by the NBA, you need to configure the capture actions for your Outgoing Email and Data In Motion triggers. Without these policy changes, the email body and file content will not be retained on the CMS.

Emails

Edit the Capture Mail Detail? setting in your Outgoing Email capture actions; set this to 'True'. Then, in the Mail Details capture action subfolder, set the Capture Body? setting to 'True'.

Files

Edit the Capture File Details? setting in your Data In Motion capture actions; amend this setting to 'Attributes and File Data'.

Notifications for Blocked File Uploads, Web Pages and Webmails

You can set up CA DataMinder user policy to notify users if a Data In Motion trigger blocks a Web page or file upload, or if an Outgoing Email trigger blocks a Webmail.

To specify the title of a blocking notification

- Edit the 'Dialog Title — Blockings' setting in the \System Settings\User Notifications policy folder.

To specify the message text

- (For blocked file uploads and Web pages) Edit the 'Message To Users' setting in your Data In Motion control triggers.
- (For blocked Webmails) Edit the 'Message To Users' setting in your Outgoing Email control triggers.
- If you want to redirect users to an alternative URL, edit the 'Message To Users' setting. Instead of an explanatory message, set the Message To Users to be a URL. Remember to include the http:// prefix. For example:
  `http://www.hr.unipraxis.com`

Character Limit

There is a 1,000 character limit for NBA notification messages (including the title).

To use the title and message text defined here in an NBA blocking notification, you must use variables in the notification template.

Notifications for Blocked or Quarantined Emails

You can set up CA DataMinder user policy to display a notification to a sender when the NBA blocks or quarantines their email.

To specify the Subject text **of the notification:**

- Edit these settings in the \System Settings\User Notifications policy folder as required:

  Dialog Title — Blockings

  Dialog Title — Warnings and Quarantine Events

To specify the Body text **of the notification:**

- Edit the Message To Users setting in your Outgoing Email control triggers. In the control trigger, edit the 'Message to Users' to indicate that the sent email has been quarantined. Otherwise users may attempt to send the email again, believing the email has been blocked.

More information:

Blocked Emails: Notifying Users (see page 88)
Quarantined Emails (see page 92)
Set Up NBA Quarantining (see page 92)

Flag Emails as 'external'

The NBA is explicitly designed to capture external emails, sent from inside your organization and destined for the Internet. Therefore, one of the simplest ways to review NBA emails is to search for external emails. This requires a change in your user policies. Specifically, you need to edit the Internal Emails setting; find this in the \System Settings\Definitions policy folder.

How does this setting work? CA DataMinder flags events as **internal** when all the recipient addresses in an outgoing email match the specified Internal Emails address pattern. By implication, any emails where one or more recipients do not match this address pattern are flagged as **external**.

Detecting URLs in Traffic Crossing the Network Boundary

You can use the NBA to apply policy to network traffic originating from specific URLs. For example, you may want to capture all Facebook traffic, but only capture traffic from other URLs if the communication breaches your corporate regulations.

Note: Network events can include file uploads, comments posted to a web site, and page requests submitted to a web server.

How does CA DataMinder Detect URLs?

1. When the NBA analyzes files crossing the network boundary, the NBA stores URL details with the event metadata.

   For HTTPGET, HTTPPOST, and HTTPURL events, the NBA writes the URL as an attribute into the network event's XML metadata.

2. The NBA passes the network event and the XML metadata to a policy engine for analysis.

3. Policy engines apply Data in Motion triggers to network events captured by the NBA.

4. The Data In Motion triggers use XML data lookup commands to detect specific URLs in the metadata of transmitted files and other network events. If a specific URL is detected, the trigger fires.

You must configure the Data In Motion triggers to use suitable XML data Lookup commands. For details about the required policy changes, see 'Detecting URLs in Traffic Crossing the Network Boundary' in the *Policy Guide*.

How Does the NBA Block Events?

The NBA blocks an email, IM comment or file if a network or application filter in the NBA policy is set to prohibit. Similarly, an item may get blocked if an application filter is set to analyze and the policy engine subsequently blocks the item after applying Outgoing Email or Data in Motion control triggers.

Note: The NBA can only block events when it is connected inline between the corporate LAN and the internet *and* running in active mode.

What Does the End User See?

If the NBA blocks an end-user trying to send an email or Webmail, browse to a Web page, or upload a file, the end-user is notified accordingly. But if the NBA blocks a user's comment in an IM conversation, the user is not made aware of the blocking although the receiver may see the comment censored with asterisks.

Blocked emails

When the user attempts to send the email, it is not sent and they receive a notification email. The exact handling varies according to the email or Webmail application.

Initially, the user may see a message such as 'Error Sending Message' or 'We can't connect to Windows Live Hotmail right now.'

Subsequently, the user receives a notification email, explaining that their original email has been blocked. The notification arrives shortly in the inbox of whichever application they used to send the original (Outlook, Notes, Hotmail, and so on).

Blocked IM conversations

If a comment in an IM conversation gets blocked, the person who posted the comment is not made aware of the blocking and assumes it was sent normally. But the other participants only see a censored version of the comment, with all text replaced by asterisks.

Blocked files and Web pages

If a Web page or file upload gets blocked, the NBA notifies the user either by displaying a dialog (or HTML file) containing an explanatory message or it redirects the user to an alternative Web page containing an explanatory message (typically on the corporate intranet).

More information:

Blocked Emails: Notifying Users (see page 88)
Blocked Web Pages and File Uploads: Notifying Users (see page 96)

How Can Items Get Blocked?

If an item is prohibited by:

- **A network filter** in the NBA policy, the NBA behaves like a firewall and simply blocks the matching data packets.
- **An application filter** in the NBA policy, the NBA simply closes the TCP or UDP stream associated with the matching data packets.
- **A policy engine,** the NBA passes the full decoded object data stream to a policy engine for analysis. It also forwards all associated data packets onto the receiving computer *except for the final packet in the stream!*

  This final packet is retained by the NBA, pending the results of processing by the policy engine. If the policy engine instructs the NBA to:

  - **Allow** the email, IM comment or file, the NBA releases the final packet and sends it on to its destination. The receiving computer can then complete the transaction (for example, an email send operation or a HTTP Post operation).
  - **Block** the email, IM comment or file, the NBA closes the data stream prematurely, causing the destination device to recognize the error and discard the entire content of the stream. Note that for some transactions, this may result in the user being notified by their computer of disconnection errors.

Machine Policy Changes

On the CMS, you also need to edit various machine policy settings in the Policy Engine folder. These settings determine how policy engines assign imported emails and files to CA DataMinder user accounts.

We recommend that you make these changes in the common gateway policy to ensure that each policy engine inherits these policy changes. The machine policy settings that you need to edit are summarized below.

Note: Full details are in Policy Engines chapter of the *Platform Deployment Guide*.

Unknown Internal Sender

This setting specifies the name of a CA DataMinder user. It defaults to UnknownInternalSender. This user account is created automatically when you install a new CMS.

Policy engines use this setting to apply policy to emails sent from someone **within** your organization. The policy engine applies the Unknown Internal Sender's policy if the sender's address **matches** an address pattern listed in the Internal Email Address Pattern setting (see below) but no corresponding user exists.

Important! If you specify a different account, this must be a user account, not a group account.

External Sender

This setting specifies the name of a CA DataMinder user. It defaults to ExternalSender; this user account is created automatically when you install a new CMS.

Policy engines use this setting to apply policy to external emails (that is, emails sent from someone **outside** your organization) or **outbound Webmails** sent from an unrecognized address (such as mysteryman@hotmail.com). The policy engine applies the External Sender's policy if the sender's address **does not** match an address pattern listed in the Internal Email Address Pattern setting (see below).

Important! If you specify a different account, this must be a user account, not a group account.

Internal Email Address Pattern

This setting specifies a semicolon separated list of full or partial email addresses. The policy engine uses this setting to determine which policy to apply. When the policy engine processes an email:

- If the sender's email address matches this address pattern *and* the address is stored in the CMS database, the policy engine applies the sender's CA DataMinder policy to the email.
- If the sender's email address matches this address pattern but the address is *not* in the CMS database, the policy engine applies the policy associated with the Unknown Internal User—see above.
- If the sender's email address does *not* match this address pattern, the policy engine simply applies the policy associated with the External User—see above.

Default Policy for Files

This setting specifies the name of a CA DataMinder user. A policy engine will apply this user's policy to scanned, captured or imported files if no other means are available to determine the policy participant.

For example, if an Import Policy job or FSA scanning job omits to specify the policy participant, or if the specified user account does not exist, the policy engine applies the Default Policy for Files to the imported or scanned files.

Chapter 8: Searching for NBA Events

This chapter describes how to retrieve and review NBA files and emails using iConsole standard searches.

This section contains the following topics:

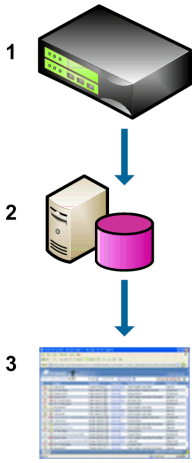
Overview

To find files captured by the NBA, you search for 'Network Events'. You can filter your search by 'channel'. For example, FTP transfers, HTTP file uploads and Webmail attachments are all different channels. You can also search for files originating on or sent to specific machines, or files with specific names. However, you cannot search by CA DataMinder user or group.

When searching for NBA emails (that is, Webmails, SMTP or POP3 messages), you search for 'Email Events'. You can filter your search to only retrieve external emails. You can also refine the search to focus on specific email address patterns (such as Hotmail or Yahoo addresses). However, you cannot normally search by CA DataMinder user or group.

When searching for IM conversations captured by the NBA, you search for 'IM Events'. You can filter your search by IM network, for example, to only retrieve MSN or Yahoo! IM events.

When reviewing NBA-captured events in the iConsole, you can zero in on individual files or emails and update their audit details in the same way as when reviewing events captured by other CA DataMinder agents.



Searching for NBA events

After files and emails imported from or captured by the NBA (**1**) have been stored in the CMS database (**2**), you can run iConsole searches to review and audit NBA network, email and IM events (**3**).

More information

Search for NBA Network Events (see page 149)

Search for NBA Network Events

This section describes how to use the iConsole to search for NBA network events. These events are defined as files, including FTP file transfers and IM or Webmail attachments, captured by CA DataMinder Network as they enter or leave the corporate network.

For network events, you can search by:

Source and Destination Machine

You can search by machine IP addresses to identify files originating on or sent to a specific machine. For file uploads or downloads, or files sent via IM conversations, you can also specify the name of the associated web server.

Protocols

You can search for files captured on specific communication channels:

- FTP: File transfers
- HTTP: File uploads or downloads
- IM Attachments: Attachments to IM conversations.
- WebMail Attachments: File attachments to messages sent using a Webmail protocol, such as Hotmail or Yahoo! Mail.

You can specify more detailed channel information in the Title or Subject field.

Title or Subject

Use this field to search for specific file types or filenames, such as *.PDF or *.DOC files.

For network events, the event title incorporates the channel type (such as HTTP-POST) and the filename. This allows you to use this field to search by channel. Event titles take this format:

```
<channel>:<original file name>
```

where <channel> can be:

```
FILE-AIMICQ
FILE-JABBER
FILE-MSN
FILE-SIP
FILE-YAHOO
FTP
FTP-GET
FTP-PUT
HTTP-GET
HTTP-POST
HTTP-URL
SMB
AOLMAIL-ATTACH
GMAIL-ATTACH
```

HOTMAIL-ATTACH
LIVEMAIL-ATTACH
YAHOO-ATTACH

For example, to search for any PDF files sent as attachments to GMail emails, you set the Title or Subject field to:

```
GMAIL-ATTACH:*pdf
```

Note: The filename modification happens after the files have been imported from the NBA and processed by policy engines.

User or group

Important! To search for network events by user or group, you must first have associated source machine IP addresses with your CA DataMinder users. You can then search by user or group in the normal way.

Search for NBA Email Events

When you search for NBA email (or Webmail) events in the iConsole, the usual range of search filters are available, including email addresses, internal or external, and so on. However, be aware that you may not be able to search by user or group.

External Emails

The NBA is explicitly designed to capture external emails, sent from inside your organization and destined for the Internet. Therefore, one of the simplest ways to review NBA emails is to search for external emails.

To allow the iConsole to identify external emails that you have imported using Event Import, you must first set up your Event Import configuration file or, for Import Policy jobs, your user policies to identify and flag internal emails. By definition, all other 'non-internal' emails are flagged as external. You can then set an 'External' filter when customizing your email searches in the iConsole.

Email Address

Using the iConsole Standard Search, you can filter your search by sender or recipient email address. In particular, this allows you to restrict your search by Webmail channel. For example, you can retrieve all emails where the sender's address matched '*@hotmail.com' or '*@yahoo.co.uk'.

Users or Groups

Important! Typically, you cannot search for NBA emails by user or group. This is because the senders of such emails frequently use private Webmail addresses.

If the sender of a Webmail has used an unrecognized email address (such as, mysteryman@hotmail.com) that is not associated with any CA DataMinder user accounts, CA DataMinder stores the email address as an event participant, but it cannot map the address to a CA DataMinder user. This means that you cannot search for such Webmails by sender or user group.

Search for NBA IM Events

When you search the iConsole for instant message events captured by the NBA, the available search filters are similar to those for email searches. For example, you can search by subject or participant email addresses. The iConsole standard search also includes an ‘Instant Message Network’ field where you can explicitly choose IM networks, such as Jabber or Yahoo!

In addition, event titles for IM events captured by the NBA are prefixed with the channel type (such as CHAT-YAHOO), enabling you to search for specific IM channels. But as with NBA email searches, you may not be able to search by user or group.

Search for NBA NNTP (News) Events

Messages posted to or read from a news group are handled as emails by CA DataMinder. To search for these news group messages in the iConsole, you must search for email events and filter the search to include NNTP recipient addresses. To search for:

- Messages posted to a news group, the recipient address filter must begin 'NNTPPOST@'
- Messages read from a news group, the recipient address filter must begin 'NNTPGET@'

For example, you can retrieve all posts to the Unipraxis new group by running an email search for messages sent to 'NNTPPOST@unipraxis.com'.

Chapter 9: Specifying NBA Policy in XML

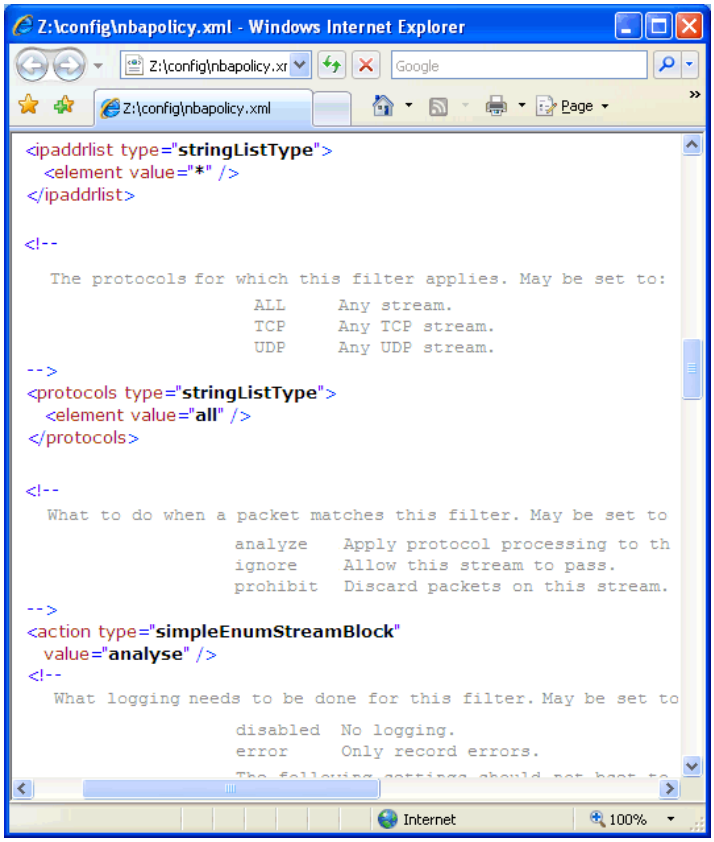
This section contains the following topics:

Overview of nbapolicy.xml

The NBA policy comprises the network filters and application filters that jointly determine how the NBA handles data packets passing through the NBA. The policy also specifies which policy engines the NBA can use and includes settings to control NBA log files. The policy is defined in nbapolicy.xml; find this file in the \config folder on the NBA.

This section focuses on the XML tags that define the NBA policy. In particular, it includes details of how to specify network and application filters based on IP addresses and TCP port numbers.

Note: When replacing the nbapolicy.xml file in the \config folder, make sure that the file modification time is later than the previous file's modification time. The NBA detects that the file has changed using this timestamp, and it reloads NBA policy only if the file is newer.



Example nbapolicy.xml

XML Syntax: nbapolicy.xml

This section describes the syntax for the NBA policy file, nbapolicy.xml. You can edit XML tags in this file to define network filters, application filters, to specify which policy engines the NBA can use, and to configure logging operations.

The structure of this file, and the available XML tags, plus any associated attributes and nested tags, are described in the following sections. The tag descriptions are organized as follows:

- General tags
- Network filter tags
- Application filter tags
- Policy engine tags
- Logging tags

Note: If nbapolicy.xml contains syntax errors, these are recorded in the Agent Management log file.

More information:

About nbaconfig.xml (see page 231)
Example NBA Policy File (see page 178)

XML Tags

The following is a list of the XML tags:

```
<networkagent>
    <description>
   <online>
    <active>

    <networkfilters>
        <filtergroup>
            <groupname>
            <networkfilter>
                <filtername>
                <ipaddrlist>
                <protocols>
                <action>
                <loglevel>

    <applicationfilters>
        <filtergroup>
            <groupname>
            <applicationfilter>
                <filtername>
                <ipaddrlist>
                <protocols>
                <action>
                <loglevel>

    <settings>
        <analyzeservers>
        <standbyanalyzeservers>
        <enterprisednslist>
        <logging>
            <loglevel>
            <numberoflogfiles>
            <maxsizeoflogfileskb>
            <logrolloverhours>
            <statslogintervalsecs>
        <ssl>
            <domainexcludelist>
            <serverexclusioncache>
            <clientexclusioncache>
        <capturepartialobjects>
        <captureftplogs>
        <htmlblocktemplate>
        <prohibittitle>
        <prohibitmessage>
```

General Policy Tags

This section defines the general XML tags used in nbapolicy.xml.

<networkagent>

Contains the NBA policy tags described below.

<description>

Sets a description of the NBA policy defined in nbapolicy.xml. Typically, this description includes a summary of the various filters used by the NBA when processing data packets.

This tag supports type and value attributes:

type

Always set to type="stringType".

value

Sets a text description of the NBA policy. For example:

```
<description type="StringType" value="Blocks outbound Webmails." />
```

<online>

Defaults to true. This tag determines whether the NBA is online or offline. It is also included in nbaconfig.xml.

Note:

- The Packet Processing button in the Policy screen of the NBA console has the same effect as this policy tag.
- For the NBA to be online, this tag must be set to true in both nbapolicy.xml and nbaconfig.xml.

This tag supports type and value attributes:

type

Always set to type="booleanType".

value

Can be true or false. For the NBA to be online, both nbapolicy.xml and nbaconfig.xml must contain:

```
<online type="booleanType" value="true">
```

<active>

Defaults to false. This tag determines whether the NBA is running in Active or Passive mode. It is also included in nbaconfig.xml.

Note:

- The Stream Blocking button in the Policy screen of the NBA console has the same effect as this policy tag.
- For the NBA to be in Active mode, this tag must be set to true in both nbapolicy.xml and nbaconfig.xml.

This tag supports type and value attributes:

type

Always set to type="booleanType".

value

Can be true or false. For the NBA to be in Active mode, both nbapolicy.xml and nbaconfig.xml must contain:

```
<active type="booleanType" value="true">
```

Network Filter Tags

This section defines the XML tags used to define NBA network filters.

<networkfilters>

Contains the network filter tags listed below. It has a single enabled attribute:

enabled

Can be set to:

- **enabled="true"**--Enables all NBA network filter groups. Note that you can still disable individual groups of network filters.
- **enabled="false"**--Disables all NBA network filters. Data packets pass through the NBA without interruption.

For example:

```
<networkfilters enabled="true" />
```

<filtergroup>

Contains tags for a group of related network filters. It supports a single attribute:

enabled

Can be set to:

- **enabled="true"**--Enables the filter group. Note that you can still disable individual filters within the group.
- **enabled="false"**--Disables the filter group. No filters in this group are applied to packets passing through the NBA.

For example:

```
<filtergroup enabled="true" />
```

<groupname>

Defines the name of the filter group. The group name is saved in the policy.txt diagnostic file to make the file easier to comprehend.

On Bivio 7000 appliances, the Group fields in the Filters screen of the NBA console have the same effect as this policy tag.

This tag supports type and value attributes:

type

Always set to type="stringType".

value

Sets the name of the filter group. For example:

```
<groupname type="StringType" value="TCP blocking filters" />
```

<networkfilter>

Contains any tags associated with a specific network filter. It supports a single enabled attribute:

enabled

This attribute can be set to:

- **enabled="true"**--Enables the filter.
- **enabled="false"**--Disables the filter.

For example:

```
<networkfilter enabled="true" />
```

<filtername>

Sets a name for the current filter. The filter name is saved in the policy.txt diagnostic file to make the file easier to comprehend.

The Filter fields in the Filters screen of the NBA console have the same effect as this policy tag.

This tag supports type and value attributes:

type

Always set to type="stringType".

value

Sets a description of the filter. For example:

```
<filtername type="StringType" value="Webmail blocking filter" />
```

<ipaddrlist>

Defines the source IP addresses and TCP ports that you want the network filter to detect.

That is, you can configure the NBA to analyze, prohibit or ignore packets sent from the specified IP addresses or being transmitted to the specified port numbers. You can specify any combination of IP addresses, address ranges, ports, and port ranges.

This tag supports a single type attribute and one or more nested <element> tags.

type

Always set to type="stringListType". For example:

```
<ipaddrlist type="StringType"/>
```

<element>

Defines a specific IP address or address range. The <element> tag can optionally also specify a port number or port range. You can define multiple <element> tags. Each supports a single value attribute:

value

Defines an actual IP address and port number, such as:

```
<element value="192.160.0.5"/>
<element value="10.0.1/24"/>
<element value="10.0.1.5:128"/>
<element value="192.160.0.5:128-1023"/>
<element value="*"/>
```

<protocols>

Specifies which transport protocol to detect: TCP, UDP, or both.

Briefly, TCP is typically used by emails and file transfer applications that require packets to be delivered reliably and in the correct order. Conversely, UDP is typically used by network applications that do not require guaranteed packet delivery such as video or audio streaming, Voice over IP and online gaming. For example, you may want to use this tag to block video streaming from a specific website; such streams are typically transmitted as UDP packets.

This tag supports a single type attribute and one or more nested <element> tags.

type

Always set to type="stringListType". For example:

```
<protocols type="stringType"/>
```

<element>

Defines which transport protocols the network filter looks for. You typically only need a single <element> tag. This tag supports a single value attribute.

value

Defines the actual transport protocol:

- **value="all"**--This filter applies to all data packets.
- **value="tcp"**--This filter only applies to TCP packets. Other packets are ignored and pass through this filter.
- **value="udp"**--This filter only applies to UDP packets. Other packets are ignored and pass through this filter.

For example:

```
<element value="all" />
```

<action>

Defaults to analyze. This tag determines how the NBA handles data packets that meet the network filter criteria. You can configure the filter to analyze, decrypt, ignore, or prohibit these packets. This tag supports type and value attributes:

type

Always set to type="simpleEnumStreamBlock".

value

Defines how the filter handles data packets arriving at the NBA:

- **value="analyze"**--The NBA passes data packets to an application filter for further analysis.
- **value="decrypt"**--The NBA decrypts data packets (if necessary) and then passes them to an application filter for further analysis.
- **value="ignore"**--The NBA ignores packets that meet the filter criteria and permits them to pass through the NBA without interruption.
- **value="prohibit"**--When the NBA is operating in Active mode, it blocks packets that meet the filter criteria. They do not pass through the NBA. When operating in Passive mode, the NBA ignores packets that meet the filter criteria and permits them to pass through the NBA without interruption.

For example:

```
<action type="simpleEnumStreamBlock" element value="analyze" />
```

<loglevel>

Defaults to error. This tag determines the level of logging for changes for the current network filter. Logging details are recorded in the NBA agent management log files. The syntax is the same as the <loglevel> tag for the overall NBA policy; see the Logging Tags section for details.

Under normal conditions, you would set this tag to:

```
<loglevel type="simpleEnumLogLevel" value="error">
```

Other supported values are: none, warnings, objects, and debug.

More information:

Logging Tags (see page 173)

Application Filter Tags

This section defines the XML tags used to define NBA application filters.

<applicationfilters>

Contains the application filter tags listed below. It has a single attribute:

enabled

Can be set to:

- **enabled="true"**--Enables all NBA application filter groups. You can still disable individual groups of application filters.
- **enabled="false"**--Disables all NBA application filters. No application filters are applied to packets passing through the NBA.

For example:

```
<applicationfilters enabled="true" />
```

<filtergroup>

Contains tags for a group of related application filters. It supports a single attribute:

enabled

Can be set to:

- **enabled="true"--**Enables the filter group. Note that you can still disable individual filters within the group.
- **enabled="false"--**Disables the filter group. No filters in this group are applied to packets passing through the NBA.

For example:

```
<filtergroup enabled="true" />
```

<groupname>

Defines the name of the filter group. The group name is saved in the nbapolicy.txt diagnostic file to make the file easier to comprehend.

On Bivio 7000 appliances, the Group fields in the Filters screen of the NBA console have exactly the same effect as this policy tag.

This tag supports type and value attributes:

type

Always set to type="stringType".

value

Sets the name of the filter group. For example:

```
<groupname type="StringType" value="Webmail filters" />
```

<applicationfilter>

Contains any tags associated with a specific application filter. It supports a single attribute:

enabled

This attribute can be set to:

- **enabled="true"**--Enables the filter.
- **enabled="false"**--Disables the filter.

For example:

```
<applicationfilter enabled="true" />
```

<filtername>

Sets a name for the current filter. The filter name is saved in the nbapolicy.txt diagnostic file to make the file easier to comprehend.

This tag supports type and value attributes:

type

Always set to type="stringType".

value

Sets a description of the filter. For example:

```
<filtername type="StringType" value="Webmail blocking filter" />
```

<ipaddrlist>

This tag defines the source IP addresses and TCP ports that you want the application filter to detect. That is, you can configure the NBA to analyze, prohibit, monitor or ignore files or messages sent from the specified IP addresses or being transmitted to the specified port numbers. You can specify any combination of IP addresses, address ranges, ports, and port ranges.

This tag supports a single type attribute and one or more nested <element> tags.

type

Always set to type="stringListType". For example:

```
<ipaddrlist type="StringType"/>
```

<element>

Defines a specific IP address or address range. The <element> tag can optionally also specify a port number or port range. You can define multiple <element> tags. Each supports a single value attribute:

value

Defines an actual IP address and port number, such as:

```
<element value="192.160.0.5"/>
<element value="10.0.1/24"/>
<element value="10.0.1.5:128"/>
<element value="192.160.0.5:128-1023"/>
<element value="*"/>
```

<protocols>

Specifies which object types or application layer protocols to detect. These include email, Webmail, IM and file transfers. You can either set the application filter to detect all protocols or you can selectively target individual protocols.

For example, you may want to monitor ICQ and Yahoo! IM conversations in real time, analyzing and blocking inappropriate comments.

The <objtypes> object types tag in nbaconfig.xml uses the same syntax.

This tag supports a single type attribute and nested <element> tags.

type

Always set to type="stringListType". For example:

```
<protocols type="stringListType" />
```

<element>

Defines the protocols that the filter looks for. This tag supports a single value attribute. To specify multiple protocols, use multiple <element> tags.

value

Specifies the actual protocols that the filter looks for. It can be se to:

- value="ALL"--Filter applies to all the following protocols except HTTPURL. See note 1 below.

To detect multiple protocols of the same type, set the attribute to:

- value="FTP"--Filter applies to FTPGET and FTPPUT.
- value="IM_ALL"--Filter applies to ICQIM, JABBERIM, MSNIM, SIPIM and YAHOOIM.
- value="SMTP"--Filter applies to SMTPDEST and SMTPSRC. See note 2 below.
- value="WEBMAIL"--Filter applies to AOLMAIL, GMAIL, HOTMAIL and YAHOOMAIL.

To detect specific protocols, set the attribute to:

- value="AOLIM"--Same protocol as ICQIM. See note 3 below.
- value="AOLMAIL"
- value="DELTASYNC"
- value="FTPGET"
- value="FTPPUT"
- value="GMAIL"
- value="HOTMAIL"
- value="HTTPGET"
- value="HTTPPOST"
- value="HTTPURL"--See note 1 below.
- value="ICQIM"--Same protocol as AOLIM. See note 3 below.

- value="JABBERIM"
- value="MSNIM"
- value="NNTPGET"
- value="NNTPPOST"
- value="POP3"
- value="SIPIM"
- value="SKYPE"
- value="SMB"
- value="SMTPDEST"--See note 2 below.
- value="SMTPSRC"--See note 2 below.
- value="YAHOOIM"
- value="YAHOOMAIL"

Note 1: ALL does not detect HTTPURL

Be aware that ALL does *not* detect the HTTPURL protocol! To verify that all protocols are detected, add the following lines to your application filter definition:

```
<protocols type="stringListType">
<element value="all"/>
<element value="httpurl"/>
</protocols>
```

Important! Analyzing HTTPURL will create a lot of extra network traffic between the NBA and the policy engines. Only include this protocol when it's really needed.

Note 2: **SMTP, SMTPSRC and SMTPDEST**

SMTP detects emails coming from or going to a specific set of listed IP addresses. List the addresses in the <ipaddrlist> part of the filter.

SMTPSRC detects emails *from* listed IP addresses.

SMTPDEST detects emails going *to* listed IP addresses.

Note 3: **AOLIM and ICQIM**

These protocols are interchangeable. They detect the same data streams. AOLIM streams are typically encrypted in a way that the NBA cannot decode. ICQIM streams can be detected.

<action>

Defaults to 'analyze'. This tag determines how the NBA handles files or messages that meet the application filter criteria. You can configure the filter to analyze, monitor ignore or prohibit these files or messages.

This tag supports type and value attributes:

type

Always set to type="simpleEnumStreamBlock".

value

Defines how the filter handles files or messages arriving at the NBA:

- **value="analyze"**--The NBA analyzes the data stream and reassembles the relevant packets into a file or email, which it then passes to a policy engine for processing. When the policy engine returns:

  An 'allow' result, the stream is permitted to pass through the NBA.

  A 'block' result, and the NBA is in active mode, the NBA blocks the data stream.

  A 'block' results, and the NBA is in passive mode, the stream is permitted to pass through the NBA. When a reviewer subsequently searches for this event in the iConsole, the results screen shows that "real-time intervention was not applied to this event".

- **value="monitor"**--This is similar to analyze, but crucially the NBA does **not** block files or messages, even if requested to do so by the policy engine (PE) as a result of processing.

  For example, if a user policy trigger blocks a file but the application filter is only set to monitor, the NBA ignores the PE request to block the file. When a reviewer searches for this file event later, the iConsole search results screen shows that "real-time intervention was not applied to this event".

- **value="ignore"**--The NBA ignores packets that meet the filter criteria and permits them to pass through the NBA without interruption.

- **value="prohibit"**--When the NBA is in active mode, it blocks packets that meet the filter criteria. They do not pass through the NBA.

  When the NBA is in passive mode, it ignores packets that meet the filter criteria and permits them to pass through the NBA without interruption.

<loglevel>

Defaults to error. This tag determines the level of logging for changes for the current application filter. Logging details are recorded in the NBA agent management log files. The syntax is the same as the <loglevel> tag for the overall NBA policy; see the Logging Tags section for details.

Under normal conditions, you would set this tag to:

```
<loglevel type="simpleEnumLogLevel" value="error">
```

Other supported values are none, warnings, objects, and debug.

More information:

Logging Tags (see page 173)

Settings Tags

This section defines the XML tags used to specify general NBA policy settings. For example, these tags specify which policy engines are available to the NBA and which network domains indicate to the NBA that an email may have already been processed.

<settings>

Contains the policy engine tags and NBA logging tags described below and on <logging>.

<enterprisednslist>

Defaults to an empty string.

Specifies a list of DNS domains. The NBA checks the 'policy processed' status of any emails arriving from these domains to ensure that emails passing through the NBA are not needlessly reprocessed.

When the Socket API generates a notification email, or when the Quarantine Manager releases an email from quarantine, these components write a custom header containing the domain details and 'policy processed' status to those emails. These domain details are then compared against the domain list defined for <enterprisednslist>.

The Enterprise MailServer DNS List field in the Policy screen of the NBA console has exactly the same effect as this policy tag.

type

Always set to type="stringListType".

<element>

Identifies a single DNS domain name. If required, use multiple <element> tags to identify multiple domains.

Each <element> supports a single attribute.

value

Specifies a DNS domain. For example:

```
<element value="unipraxis.com"/>
```

<analyzeservers>

Defaults to an empty string (no policy engines).

Identifies the policy engines or PE hub that the NBA will use when running in active mode. Machines hosting a policy engine or PE hub are identified by their IP address and port number.

More accurately, this tag identifies machines hosting the CA DataMinder Socket API which, by definition, also host a policy engine or PE hub.

The Policy Analyzer Addresses field in the Policy screen of the NBA console has exactly the same effect as this policy tag.

This tag supports a single type attribute and one or more nested <element> tags.

type

Always set to type="stringListType".

<element>

Identifies a single host machine.

If multiple <element> tags are used to identify multiple machines (each hosting a policy engine), the policy engines are assigned separately to NBA processors on a round-robin basis. For example, if six host machines are specified, one policy engine is assigned to each of the six processors on the NBA. If fewer than six are specified, the policy engines are shared by the NBA processors.

Each <element> supports a single value attribute—see below.

value

Specifies the IP address and port number of a host machine. The default port is 8539. The syntax is:

`value="`*`IP_address`*`:`*`port`*`"`

For example:

```
<element value="10.0.1.96:8539"/>
<element value="10.0.1.98"/>
```

<standbyanalyzeservers>

Defaults to an empty string (no policy engines).

Identifies the standby policy engines or standby PE hub that the NBA can use. If the active policy engines or hub become unavailable (for example, because of a system failure), the NBA switches to using the standby policy engines or hub.

The Standby Analyzer Addresses field in the Policy screen of the NBA console has exactly the same effect as this policy tag.

The tag syntax is the same as for the <analyzeservers> tag above.

<capturepartialobjects>

Defaults to true. This tag specifies whether the NBA applies policy to incomplete data streams, or whether it disregards them.

Incomplete data streams are missing one or more data packets (or frames). The missing packets mean that the NBA is unable to fully reassemble the email or file object.

This tag supports type and value attributes:

type

Always set to type="booleanType".

value

Can be set to:

value="true"--Captures and processes incomplete streams.

value="false"--NBA disregards incomplete streams.

For example:

```
<capturepartialobjects type=boolezanType" value="true" />
```

<captureftplogs>

Specifies whether to record user FTP session details. When enabled, the NBA records any FTP commands run by a user during an individual FTP session. The sequence of commands and responses are combined into a single event and sent to policy engines for processing.

This provides reviewers with additional insight into any FTP transfers performed or attempted by users. For example, these 'FTP events' identify the logged on user and the file being transferred. When sent to policy engines, FTP events are analyzed by Data In Motion triggers.

Be aware that you cannot configure triggers to block these FTP commands because they are captured after the commands have run. This tag is provided primarily for monitoring purposes.

To use this feature, the NBA policy must include an application filter where the <protocols> tag is set to detect the FTP protocol and the <action> tag set to analyze. These are described in the Application Filter Tags section.

type

Always set to type="booleanType".

value

Set this to true to capture FTP session commands. Set it to false if you do not want the NBA to record this information.

<htmlblocktemplate>

Identifies an HTML template that contains the notification message shown to users when a Web page, file upload or Webmail is blocked.

By default, the NBA looks for this file in the \config folder on the NBA (that is, in the same folder as nbapolicy.xml). The NBA FTP folder structure is described in What is on the NBA? (see page 73).

type

Always set to type="stringType".

value

Specifies the name of your template file.

You can set the value to include path details if, for example, you store your template in a subfolder below the \config folder.

If this tag is omitted completely from the NBA policy, the NBA defaults to use blocktemplate.html.

<prohibittitle>

Specifies the title text for the notification message shown to users when a Web page, file upload or Webmail is prohibited by an NBA application filter.

type

Always set to type="stringType".

value

Specifies the title for the notification message. For example:

```
<prohibittitle value="Unipraxis Advisory"/>
```

<prohibitmessage>

Specifies the body text for the notification message shown to users when a Web page, file upload or Webmail is prohibited by an NBA application filter. Or it can specify an alternative URL.

type

Always set to type="stringType".

value

Specifies the body text for the notification message, for example:

```
<prohibitmessage value="You are not authorized to visit this Web site"/>
```

To include line breaks within the body text based on
 tags, for example to divide the message into separate paragraphs, you must use < and > codes instead of angle brackets:

```
<prohibitmessage value="You are not authorized to visit this site.
&lt;BR&gt;Contact HR for details.">
```

Alternatively, you can redirect users to an alternative URL. Instead of an explanatory message, the tag can specify a URL; remember to include the http:// prefix. For example:

```
<prohibitmessage value="http://www.hr.unipraxis.com"/>
```

Note: There is a 1,000 character limit for NBA notification messages (including the title).

Logging Tags

This section defines the XML tags used to define NBA logging operations.

<logging>

Contains the tags that control NBA logging activity.

<numberoflogfiles>

Defaults to 10. This tag specifies the maximum number of log files per NBA processor.

When the maximum number of log files exists and the maximum size of the latest is reached (see below), the oldest log file is deleted to enable a new one to be created.

This tag supports type and value attributes:

type

Always set to type="numberType".

value

Specifies the maximum number of log files of each type, for each NBA processor. For example:

```
<numberoflogfiles type="numberType" value="10" />
```

<maxsizeoflogfileskb>

Defaults to 1024. This tag specifies the maximum size (in KB) for each log file. When the current log file reaches its maximum size, the NBA creates a new log file.

This tag supports type and value attributes:

type

Always set to type="numberType".

value

Specifies the maximum size for log files. For example:

```
<maxsizeoflogfileskb type="numberType" value="1024" />
```

<loglevel>

Defaults to error. This tag determines the default level of logging for the packet capture process. You can override this logging level for individual network filters or application filters, all of which have their own <loglevel> tag.

The Logging Level field in the Policy screen of the NBA console has the same effect as this policy tag.

This tag supports type and value attributes:

type

Always set to type="simpleEnumLogLevel".

value

Can be set to:

- **value="none"**--Only startup messages and a few significant events are logged.
- **value="error"**--As "none", but errors are also logged.
- **value="warning"**--As "error", but warnings are also logged.
- **value="objects"**--As "warning", but captured object information is also logged.
- **value="debug"**--As "object", but debug messages are also logged.

In normal NBA operations, the logging level is typically set to error. Other levels are supported for evaluation, diagnostic, or testing purposes. In particular, debug logging causes the log file to grow extremely rapidly.

<logrolloverhours>

Defaults to 0 (no time limit). This tag specifies how often (in hours) a new log file is created, even if the current log file has not reached its maximum size. If set to zero, a new log file is only created when the current log file reaches its maximum size.

This rollover tag applies to all types of log file for each NBA processor.

This tag supports type and value attributes:

type

Always set to type="numberType".

value

Specifies log file rollover interval (in hours). For example:

```
<logrolloverhours type=numberType value="24"/>
```

<statslogintervalsecs>

Defaults to 60. This tag specifies how often (in seconds) the NBA statistics log files are updated. Statistics are recorded in the statistics log files.

This tag supports type and value attributes:

type

Always set to type="numberType".

value

Specifies the statistics update frequency (in seconds). For example:

```
<statslogintervalsecs type=numberType value="60"/>
```

More information:

Log Files (see page 205)

SSL Decode Tags

This section defines the XML tags used to control NBA SSL Decode.

<ssl>

Contains the tags that control NBA SSL Decode.

<domainexcludelist>

Specifies a list of URL domains that are not subject to SSL Decoding. Domains are checked in two ways:

If you have an RFC2817 HTTP CONNECT proxy that browsers use to connect to secure web sites and the NBA is between the clients and the proxy, the destination domain for each connection is checked by the NBA. If the domain matches a listed domain, the SSL connection is allowed to proceed without decode.

Connections that do not go through a proxy have their domains checked against the "Subject" or "Issued to" property of the SSL certificate. The first connection to the domain gets closed and subsequent connections are allowed to proceed without decoding.

Sub-domains are also excluded from decoding. If the excluded domain is "company.com" but the site is "special.company.com", the domain is still excluded.

type

Always set to type="stringListType".

<element>

Identifies a single domain name. Use multiple <element> tags to identify multiple domains.

Each <element> supports a single attribute.

value

Specifies a domain.

Example:

```
<element value="update.microsoft.com"/>
<element value="activation.sls.microsoft.com"/>
```

<serverexclusioncache>

(Optional) The server exclusion cache will allow unmonitored sessions to SSL servers that will not accept connections from the decoder. This might be because the decoder's SSL protocols are unacceptable to the server. An attempt to connect to the server has to be made before the decoder can determine this, so it's only subsequent connections that will be permitted. The IP address and port number of the server are cached so that future connections to this server and port will not be subject to SSL decode.

If there is a web proxy or some other device between the decoder and the internet that hides the real server's IP address from the internal network, the server exclusion cache cannot be used and it must be disabled. This is because all servers will appear to have the same IP address so one exclusion will affect all connections.

type

Always set to type="booleanType".

value

Defaults to false, disabling this cache.

<clientexclusioncache>

(Optional) The client exclusion cache will allow unmonitored sessions from clients that fail to connect to the decoder. This could be because the client has not had the decoder's master root certificate installed (though some applications don't cause the SSL negotiation error needed to trigger the cache - they just close the connection after it has been negotiated). The IP addresses of both server and client as well as the port number of the server are cached so that future connections from this client to this server will not be subjected to SSL decode.

If there is a web proxy or other device between the decoder and clients that hides the real client IP addresses from the NBA, this cache must be disabled. This is because all clients will appear to have the same IP address so one exclusion will affect all clients.

type

Always set to type="booleanType".

value

Defaults to false, disabling this cache.

Example NBA Policy File

The NBA policy XML file defines the network filters and application filters. It also identifies the policy engines or hubs available to the NBA. Finally, it specifies the logging levels for NBA operations.

```
<?xml version="1.0" encoding="UTF-16"?>
<wigan xmlns:xsi="http://www.w3.org/1999/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="wigan://NBAPolicy v1.0">
    <networkagent>
        <description type="stringType" value="This text is logged when
         a new policy is ingested by the NBA."/>
        <online type="booleanType" value="true"/>
        <active type="booleanType" value="true"/>
        <applicationfilters enabled="true">
            <filtergroup enabled="true">
                <groupname value="File group"/>
                <applicationfilter enabled="true">
                    <filtername type="stringType"
                     value="NBA Application Filter 1"/>
                    <ipaddrlist type="stringListType">
                        <element value="*"/>
                    </ipaddrlist />
                    <protocols type="stringListType">
                        <element value="ALL"/>
                        <element value="HTTPURL"/>
                    </protocols>
                    <action type="simpleStreamBlockEnumeration" value="analyze"/>
                    <loglevel type="simpleEnumLogLevel" value="error"/>
                </applicationfilter>
            </filtergroup>
        </applicationfilters>
```

```
<networkfilters enabled="true">
    <filtergroup enabled="true">
        <groupname value="main group"/>
            <networkfilter enabled="false">
                <filtername type="stringType"
                 value="NBA Network Filter 1"/>
                <ipaddrlist type="stringListType">
                    <element value="*"/>
                </ipaddrlist >
                <protocols type="stringListType">
                    <element value="ALL"/>
                </protocols >
                <action type="simpleStreamBlockEnumeration"
                 value="analyze"/>
                <loglevel type="simpleEnumLogLevel" value="error"/>
            </networkfilter>
    </filtergroup>
</networkfilters>
<settings>
    <enterprisednslist type="stringListType">
        <element value = "unipraxis.com"/>
        <element value = "unipraxis.co.uk"/>
    </enterprisednslist>
    <analyzeservers type="stringListType">
        <element value = "10.0.1.96:4456"/>
        <element value = "10.0.1.97:4456"/>
        <element value = "10.0.1.98:4456"/>
    </analyzeservers>
    <standbyanalyzeservers type="stringListType">
        <element value = "10.0.1.96:4456"/>
        <element value = "10.0.1.97:4456"/>
        <element value = "10.0.1.98:4456"/>
    </standbyanalyzeservers>
    <ssl>
        <domainexcludelist type="stringListType">
            <element value="update.microsoft.com"/>
            <element value="download.microsoftupdates.com"/>
            <element value="activation.sls.microsoft.com"/>
            <element value="windowsupdate.microsoft.com"/>
        </domainexcludelist>
        <serverexclusioncache type="booleanType" value="false"/>
        <clientexclusioncache type="booleanType" value="false"/>
    </ssl>
```

```
            <capturepartialobjects type="booleanType" value="false"/>
            <captureftplogs type="booleanType" value="false"/>
            <htmlblocktemplate type="stringType"" value="blocktemplate.html"/>
            <prohibittitle type="stringType"" value="Unipraxis Advisory"/>
            <prohibitmessage type="stringType"" value="This Web site is blocked."/>
            <logging>
                <numberoflogfiles type="numberType" value="10"/>
                <maxsizeoflogfileskb type="numberType" value="1024"/>
                <loglevel type="simpleEnumLogLevel" value="error"/>
                <logrolloverhours type="numberType" value="0"/>
                <statslogintervalsecs type="numberType" value="60"/>
            </logging>
        </settings>
    </networkagent>
</wigan>
```

IP Address and Port Filters

You can set up network and application filters in the NBA policy to detect packets sent from specified IP addresses and port numbers. These packets are then analyzed, prohibited or ignored, depending on how the filter is configured.

To set up a filter based on IP addresses and ports, you configure an <ipaddrlist> tag in nbaconfig.xml. This tag can contain multiple <element> tags, each specifying a separate combination of IP addresses and port ranges; see <ipaddrlist> for tag details. Specify each address, address range, port, or port range on a separate line.

This section describes how to configure the <element> tag to define the IP address and port filters that you want the NBA to use.

More information:

Specifying IPv6 Adresses (see page 80)

Implied Address Masks

When you specify an IP address, you can use 'implied address masks' to specify an address range. If a specified IPv4 address has less than four octets, there is an implicit mask for the unspecified octets.

Example: 10.5 is interpreted to be the same as 10.5/16.

When specifying an address range, the implied mask only applies to the start address; if the end address has less than four octets, these are interpreted as the least significant octets and the omitted octet(s) are inherited from the start address.

Example: The address range 10.0.1-40 equates to 10.0.1.0-10.0.1.40.

You cannot specify an implied address mask for an IPv6 address.

IP Address and Port Syntax

You use the <ipaddrlist> tag to specify the source IP addresses and TCP ports that you want the network filter. Its syntax is:

```
<ipaddrlist type="stringListType"> <element value="IPfilter"/> </ipaddrlist>
```

Where <element value="IPfilter"/> represents a combination of IP addresses and port numbers.

When specifying IP addresses and port numbers, the IPfilter syntax comprises the following elements:

IPfilter

Specifies a single pre-filter, comprising an address range and, optionally, a port range. It takes this format (note the colon separator):

```
IPrange[":"PortRange]
```

For details on specifying a range of IP addresses or ports (that is, IPrange and PortRange), see below. Note also:

- You can use a single * wildcard to indicate all IP addresses and ports.
- If you omit IPrange (you do not specify any addresses) the NBA detects packets sent to or from any IP address.
- Likewise, if you omit PortRange (you do not specify any ports) the NBA detects packets sent to or from any port.

IPrange

Specifies one of the following: a single address; a hyphen-separated address range; or a masked address range. It takes this format:

```
IPaddress["-"IPaddress|"/"Mask]
```

IPv4 address

Specifies up to four address octets. If less than four octets are specified, an address mask is implied—see the previous section. It takes this format:

```
Address *3["."Address]
```

IPv6 Address

Specifies an IPv6 address in colon-separated format as eight 16-bit words in hexadecimal. Two or more consecutive 16-bit words may be represented by a double colon. If you specify a port number or port range in a filter, enclose the IPv6 addresses in brackets, for example:

```
[fe80::]-[fe81::]:137-139
```

Address

Specifies a single address octet. This can be any value from 0 to 255.

Mask

Defines a subnet mask of up to 32 bits. This can be any value from 0 to 32.

The mask specifies the number of most significant bits used when matching IP addresses to the filter address range. Address bits outside the mask are ignored and, for matching purposes, can be any combination of ones and zeros.

PortRange

Specifies a port or a hyphen-separated port range. It takes this format:

```
Port["-"Port]
```

Port

Specifies a port number. This can be any value from 0 to 65535.

More information:

Specifying IPv6 Adresses (see page 80)
Appending Port Numbers to IP Addresses (see page 80)

Example Address and Port Filters

These examples show how to specify IP addresses and port numbers, including masked addresses.

Wildcard Example

```
<element value="*"/>
```

The NBA applies a filter to packets sent to or from any IP addresses and any ports.

Note: This is the only supported usage for wildcards when specifying address and port filters. For example, you *cannot* specify:

```
<element value="*:10"/>
```

Address Only Examples

```
<element value="10.0.1.53"/>
```

The NBA applies a filter to packets sent to or from IP address 10.0.1.53.

```
<element value="10.0.1.53-10.0.1.80"/>
```

The NBA applies a filter to packets sent to or from IP addresses 10.0.1.53 to 10.0.1.80.

```
<element value="fe80::-fe81::"/>
```

The NBA applies a filter to packets sent to or from IP addresses fe80:0:0:0:0:0:0:0 to fe81:0:0:0:0:0:0:0.

Address and Port Examples

```
<element value="192.160.0.5:10"/>
```

The NBA applies a filter to packets sent to or from IP address 192.160.0.5, but only if the packets are sent to or from port 10.

```
<element value="192.160.0.5:128-1023"/>
```

The NBA applies a filter to packets sent to or from IP address 192.160.0.5, but only if the packets are sent to or from ports 128 to 1023.

```
<element value="10.0.0.5-10.0.3.250:100"/>
```

The NBA applies a filter to packets sent to or from IP addresses 10.0.0.5 to 10.0.3.250, but only if the packets are sent to or from port 100.

```
<element value="[fe80::]-[fe81::]:80"/>
```

The NBA applies a filter to packets sent to or from IP addresses fe80:0:0:0:0:0:0:0 to fe81:0:0:0:0:0:0:0, but only if the packets are sent to or from port 80.

Note: Enclose the IPv6 addresses in brackets when you specify a port number or range.

Masked AddressExamples

`<element value="123"/>`

The NBA uses an implied mask to apply a filter to packets sent to or from IP addresses 123.0.0.0 to 123.255.255.255. In binary form, this gives the following values. Masked bits are shown in italics:
01111011.00000000.00000000.00000000

to
01111011.11111111.11111111.11111111

`<element value="10/7"/>`

The NBA applies a filter to packets sent to or from IP addresses 10.0.0.0 to 11.255.255.255. In binary form, this gives the following values. Masked bits are shown in italics:
00001010.00000000.00000000.00000000

to
00001011.11111111.11111111.11111111

`<element value="10.64/12:128-1023"/>`

The NBA applies a filter to packets sent to or from IP addresses 10.64.0.0 to 10.79.255.255. But note the filter is only applied to packets sent from or to ports 128 to 1023.

In binary form, this gives the following values. Masked bits are shown in italics:
*00001010.0100*0000.00000000.00000000

to

*00001010.0100*1111.11111111.11111111

Chapter 10: Importing NBA Events

This section contains the following topics:

Importing Events Overview

Note: Importing events is not the preferred ingestion method. More commonly, the NBA runs in 'output to socket' mode, passing items to CA DataMinder policy engines for analysis and capture.

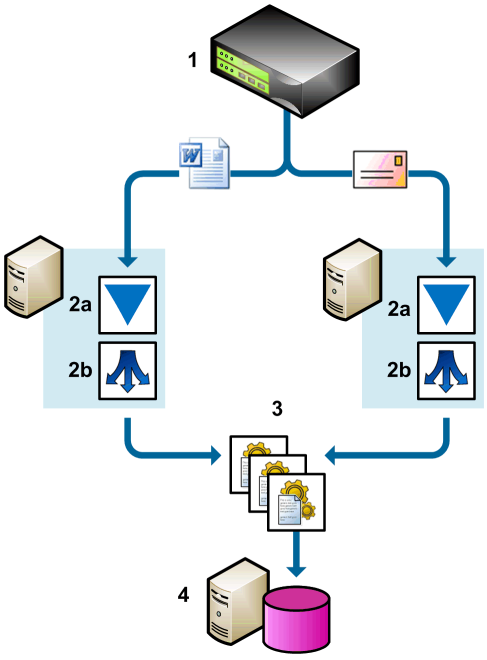
The CA DataMinder Import Policy utility connects Event Import to policy engines in order to apply triggers to emails or files as they are imported. If you run the NBA in 'output to disk' mode, you can use Import Policy to apply policy 'after the event' to any emails or files imported from the NBA. For example, you can categorize, or apply smart tags to, IM conversations or emails sent using Hotmail or Yahoo! For details about Import Policy, see the *Archive Integration Guide*.

The following sections describe the steps needed to set up Import Policy jobs for the NBA:

1. Specify user accounts for NBA import operations.
2. Set up Event Import. Specifically, you must configure the file and email import operations.
3. Verify that the necessary CA DataMinder user policies are correctly set up.

   User policies are described in Apply User Policy to NBA Events (see page 131).
4. Verify that machine policy settings for your policy engines are correctly set up.

   These settings are described in the Policy Engines chapter of the Platform Deployment Guide.



Example NBA import policy operations

Two Import Policy servers are deployed (**2**). One ingests files; the other ingests emails. In each case, Event Import (**2a**) imports items from the NBA (**1**) and passes these events via a hub (**2b**) to a shared pool of policy engines (**3**) for processing. The resulting events are replicated to the CMS (**4**).

Filename Formats for Captured Data

General files in the \files subfolder and EML files in the \mails folder use the following filename formats.

Emails

`sssssssss_n_sourceIP_destIP_type_.eml`

Example:
`0000000520_4_201.86.52.4_10.0.0.13_SMTP_.eml`

Where:

sssssssss

Specifies a serial number used to ensure the filename is unique

n

Specifies the CPU on the NBA that the file was captured on.

On Bivio 7000 appliances, **n** can be 1c0 to 6c1.

On the Linux Server Platform, **n** can be 1 to 8.

sourceIP

Specifies the source IP address of the file.

destIP

Specifies the destination IP address of the file.

type

Is one of the following:

AOLMAIL	AOLMAIL-SSL
CHAT-AIMICQ	
CHAT-JABBER	
CHAT-MSN	
CHAT-SIP	
CHAT-YAHOO	

DELTASYNC-RECV	DELTASYNC-RECV-SSL
DELTASYNC-SEND	DELTASYNC-SEND-SSL
GMAIL	GMAIL-SSL
HOTMAIL	HOTMAIL-SSL
HTTP-POST	HTTP-POST-SSL
POP3	POP3-SSL
SMTP	SMTP-SSL
YAHOONEW	YAHOONEW-SSL

Files

`sssssssssss_n_sourceIP-destIP_type_filename`

Example:

`4C3C792BC2_2c0_au.download.windowsupdate.com(130.119.248.209)_130.119.44.131_HTTP-GET_windows-kb890830-v3.9-d...61c5fada43a2f8788d42.exe`

Where:

sssssssss

Is a serial number used to ensure the filename is unique

n

Specifies the CPU on the NBA that the file was captured on.

On Bivio 7000 appliances, **n** can be 1c0 to 6c1.

On the Linux Server Platform, **n** can be 1 to 8.

sourceIP

Specifies the source machine name and, in brackets, the file's source IP address. If the machine name is not available, only the IP address is used.

destIP

Specifies the destination machine name and, in brackets, the file's source IP address. If the machine name is not available, only the IP address is used.

type

Is one of the following:

AOLMAIL-ATTACH	AOLMAIL-ATTACH-SSL
CHAT-SKYPE	
FILE-AIMICQ	
FILE-MSN	
FILE-YAHOO	
FTP-GET	
FTP-PUT	
GMAIL-ATTACH	GMAIL-ATTACH-SSL
HOTMAIL-ATTACH	HOTMAIL-ATTACH-SSL
HTTP-GET	HTTP-GET-SSL
HTTP-POST	HTTP-POST-SSL
HTTP-URL	HTTP-URL-SSL

NNTP-GET	
NNTP-POST	
SMB	
YAHOO-ATTACH	YAHOO-ATTACH-SSL

Specify User Accounts for NBA Import Operations

Before you can run an Import Policy operation to import events from the NBA, confirm that Event Import is using the correct account logon to the CMS. If required, you can also create a CA DataMinder user account explicitly for the NBA to allow policy to be applied to imported files.

Create 'NBA' CA DataMinder User

This step is optional. By default, policy engines apply the Default Policy For Files to any files imported from the NBA. See the Deployment guide for details; search the index for 'Default Policy For Files setting'.

But if you want to apply a custom policy, with tailored Data In Motion triggers explicitly set up for files imported from the NBA, you can create a custom CA DataMinder user account (for example, NBA Policy User) and assign an SMTP email address to this account. You can then tailor this account's user policy to apply triggers to files imported from the NBA. You then need to assign this email address to the ImpFile.PolicyParticipant parameter. The policy engine subsequently maps the specified address to the custom CA DataMinder user account.

More information:

Import Parameters (see page 195)

Logon Requirements for CMS

You need to securely cache the CMS logon credentials. Event Import needs to log on to the CMS as a CA DataMinder administrator. To avoid storing the administrator credentials in import.ini (which would represent a security loophole), you can configure wgnimpsv.exe to securely cache the credentials itself.

To do this, use the following command line syntax to open a new command window where you can enter a valid username and password:

```
wgnimpsv -setcredentials
```

Logon Requirements for Event Import

Files and messages captured by the NBA are imported from pickup folders.

Wgnimp.exe

If you run wgnimp.exe to import this data, you must log on to the Event Import host machine using an account that has administrative rights to access these pickup folders.

If you run command line import jobs, you will need to set the CMS credentials in the import.ini file.

Wgnimpsv.exe

If you want to use the Event Import service, wgnimpsv.exe, to import this data, you must ensure the service logon account has administrative rights to access these pickup folders.

By default, the Event Import service logs on as LocalSystem. Also, it is installed as a 'Manual' startup type so you must change this to an 'Automatic' startup type as soon as you have created your import.ini configuration file.

Note: For full details, see the *Archive Integration Guide*; search the index for 'Event Import, logon requirements'.

Set Up Event Import

If you run the NBA in passive mode, you can import items from the NBA onto the CMS. But first you must set up pickup folders. Then you must set up Event Import to handle concurrent file and email importing:

Pickup folders

You must set up a mechanism to regularly move the content of the \files and \mails subfolders in the NBA FTP folder to disk locations accessible to Event Import machines. This guide refers to these locations as pickup folders.

Concurrent file and email importing

A single instance of Event Import cannot import both files and emails. How you set up Event Import to handle this depends on whether you want to apply policy to imported data:

Import without applying policy

If you only want to import data onto the CMS, without applying policy, you must set up two instances of the Event Import service running on the same host machine. One instance imports files and the other imports emails. For guidelines on setting up dual instances of the Event Import service (and creating separate import configuration files and disabling the default instance), contact CA Support at: http://ca.com/support.

Import and apply policy

If you want to apply policy to files and emails imported from the NBA, we recommend that you deploy two separate Import Policy servers. This provides maximum flexibility when deploying policy engines to process imported files and emails.

More information:

Set Up Dual Import Policy Servers (see page 191)
Passive and Active Mode (see page 12)
FTP Folders and Files (see page 73)

Set Up Dual Import Policy Servers

We recommend that you deploy two Import Policy servers in hub mode, one to process imported files and the other to process imported emails. In hub mode, each server hosts Event Import and a hub. In turn, the two hubs pass imported files and emails to a shared pool of policy engines. This setup allows maximum flexibility to distribute the policy processing load and to add additional policy engines if the volume of data captured by the NBA increases.

To set up Import Policy in hub mode, you use the Engine.UsePolicyEngineConnector=Hub import parameter. Each Event Import instance must have its own version of the import.ini file, one configured to import emails and the other to import files.

For details about Import Policy, see the *Archive Integration Guide*.

More information:

Import Parameters (see page 195)
Example import.ini for Emails (see page 193)
Example import.ini for Files (see page 194)

Alternative Import Policy Configurations

You can deploy a simpler Import Policy configuration. However, these are less flexible in terms of data processing capacity. For example, you can deploy two Import Policy servers running in direct mode (each instance of Event Import passes events directly to a single local policy engine), but this configuration does not allow you to add extra policy engines if NBA data volumes increase.

Similarly, you can deploy a single Import Policy server hosting two instances of the Event Import service (one to import files and the other to import emails). Both instances feed imported items to the same local policy engine for processing. However, because all events imported from the NBA must be processed by a single policy engine, this significantly restricts ingest rates. Nevertheless, this configuration may be appropriate for testing purposes.

Import Failures

Because Event Import is set up for continuous import operations and the File.DeleteAfterImport parameter is set to Yes, any files that fail to be imported are moved into a \Failed subfolder on the Event Import host server.

Note: For details, see the *Archive Integration Guide*. Check the File Handling Parameters section in the Event Import chapter.

Import Configuration Files

You configure NBA import operations with parameters in an import configuration file. By default, Event Import looks for a configuration file named import.ini, stored in the \Import subfolder in the CA DataMinder installation folder on the Event Import host machine.

The example configuration files are set up to import emails and files.

Note: After wgnimpsv.exe has started, any subsequent changes to the .ini import configuration file will only take effect after you restart the service (or service instance).

Example import.ini for Emails

This example import configuration file is set up to import emails (saved in EML files) captured by the NBA. Only the key parameters are included.

```
import.type=eml
    # Specifies an EML import operation.
Engine.UsePolicyEngineConnector=Hub
    # Imported emails are distributed via a local hub across multiple policy engines.
Engine.loglevel=2
    # Sets the logging level for import operations. Level 2 means 'only log errors
    and warnings'.
EMail.EventDateFromEMail=yes
    # Sets the timestamp in imported emails to be the send time and date in the emails.
File.Pathspec=\mails_from_NBA
    # Specifies the pickup folder that you copied captured emails to. You copy captured
    emails from the \mails subfolder in the NBA FTP folder.
File.IncludeSubdirs=no
    # The NBA does not write captured emails to subfolders below the \mails subfolder,
    so Event Import does not need to search subfolders.
File.ContinuousInput=yes
    # Specifies continuous importing. This is essential when using the wgnimpsv.exe
    service.
File.DeleteAfterImport=yes
    # Source EML files are deleted after being successfully imported. This parameter
    is mandatory if continuous importing is also specified.
```

Note: Use '#' characters to specify comments in import.ini.

More information:

Import Parameters (see page 195)

Example import.ini for Files

This example import configuration file is set up to import files captured by the NBA onto your CMS. Only the key parameters are included.

```
import.type=file
    # Specifies a file import operation.
Engine.UsePolicyEngineConnector=Hub
    # Imported files are distributed via a local hub across multiple policy engines.
Engine.loglevel=2
    # Sets the logging level for import operations. Level 2 means 'only log errors
    and warnings'.
File.Pathspec=\files_from_NBA
    # Specifies the pickup folder that you copied captured files to. You copy captured
    files from the \files subfolder of the NBA FTP folder.
File.IncludeSubdirs=no
    # The NBA does not write captured files to subfolders below the \files folder,
    so Event Import does not need to search subfolders.
File.ContinuousInput=yes
    # Specifies continuous importing. This is essential when using the wgnimpsv.exe
    service.
File.DeleteAfterImport=yes
    # Source files are deleted from the NBA after being successfully imported. This
    parameter is mandatory if continuous importing is also specified.
ImpFile.PolicyParticipant=nba.objects@unipraxis.com,yes
    # Specifies which user policy is applied to imported files. The specified email
    address must be associated with a CA DataMinder user.
ImpFile.EventDateFromFile=yes
    # Sets the timestamp in imported files to be the time and date when they were
    captured by the NBA.
ImpFile.SourceIsNBA=yes
    # Flags all imported files as NBA events.
```

Note: Use '#' characters to specify comments in import.ini.

More information:

Import Parameters (see page 195)
Set Up Event Import (see page 190)

Import Parameters

This section provides summary details about key parameters used in NBA import operations. Details of all available import parameters are available in the *Archive Integration Guide*; search for 'parameters: Event Import'.

Import.Type=EML or FILE

Important! This parameter is mandatory.

This parameter specifies which types of file to import from the NBA. Set it to:

- EML to import .EML Internet Mail files and IM conversations, or
- FILE to import any type of files (typically text-based documents, including email attachments and downloaded files).

Engine.UsePolicyEngineConnector=Hub, Yes or No

For Import Policy jobs only.

Defaults to No. This parameter specifies how to implement Import Policy. When running Import Policy jobs to import data from the NBA, we strongly recommend that you set this parameter to Hub. The available options are:

Hub

Event Import passes imported NBA events to multiple policy engines via a policy engine hub (the policy engine connector). Running Import Policy jobs in hub mode ensures that you have sufficient capacity to process high volumes of data imported from the NBA.

Yes

Event Import passes NBA events directly to the local policy engine. This mode is not appropriate when importing data from the NBA.

No

Event Import stores NBA events in the local database without applying policy.

Engine.LogLevel=<number>

Defaults to 2. This determines the level of logging for the NBA import process. The default level only logs errors or important system messages.

Log entries are written to the evtimport_<instance name>_<date>.log file, where <instance name> is the name set by the Event Import service wgnimpsv.exe and <date> is the date and time when the log file was created; find this file in the system\data\logs folder.

EMail.EventDateFromEMail=Yes or No

Defaults to Yes. This parameter specifies where the capture date assigned to *imported events* is set from. If this parameter is set to:

Yes

The timestamp reflects the time and date in the email. It is based on the delivery time or time sent. If the email does not contain the delivery time or time sent, Event Import sets the timestamp to the time of import.

No

The timestamp reflects the time of import.

File.Pathspec=<file path>

This parameter specifies a fully-qualified path to the source folder for the import operation. This must be the pickup folder containing captured files or EML emails. (Before importing, you must previously have copied these items into the pickup folders from the \files and \mails subfolders of the NBA FTP folder.) For example, set this parameter to:

Files

\files_from_NBA

Emails

\mails_from_NBA

File.IncludeSubdirs=Yes or No

Defaults to No. This parameter specifies whether to search for matching files in subfolders below the source folder specified by File.Pathspec (see above). For NBA import operations, this parameter **must** be set to No.

File.ContinuousInput=Yes or No

Defaults to No. This parameter specifies whether Event Import repeatedly scans for and imports files specified by File.PathSpec or whether it shuts down after the input folders and files have been processed. Continuous import is necessary when Event Import is running as a service and perpetually scanning an input directory. For NBA import operations, we recommend you run continuous import operations.

When this parameter is set to **Yes**:

- File.IncludeSubdirs (see above) is invalid unless set to No.
- File.DeleteAfterImport (see below) must be set to either Yes or Always. The import operation will fail if this parameter is set to No.

File.DeleteAfterImport=Yes, No or Always

This parameter specifies whether to delete the source files after an import operation. It can be set to:

Yes

This is the recommended setting for NBA import operations. The source file is only deleted after a successful import operation. Handling for import failures is described below.

No

The source file is not deleted after import. This option is not supported if File.ContinuousInput=Yes.

Always

The source file is always deleted, whether the import succeeds or not, and even if File.ContinuousInput=Yes (see above).

If File.DeleteAfterImport=Yes, any files that fail to be imported are moved into a \Failed subfolder.

First, Event Import creates a new subfolder below the \mails_from_NBA or \files_from_NBA folders. This subfolder has the same name as the Event Import host machine. Event Import then creates the \Failed subfolder below this host machine subfolder.

Example: If Event Import on machine UNI-KEEGAN fails to import an email, it moves the EML file into this folder:
`\files_from_NBA\UNI-KEEGAN\Failed`

ImpFile.EventDateFromFile=Yes or No

Defaults to Yes. This parameter specifies how the capture date assigned to imported files is determined. If this parameter is set to:

Yes

The timestamp reflects the time and date when the file was last modified.

No

The timestamp reflects the time of import.

ImpFile.PolicyParticipant=<email address>, Yes or No

This parameter identifies which user policy to apply to imported files. It must be set to an email address followed by a comma and Yes or No. This address **must** match an address associated with a CA DataMinder user (as listed in the User Properties dialog in the Administration console). You need only specify this parameter if importing files as part of an Import Policy operation.

For NBA import operations, you may prefer to set up a dedicated 'NBA user' account on the CMS with its own email address. When the policy engine processes a file imported from the NBA, it maps this email address to the NBA user account and applies that account's policy.

The Yes or No option determines whether this 'policy user account' is added to the list of event participants (in this case, the users associated with the imported file). If set to:

Yes

The specified account is added to the list of event participants. Choose this option if you want to apply a specific user's policy to the imported file and associate that same user with the resulting file event.

No

The specified account is **not** added to the list of event participants. You typically choose this option if the 'policy user account' is not a real person, but simply an account that you use to apply a specific set of policy triggers. An 'NBA user' account falls into this category.

ImpFile.SourceIsNBA=Yes or No

Defaults to No. Use this parameter to explicitly flag imported file events as being captured by the NBA. This enables you to search directly for NBA events in the iConsole or Data Management console. If this parameter is set to:

Yes

All imported files in the current job are flagged as NBA file events.

When you search for file events in the iConsole or Data Management console, NBA file events are identified in the Event Type column by  icons and described as ' A file moving over the network'.

No

Imported files are not differentiated by import or capture source. When you search for file events, all file events (including those captured by the File Scanning Agent or imported from Windows machines) are represented in the Event Type column by  icons and described simply as 'File'.

More information:

Set Up Dual Import Policy Servers (see page 191)
Set Up Event Import (see page 190)
Machine ID as Stored File or IM Event Participants (see page 134)

Chapter 11: Technical Information

This chapter describes diagnostic tools that you can use to monitor and check the NBA.

This section contains the following topics:

SNMP Support

The NBA supports Simple Network Management Protocol (SNMP). This allows administrators to monitor the NBA for conditions that require urgent attention. In particular, administrators can use their preferred SNMP manager to monitor CPU statistics.

About SNMP

SNMP enables network devices to exchange management information. An SMNP agent runs on the NBA and reports system information such as CPU statistics to a remote SMNP manager such as HP Openview. For example, using SNMP you can check the latest number of frames flagged for analysis by NBA filters.

MIBs

The device information that can be queried using SNMP is defined in a Management Information Base (MIB), in the form of a hierarchical collection of managed objects.

The NBA provides MIBs that you must load into your preferred SNMP manager (six on Bivio 7000 appliances; four on the Linux Server Platform).

Using object details defined in these MIBs, an SNMP manager can retrieve the latest statistical and status data for the NBA.

Traps

SNMP agents can also send traps (that is, notification messages) to administrators, advising them when specific conditions or events occur. For the current NBA release, traps can indicate such information as hardware and application status changes.

When you enable SNMP on the NBA, you need to specify a sink address for traps. This is typically the IP address of the machine(s) hosting your preferred SMNP manager.

More information:

NBA Traps (see page 202)

What SNMP Data Is Available for the NBA?

The NBA MIB (CADLP-NBA-MIB.mib) includes the orchNBAStatusStatsTable object. The table referenced by this SNMP object includes a range of statistical and status data for the NBA. For example, the table shows the status of the data inspection ports and statistical data such as the number of packets seen by the NBA and the number of data streams (that is, whole objects) detected by each of the NBA's CPUs.

Descriptions of available SNMP objects, including column headings, are in CADLP-NBA-MIB.mib.

Configure SNMP for the NBA

Before you enable SNMP on the NBA, you need to configure the system name, contact and location details and 'community' authentication details.

To configure SNMP for the NBA

1. Log on to the NBA console and browse to the SNMP screen.
2. Go to the SNMP Configuration section and provide the following details:

   SNMP Monitoring

   Use this setting to enable or diable SNMP monitoring.

   SNMP Write Access

   By default, Write access to objects in the SNMP MIB is disabled and you cannot reset statistics.

   Use this setting to enable Write access to objects in the SNMP MIB.

   System Name

   This is a read-only property that identifies the NBA.

   On the Linux Server Platform, the System Name is identical to the system's DNS name.

   System Contact

   This is a text string that identifies the person or team in your organization with responsibility for maintaining the NBA.

   System Location

   This is a text string that identifies the NBA's location within your organization.

   Read-only community

   This is a text string (in effect, a password) used to authenticate the SNMP manager when it reads object data from the SNMP agent on the NBA. The default value is 'public'.

   Read-write community

   This is a text string (a password) used to authenticate the SNMP manager when it sends instructions to the SNMP agent (for example, to remotely configure the NBA). The default value is 'private'.

3. Click Apply to store the SNMP configuration changes.

Load MIBs for NBA

Your SNMP manager requires two CA DataMinder MIBs in order to recognize and understand the NBA-specific objects and two Bivio MIBS to recognize device-specific objects. You need to load these four MIBs into your SNMP manager. You can download them directly using the NBA console.

Two NET-SNMP MIBs are also provided. You need these if your SNMP manager does not already have them installed.

To load the MIBs

1. Log on to the NBA console and browse to the SNMP screen.
2. Go to the MIB section and click the download hyperlink. This will download a mib.tar.gz file to the location you specify. The resulting .tar file contains:
   - CADLP-MIB.mib
   - CADLP-NBA-MIB.mib
   - (Bivio 7000 appliance only) BIVIO-MIB.mib
   - (Bivio 7000 appliance only) BIVIO-SERVICES-MIB.mib
   - NET-SNMP-MIB.txt
   - NET-SNMP-AGENT-MIB.txt
3. Load and compile these MIBs into your preferred SNMP manager.

Object OIDs

All objects defined in the:

- CADLP MIBs contain the enterprise OID (object ID) prefix:

  `1.3.6.1.4.1.13503`
- BIVIO MIBs contain the enterprise OID prefix:

  `1.3.6.1.4.1.8983`

These OID prefixes are assigned by IANA (Internet Assigned Numbers Authority).

NBA Traps

SNMP traps are notification messages that enable the NBA to notify administrators of significant conditions on the NBA. You view the traps using your SNMP manager.

Specify a Trap Destination

When you set up SNMP support on the NBA, you need to specify where the traps are sent to. Specifically, you need to supply the IP address of the machine hosting your SNMP manager.

To specify a trap destination

1. In the SNMP screen, go to the **Traps** section.
2. In the Sink Addresses field, specify the IP address of the target machine and then click Add. The new address is displayed in the console immediately above the Sink Addresses field.

   Note: You must specify an IP address; you cannot specify a machine name.
3. If required, you can add further machines. Use the Add and Remove buttons to modify the list of sink addresses.

Configuring Trap Severity

All the traps defined in these MIBs are classified with a severity of 'Info', 'Minor', 'Major' or 'Critical'. For more information, see the descriptions for the following objects in the relevant MIBs:

- bivioTrapSeverity in BIVIO-SERVICES-MIB.mib
- orchNBAEvObSeverity in CADLP-NBA-MIB.mib

We recommend that you enable 'Major' and 'Critical' traps but disable 'Info' and 'Minor' traps. To do this, browse to the objects defined for the branches below in your SNMP manager:

- CA DataMinder trap severities:

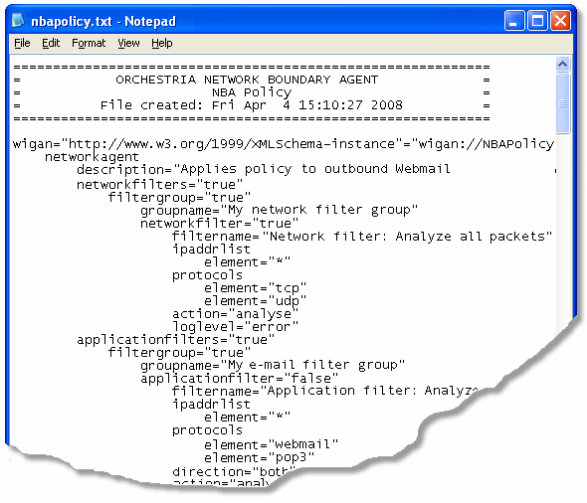
  1.3.6.1.4.1.13503.2.10.1.20.1
- Bivio trap severities:

  1.3.6.1.4.1.8983.1.1.3.1

NBA Policy Copied to a Text File

The policy settings used by the NBA are always copied to a text file for diagnostic purposes. This makes complex XML files more accessible to read and comprehend.

Specifically, when you edit nbapolicy.xml, all policy settings, including descriptive filter names, are automatically copied to nbapolicy.txt. Find this text file in the \config folder on the NBA.



Example nbapolicy.txt

Log Files

You configure logging operations by editing the tags in nbapolicy.xml, especially the <loglevel> tag. These tags determine the default level of logging for the packet capture process, plus logging levels for individual filters.

NBA log files are written to the \log, \log\err, and \log\out folders on the NBA. The .log and .txt files are useful in diagnosis of configuration and system problems. The .err and .out files provide diagnostic information for CA support staff.

The NBA maintains the following log files:

Activity log files

These are generated by the NBA service, with separate log files generated for each CPU on the NBA.

They record startup messages and errors, operation messages, and configuration changes recorded in nbaconfig.xml or nbapolicy.xml. For example, if the NBA goes offline, or if its operating mode is changed from active to passive, these details are recorded in the activity log files. The logging level is configurable through the <loglevel> tag in nbapolicy.xml.

Activity log files use this name format:

`<n>logMSG<date>.log`

Where <n> identifies the CPU and <date> is the date and time when the log file was created.

Agent Management log files

These are generated by the NBA Management Application service, which always runs on CPU-X.

They mainly record NBA policy changes written to nbapolicy.xml. For example, if a network or application filter is amended, this change is recorded in the agent management log file. The logging level is configurable through the <loglevel> tags in nbapolicy.xml.

The log files use this name format:

`<n>logAMM<date>.log`

Where <n> and <date> are as described above.

Statistics log files

These contain statistics about packet processing. Data is organized in a tabular layout. The log files use this name format:

`<n>logstats<date>.log`

Where <n> and <date> are as described above.

Stderr log files

These are written to the \log\err subfolder during startup of the NBA service and the NBA Management Application service respectively. The files contain error information and use these name formats:

<n><app><date>.err

Where <n> and <date> are as described above and <app> is the name of the NBA application that wrote the file.

Stdout log files

These are written to the \log\out subfolder during startup of the NBA service and the NBA Management Application service respectively. The files contain startup progress messages.

<n><app><date>.err

Where <n>, <date>, and <app> are as described above.

Collect Diagnostic Data

The NBA provides a diagnostics utility that collates all relevant log files, NBA status details and configuration files into a single .gz file. If you contact CA Technical Support, you must provide this .gz file .

To collect diagnostic data

You run this utility from the web console:

1. Log on to the NBA console and browse to the Administration screen.
2. Click the Collect Diagnostics button.

   This generates a file of the available NBA diagnostic, status, configuration and log data. This data is written to a .gz file in the \diag folder on the NBA.
3. Click the Diagnostics Bundle: Download button to download the most recent diagnostic file.

   Note: Downloaded diagnostic files are written to compressed .tar or .gz files. Depending on your browser, you may need to change a .tar file extension to .gz before reviewing the contents of a diagnostic file.

Health Screen

Use the NBA console to check the ‘health’ of the NBA CPUs and the agent management application.

The Health screen shows the summary state (OK, Warning or Error) and statistics for each CPU and the agent management application. The 'Overall' health reflects the 'least healthy' state of any individual CPU or the management application.

For example, the Health screen shows:

- The number of packets processed on each CPU since the statistics were last reset
- How many times the NBA application has restarted
- The state of connections to policy engines.

In general, if any statistic state is Warning or Error, this indicates a problem that requires attention.

To measure the NBA’s health over a specified period

1. Log on to the NBA console and browse to the Health screen.
2. Click the Reset button.

   This resets the health statistics back to zero and restarts statistics collection.

Check the NBA Status

You can use the NBA console to check the status of the various NBA components.

To check component status

Log on to the NBA console and browse the following screens:

- The Alarms screen shows any active alarms.
- (Bivio 7000 appliances only) The CPUs screen shows the status and temperature of all CPUs.
- (Linux Server Platform only) The Analyzers screen shows the status of the network analyzer processes.

Stop and Restart CPUs

Depending on the platform, you can either restart individual CPUs, or restart every CPU.

Note: We strongly recommend that you only do this under the guidance of CA Technical Support.

To stop and restart CPUs

1. Log on to the NBA console and browse to the Administration screen.
2. Use the Reboot button to restart all CPUs.

Before Powering Down

Before you shut down the NBA hardware, we recommend that you halt the NBA. This enables any processes running on the NBA to shut down in a controlled manner.

To halt the NBA

1. Log on to the NBA console.
2. Browse to the Administration screen
3. Click the Halt button.

Note: The Administration screen contains a Reboot button. We strongly recommend that you only use the Reboot button under the guidance of CA technical staff.

Policy Screen

The Policy screen in the NBA console allows you to make the following policy changes. These changes are copied automatically to the corresponding tags in the NBA policy file. (Some platforms do not support every setting.)

Policy Analyzer Names/IP Addresses

Use this setting to identify policy engines or the PE hub that the NBA will use when running in socket mode. Host machines are identified by their name or IP address and port. Use commas to separate multiple addresses. This setting corresponds directly to the <analyzeservers> tag in nbapolicy.xml.

Standby Analyzer Names/IP Addresses

Use this setting to identify the standby policy engines or PE hub that the NBA uses when running in socket mode. This setting corresponds directly to the NBA policy tag <standbyanalyzeservers>.

Connections per Network Analyzer

Use this setting to control the number of Policy Analyzers that each Network Analyzer can connect to.

If you are not using a policy engine hub, this setting must equal the number of configured Policy Analyzers.

Enterprise Mail Server DNS List

Use this setting to specify a DNS domain list that the NBA uses when checking the 'policy processed' status of emails. This setting corresponds directly to the <enterprisednslist> tag in nbapolicy.xml.

Logging Level

Use this setting to set the default level of logging for the packet capture process. Note that you can override this logging level for individual network filters or application filters. This setting corresponds directly to the 'Settings' tag <loglevel> in nbapolicy.xml.

HTTPGET File Types

Use this setting to specify a semi colon-separated list of filename extensions. This list determines which file types are processed by the NBA when they are fetched using the HTTPGET protocol.

More information:

XML Syntax: nbapolicy.xml (see page 155)
Settings Tags (see page 169)
Logging Tags (see page 173)

Status Details for CPUs

The CPUs or Analyzers screen in the NBA console shows summary statistics for each CPU or Network Analyzer process. Use these statistics to check whether frames are being received by the network filters and application filters, whether connections to policy engines are established, and the volume of traffic being captured and analyzed.

The CPUs screen shows the following columns of data:

Name

Shows the names of the CPUs. These include the management CPU, cpu-x, and the data processing CPUs:

- Bivio 7000 appliances: cpu-1c0 through cpu-6c1
- Linux Server Platform: NBA-1 through NBA-8

On Bivio 7000 appliances, cpu-xlr handles network packet load-sharing.

State

Shows the CPU status, such as Active, Inactive or Rebooting.

On Bivio 7000 appliances, the temperature is also shown for active CPUs. cpu-xlr handles network packet load-sharing.

UpTime

Shows the length of time that the CPU has been in an Active state.

PEAddr

Shows the IP address of the policy engine currently connected to the CPU. If no policy engine is connected, or if the NBA is trying to reconnect to a policy engine, this column shows ‘Unconfigured’ or ‘Reconnecting’ respectively.

PEPort

Shows the port number of the policy engine currently connected to the CPU.

MBits/sec

(Bivio 7000 appliances only) Shows the throughput rate of data being processed by the CPU.

Filtered

Shows the total number of Ethernet frames seen on the data inspection ports s0.e0 and s0.e1. This value shows the volume of traffic seen by the NBA. A static value (one that does not change when you refresh the screen) indicates that the NBA is incorrectly connected to your network.

The TCP and UDP frames seen by the NBA are passed to the network filters defined in the NBA policy.

Analyzed

Shows the number of Ethernet frames flagged for further analysis by the network filters and consequently assembled into whole objects.

In technical terms, these are frames that met the criteria of any network filter whose <action> is set to 'analyze' in the NBA policy.

If this number is static (it does not change when you refresh the screen), this indicates that the NBA's network filters are too restrictive (that is, they are ignoring or blocking frames instead of flagging them for further analysis).

FoundStrms

Shows the number of emails, Webmails and files detected by the NBA.

In technical terms, this is the number of object data streams detected by the NBA and passed to an application filter for further analysis.

SavedStrms

Shows the number of emails, Webmails and files passed to a policy engine for processing.

That is, these are decoded object data streams that met the criteria of any application filter whose <action> is set to 'analyze' in the NBA policy.

OpenStrms

Shows the number of outstanding emails, Webmails and files held by the NBA while it waits for the results of policy engine processing. In each case, the outcome of the policy processing will be either 'allow' or 'block'.

WriteQueue

The number of internal NBA buffers currently allocated to pending emails and files (as measured by the OpenStrms value).

The NBA uses these buffers to store data packets while they wait to be written to disk or passed to policy engines. They are set by the <numberofbuffers> and <sizeofbuffers> tags in the NBA configuration file, nbaconfig.xml.

LEDs Show Bypass Status

(Bivio appliances only)

LEDs on the back of the NBA indicate the status of the 'hardware bypass'. The bypass enables the NBA to continue forwarding data packets even if there is a sudden hardware failure or loss of power. The bypass configuration is pre-set and must not be changed.

There is a built-in hardware bypass for each pair of data inspection ports. If a hardware or power failure occurs, the NBA is reduced to a direct physical connection between the network cables attached to each port. The paired ports on Bivio 7000 appliances are:

- s0.e0 and s0.e1,
- s0.e2 and s0.e3,
- s0.e4 and s0.e5,
- s0.e6 and s0.e7,

The bypass status is indicated by LEDs on the back of the NBA. A pair of LEDs is assigned to each pair of data inspection ports. For each pair of LEDs:

- When the top LED is **on,** the ports are linked.

  If the bottom LED is also on, the bypass **is** enabled and data will continue to pass through the NBA even if a failure occurs.

  Note: The NBA is pre-configured so that the bypass is always enabled for the data inspection ports and the bottom LED is always on. Do not change this setting.

- When the top LED is **off**, the ports are in a failed state.

  If the bottom LED is also **off**, the system has lost power or is rebooting. If the bottom LED is **on**, the failed state was forced by an administrative command.

On Bivio 7000 appliances, all port pairs are used for data packets transiting through the device. During normal operations, *all* the hardware bypass LEDs are on.

More information:

Connect Bivio 7000 Ports (see page 29)

Bivio 7000 Technical Specifications

Physical Dimensions

Height: 3.5 inches, 8.9cm

Width: 17.0 inches, 43.2cm

Depth: 24.0 inches, 61cm

Weight: 45 lbs, 20kg

Environmental Specification

Operating Environment: 0-40°C

Relative Humidity: 10-90% non-condensing

Electrical Specification

AC input voltage: 100-240 VAC, single phase

AC input line frequency: 50/60Hz

Power input: 550W nominal

Safety

This product complies with the following safety standards:

- UL 60950/CSA C22.2 No. 60950 (US/Canada)
- EN 60950 (Europe)
- IEC60950 (International)

Electromagnetic Compatibility

This product complies with the following electromagnetic compatibility (EMC) regulations:

- FCC Part 15 Class A Subpart B (US/Canada)
- EN 55024 1998 (Europe)
- EN 55022 (CISPS 22) 1998 (Europe)
- VCCI Class A ITE (Japan)

Heat Output

Maximum:

- 7132 = 1300 BTU/hr (400 W)
- 7562 = 2000 BTU/hr (600 W)

Nominal:

- 7132 = 1000 BTU/hr (300 W)
- 7562 = 1500 BTU/hr (450 W)

Chapter 12: Troubleshooting

The following sections provide the information you need to solve problems that may arise when you use the NBA.

This section contains the following topics:

Bivio 7000 FTP IPv6 Connections

Symptom:

The Bivio 7000 does not support FTP connections using the IPv6 protocol.

Solution:

Use IPv4 addresses for FTP connections.

More Information:

Comply With FTP Folder Connection Requirements (see page 74)

Cannot Access HTTPS Sites With Cavium Coprocessor Enabled

Symptom:

Connectivity problems may occur when using the latest browsers (for example, Internet Explorer 10 or Firefox 19) to browse web sites using the SSL protocol (HTTPS). The issue has been observed when the SSL protocol is being decoded by CA DataMinder Network on the Bivio 7000 platform.

Solution:

The workaround is to configure CA DataMinder Network to use software rather than hardware SSL acceleration.

Follow these steps:

1. Create a file that has the name 'disablesslcoprocessor' inside the CA DataMinder Network 'config' folder. The content of the file is not relevant. Use the FTP server or alternatively via the command line and enter the following command:

   ```
   echo > /home/smb/config/disablesslcoprocessor
   ```

2. Restart the unit either by using the web management console and clicking Reboot on the Administration page, or by using the following commands on a shell prompt:

   ```
   nrsp stop nba
   nrsp start nba
   ```

Teredo Sessions Are Not Blocked

Teredo is an IPv6 transition technology, used to give computers access to remote IPv6 resources (for example, web sites) over an IPv4 network. The NBA can decode Teredo sessions and extract emails and files, but it does not attempt to block them if policy triggers. The NBA does not decode SSL sessions that use Teredo.

NBA filters can identify Teredo sessions using the IPv6 address range '2001:0::/32'. You can create a filter to ignore Teredo sessions, but if you set an application filter to Analyze or Prohibit them, the action is automatically changed to Monitor.

IPSec Is Not Decoded

How does the NBA handle IPSec frames?

1. If the NBA encounters IPSec frames, it will pass them through without analysis. The NBA does not decrypt IPSec frames.
2. If the NBA encounters IPSec NULL frames, it consolidates them into objects for analysis, but prevention is not possible.
3. If the NBA encounters an HTTPS session running over IPSec NULL, it cannot decode that session and passes it through without analysis.

Bivio 7000 Does Not Support SNMP Using IPv6

When you set SNMP parameters on the console, the file /etc/snmp/snmpd.conf is written. If you need to enable SNMP access over IPv6, configure it as follows:

Follow these steps:

1. Set all the parameters available on the console.
2. Log in as root.
3. Edit /etc/snmp/snmpd.conf as follows:
   a. Duplicate the two "com2sec" lines and change the new lines to start with "com2sec6".
   b. Replace the "agentaddress" line with
   `agentaddress udp:161,udp6:161`
4. Restart the SNMP agent using these commands:

```
nrsp stop nba_eventagentx
nrsp stop nba_svcagentx
nrsp stop snmpd
nrsp start snmpd
nrsp start nba_svcagentx
nrsp start nba_eventagentx
```

Note: If you have SNMP write access enabled, this can be done by running /home/nba/bin/webgui/gui/cgi-bin/snmpconfigedit.sh

IP Addresses in Captured Data do not Match Workstation Addresses

If the network location of your NBA is inappropriate, the machine IP addresses added to the filenames for captured files and emails may not match the workstation IP addresses for the associated users.

Specifically, the NBA must **not** be connected between your corporate firewall and the Internet. It needs to listen to traffic **before** the firewall or other device that uses NAT (Network Address Translation) to hide local addresses from the Internet.

NBA Applies Policy to Incomplete Data Streams

If a data stream is missing one or more data packets (or frames), the NBA will be unable to fully reassemble the email or file object. This situation can occur if the NBA is connected to a SPAN port on an Ethernet switch for passive mode operations, and the switch fails to pass all the frames to the NBA.

If you want to stop the NBA processing incomplete data streams, set the <capturepartialobjects> tag to false in the NBA policy. For details, see <capturepartialobjects>.

More information:

Passive Mode (see page 12)

Some Files and Emails are Not Captured

If the NBA is capturing some files and emails but missing others, check the following:

More information:

Data Not Captured After Changing Filters (see page 219)
Data Not Captured After Rebooting the NBA (see page 219)
Is All Network Traffic Being Passed to the NBA? (see page 219)
Is the Volume of Network Traffic Too High? (see page 219)
Does the Network Traffic Contain Large Frames? (see page 220)
Attachments in Forwarded Webmails (see page 220)

Data Not Captured After Changing Filters

Changes to filter IP addresses and actions take place immediately and can result in some files and emails not being captured.

Take care when changing the NBA's network and application protocol filter settings while sessions are in progress: The NBA tracks the start and end of TCP sessions to reassemble the files and emails that are sent inside decoded application protocols.

For example, adding a filter to capture Instant Messaging might not capture any messages until users log out from the messaging client and then log back in. Similarly, SSL sessions that are in progress while you enable decoding are not decoded; SSL sessions that are being decoded while you disable decoding will stall and need to be restarted in the browser.

Data Not Captured After Rebooting the NBA

The NBA tracks the start and end of TCP sessions in order to be able to reassemble the files and emails sent inside decoded application protocols. If the NBA is rebooted so that it loses track of sessions, some files and emails will not be captured.

For example, if a user logs on to their GMail account and the NBA is rebooted, emails sent after this point may not be captured. When the user logs back in to GMail, emails are captured again.

Is All Network Traffic Being Passed to the NBA?

When the NBA is connected to a data inspection port (also called SPAN ports or mirror sports) on an Ethernet switch, the port often fails to forward all traffic to the NBA when the network is very busy. Consequently, if an individual packet from an email or file stream is not forwarded, the NBA fails to capture the entire email or file stream.

Is the Volume of Network Traffic Too High?

If the NBA is connected at the wrong point in a network, it may ‘see’ more than just traffic destined for the Internet. Consequently, if individual packets from an email or file stream are missed, this will cause the entire stream capture to fail.

Note: You can filter this traffic by exempting specific IP addresses or port numbers, though this may not reduce traffic volumes sufficiently to solve the problem.

If the NBA is seeing too much traffic, the evidence will be in the form of dropped packets.

To check for dropped packets (Linux Server platforms with Napatech card only)

1. Start a command shell either locally on the server or remotely over SSH.
2. Enter the following commands:

```
cd /opt/napatech3/bin
./monitoring
```

3. Verify that this counter shows zero:

```
Drop events      : 0x0000000000000000
```

To check for dropped packets (Bivio platform only)

1. Generate a diagnostics file.
2. From the resulting .gz file, extract cpu-info.txt.
3. Open cpu-info.txt and check the 'QFull Drop' counts for each CPU

   A non-zero figure confirms that the NBA is dropping packets.

Does the Network Traffic Contain Large Frames?

Typically, large frames are Ethernet frames with over 1518 bytes of payload. These packets are sometimes seen on corporate networks. The NBA is designed to process packets up to 9030 bytes long, but it has only been verified to process packets up to 1522 bytes. Packets larger than 9030 bytes will not be analyzed by the NBA.

To check whether the NBA is seeing very large packets (Linux Server platforms with Napatech card only)

1. Start a command shell either at the server's keyboard or over SSH.
2. Enter the following commands:

```
cd /opt/napatech3/bin'.
./monitoring'.
Type '1' to show the 'ExtRMON' screen.
```

3. Verify that the 8192-Max octets counter shows zero:

```
1519-2047 octets: 0x0000000000000000  2048-4095 octets: 0x0000000000000000
4096-8191 octets: 0x0000000000000000  8192-Max octets : 0x0000000000000000
```

Attachments in Forwarded Webmails

Because forwarded Webmails leave the attachment files on the Web server, the attachments are not transmitted over the network. This means that the files cannot be captured as attachments to the Webmail.

No Files or Emails are Captured

If the NBA fails to capture any data, the following steps can be useful when diagnosing the cause:

1. Confirm that the NBA is failing to detect any network traffic.

   In the NBA console, browse to the CPUs or Analyzers screen and check:

   Filtered column

   Shows the number of Ethernet packets seen by each CPU. If the packet counts do not change when you refresh the page, this indicates that the NBA is not detecting any network traffic.

2. Check that the network and application filters are not inadvertently eliminating too much network traffic. The activity log files list the active filter details.

   In the NBA console, browse to the CPUs screen and check:

   Analyzed column

   Shows the number of Ethernet packets that passed through the network filters for further analysis.

   If this packet count does not change, or changes very little, when you refresh the screen, the network filters may be removing too much traffic.

   FoundStrms column

   Shows the number of data streams found in the packets that passed through the network filters.

   If this stream count does not change, or changes very little, when you refresh the screen, the network filters may be removing too much traffic.

   SavedStrms column

   Shows the number of data streams sent to a policy engine for content analysis.

   If this stream count does not change, or changes very little, when you refresh the screen, the application filters may be removing too much traffic or the found streams may have missing packets.

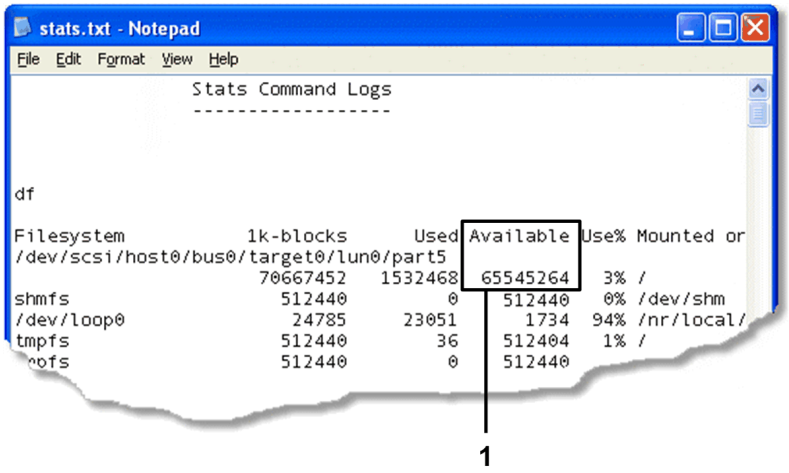
3. Check whether the NBA disk is full. To do this:

   a. Generate a diagnostics file.

   b. Extract stats.txt from the resulting .gz file.

   c. Open status.txt and check the available blocks count for the Filesystem.

The screenshot below shows an extract from status.txt. The high count for available blocks shows that the NBA disk is not full:



Example status.txt

1 Number of available blocks.

FTP File Transfers Fail to Complete

The NBA pauses a FTP file transfer while the data is being analyzed by a CA DataMinder policy engine.

If the analysis takes a long time, the FTP client may halt the transfer if its own timeout is exceeded. You may need to increase the FTP client timeout to prevent such transfer failures.

NBA Console Fails to Load

Symptom:

I have changed the speed or duplex settings of the Bivio 7000 management port. Now the NBA console fails to load. The network connection to the mgt0 interface does not respond.

Solution:

Setting a port at the wrong speed or duplex results in connection problems. Set the Bivio appliance to the same speed and duplex as the other end of the cable. If the other end is set to auto-negotiate, the appliance can lock up if you connect with a set speed; in this case, also set mgt0 to auto.

1. Log on to the Bivio as "admin" using the serial port.
2. Enter the following commands:

   ```
   configure interfaces
   set interface <interface> speed [TH]<auto|10|100|1000> duplex <full|half>
   commit boot
   ```

   The connection is restored.

More information:

If Bypass Relay Is Closed (see page 225)

Automated NBA RPM Package Install Process Stalls

Symptom:

I am using the automated RPM install process to install the NBA rpm package. The process stalls at the end user license acceptance prompt, waiting for manual input.

Solution:

Before running the rpm install process, set the environment variable 'I_ACCEPT_CA_LICENSE' to any value. This setting deactivates the manual license acceptance prompt. The automated installation continues without manual intervention.

More information:

Policy Engines Are Not Balancing Load (see page 224)

Bivio Expresslane Network Clashes With External Network

Symptom:

The internal Expresslane network in the Bivio has a network range that clashes with an external network in my organization (10.10.10.x).

Solution:

Choose a subnet address different from 10.10.10.x for your subnet.

1. Log on to the Bivio CLI console as user ‘admin’.
2. Choose a value different from 10.10.10.x for *<IP subnet>* . Enter the following commands.

```
configure system
set mccp subnet <IP subnet>
commit boot
exit
```

Policy Engines Are Not Balancing Load

Symptom:

I connected the NBA directly to Policy Engines rather than via a Hub component. The files and emails captured by the NBA are not evenly balanced to the Policy Engines.

Solution:

Change the ‘Connections per Network Analyzer’ setting on the NBA console Policy page. You can also adjust this value directly by updating the <policyenginespercpu> setting in nbaconfig.xml. The value of the setting should be equal to the number of configured Policy Engines.

More information:

Example Configuration File (see page 238)

If Bypass Relay Is Closed

(Applies to Bivio platforms *and* Linux server platform)

This section identifies configuration requirements and guidelines when the NBA is in a failed state and the LAN bypass relay is closed.

Maximum Cable Length

When the bypass relay is closed, the two network cables are electrically joined together to form a single cable. Make sure that the combined length of the two cables does not exceed the normal maximum length for a single LAN cable when the relay is operational.

Crossover or Straight Through Cables?

Consider whether to use crossover or straight through cables to make sure the network link continues to operate when the bypass relay is closed. This decision depends on the configuration of other equipment that the NBA is attached to.

Port Speeds

To minimize any network disturbance, the ports on the NBA and other equipment that it is attached to must all have the same speed and duplex settings to avoid auto-negotiation delays if an NBA failure occurs.

Note: We strongly recommend that you test both 'normal' and 'failed' bypass states before deploying the NBA.

More information:

NBA Console Fails to Load (see page 223)

No IP Address For Management Port (Linux Server Platform)

Symptom:

The Linux Server Platform installer configures network ports for managing the device with DHCP. If no DHCP server is available, these ports do not receive IP addresses.

Solution:

You can configure a port with static IP addresses using the Interfaces page of the NBA Console, however, the console requires that at least one port already has an IP address. If that is not the case, enter the following on the command line.

To configure a port with a static IP address

1. Choose the port name you wish to configure. Enter ‘ifconfig’ to discover available port names, for example, ‘eth0’.
2. Enter ‘vi /etc/sysconfig/network-scripts/ifcfg-eth0’.
3. Type ‘i’ to switch to character insert mode. Change the file to the following configuration:

   ```
   DEVICE="eth0"
   HWADDR="xx:xx:xx:xx:xx:xx"
   NM_CONTROLLED="yes"
   BOOTPROTO="none"
   IPADDR="192.168.0.12"
   NETMASK="255.255.255.0"
   GATEWAY="192.168.0.1"
   ETHTOOL_OPTS="autoneg on"
   IPV6INIT="yes"
   ONBOOT="yes"
   ```

   Replace "xx:xx:xx:xx:xx:xx" by the hardware address of the network card.
4. Type ESC to leave character insert mode. Type ‘:wq’ to save the file.
5. Type ‘service network restart’ to activate the network port.
6. Type ‘ifconfig’ and verify that the port now has an IP address.

Napatech Drivers

We recommend that use a Napatech high-speed network card drivers on your Linux servers. The following sections describe how to troubleshoot these drivers.

More information:

Collect Napatech Logs (see page 227)
Diagnose Napatech Driver Problems (see page 228)
Contact Your Napatech Distributor (see page 229)

Collect Napatech Logs

Follow this procedure to collect Napatech logs.

1. Use your terminal emulation application (such as PuTTY) to onnect to the NBA as root.
2. From a shell prompt, run this sequence of commands:
   ```
   cd /opt/napatech3/bin
   ./profiling
   ```
3. Copy the screen output from your terminal emulation application to a text file.
4. Enter ‘q’ to exit profiling.
5. From a shell prompt, run the next sequence of commands:
   ```
   mkdir /home/smb/log/napatech
   chmod 777 /home/smb/log/napatech
   dmesg >/home/smb/log/napatech/dmesg-log.txt
   ./ntlog >/home/smb/log/napatech/ntlog-log.txt
   cd ../config
   cp n* /home/smb/log/napatech
   cd /home/smb/log/napatech
   tar cvf napatech-info.tar *
   gzip napatech-info.tar
   ```
6. Send the napatech-info.tar.gz file and the 'profiling' text file that you created in step 3 to Napatech Support.

Diagnose Napatech Driver Problems

To diagnose Napatech driver problems, consult the Napatech log to discover whether the driver has successfully started.

To diagnose driver problems

1. Log in as root to the CentOS shell and change directory to the Napatech tools directory:
   ```
   cd /opt/napatech3/bin
   ```
   Note: If this directory does not exist, then the driver software is not installed correctly. Go through the installation process as detailed in the Linux Server Platform Software Installation (see page 51) chapter.
2. Copy the log to a temporary file:
   ```
   ./ntlog > currentlog.txt
   ```
3. Read the log and look out for the following:
   ```
   >>> Warning: Failed to initialize the API.
   >>>          Not able to read low level driver log
   >>>          Is the driver loaded?
   >>> Error: Failed to read log: "The log is not ready."
   >>> Error: Allocating memory for log buffer failed.
   ```
   This error tells you that the Napatech driver is not loaded.
4. Load the driver using the following command:
   ```
   service nbanic start
   ```
   Note: If this command returns "No Napatech cards, so not starting driver" then you do not have any suitable Napatech cards installed yet.
5. Refresh the log file.
   ```
   ./ntlog > currentlog.txt
   ```
6. Review the log using the 'less' command. You scroll using the Page Up/Page Down keys, and exit by pressing the 'Q' key.
   ```
   less currentlog.txt
   ```
7. Verify that the log contains the following lines:
   ```
   Adapter0 running with profile: "Inline"
   ***************************
   NTService is now operational.
   ***************************
   ```
   Verify that these log messages also appear on the CA DataMinder Network console in the Messages tab.

   The Napatech driver is now operational.

Contact Your Napatech Distributor

If you experience problems when installing the Napatech network card drivers, contact your Napatech distributor:

www.networkallies.com

1600 Osgood Street, Andover, MA 01845, USA

Tel: 978-486-0300; Fax: 978-486-0195

Appendix A: About nbaconfig.xml

This chapter provides a brief overview of the NBA configuration file, nbaconfig.xml. It contains various NBA operation and disk management settings. Under normal operating conditions, you do not need to edit this file. Its purpose is to allow CA technical staff to override settings in the NBA console or fine tune NBA disk management or the available memory buffers. But see the warning below. The file is in the \config folder on the NBA appliance.

Important! Edit the configuration tags in nbaconfig.xml **only** if instructed to do so by CA Technical Staff!

XML syntax: nbaconfig.xml

NBA configuration file nbaconfig.xml contains the following XML tags:

```
<networkagent>
    <debugflags>
    <settings>
        <online>
        <active>
        <capture>
        <objtypes>
        <httpgetfiletypes>
        <numberofbuffers>
        <sizeofbuffers>
        <policyenginespercpu>
        <diskfullpercent>
        <diskfullintervalsecs>
        <ssl>
            <validityperioddays>
            <commonname>
            <organizationname>
            <localityname>
            <provincename>
            <countryname>
            <ciphersuites>
```

Note: If nbaconfig.xml contains syntax errors, these are recorded in the Agent Management log file.

<networkagent>

Contains the <debugflags> and <settings> tags described below.

<debugflags>

Flags that control logging of additional debug information for use by software developers.

type

Always set to type="simpleEnumDebugFlags".

value

Always set to value="none".

<settings>

Contains the configuration tags described below.

<online>

Determines whether the NBA is online or offline. You can edit this tag directly, or you can use the NBA console to set the NBA's status. If you use the console, this XML tag is updated automatically. This tag is also supported in nbapolicy.xml

The Packet Processing switch in the Administration screen of the NBA console has the same effect as this policy tag.

This tag supports type and value attributes:

type

Always set to type="booleanType".

value

Can be set to:

- **value="true"**--The NBA is online.
- **value="false"**--The NBA is offline.

For the NBA to be online, this tag must be set to true in both nbapolicy.xml and nbaconfig.xml. That is, both must contain:

```
<online type="booleanType" value="true"/>
```

<active>

Determines whether the NBA is running in active or passive mode. You can edit this tag directly, or you can use the NBA console to set the mode. If you use the console, this XML tag is updated automatically. This tag is also supported in nbapolicy.xml.

The Stream Blocking switch in the Administration screen of the NBA console has the same effect as this configuration tag.

This tag supports type and value attributes:

type

Always set to type="booleanType".

value

Can be set to:

- **value="true"**--The NBA runs in active mode.
- **value="false"**--The NBA runs in passive mode.

For the NBA to be in active mode, this tag must be set to true in both nbapolicy.xml and nbaconfig.xml. That is, both must contain:

```
<active type="booleanType" value="true"/>
```

<capture>

This tag determines the NBA output location for captured emails and files. The NBA can output these emails and files via a socket connection to a policy engine or hub, or it can save them to local folders on the NBA.

The Output Mode setting in the Administration screen of the NBA console has the same effect as this configuration tag. If you can use the NBA console to set the NBA output mode, this XML tag is updated automatically.

This tag supports type and value attributes:

type

Always set to type="simpleNBAcaptureMode".

value

Can be set to:

- **value="socket"**--The NBA outputs captured items via a socket connection to a policy engine or hub.
- **value="disk"**--The NBA outputs captured items to local files and mails folders on the NBA.
- **value="socket and disk"**--The NBA outputs captured items via a socket connection *and* saves them to the local files and mails folders.
- **value="off"**--The NBA does not output any captured items.

<objtypes>

This tag specifies which protocols or 'channels' to capture. You can either configure the NBA to capture all channels or you can selectively add individual channels. The syntax is the same as for the <protocols> tag in application filters.

Notes:

- If a protocol is not included in the list of object types, the NBA does not look for that protocol when analyzing data packets. This enables more efficient packet analysis. You can therefore use this tag to optimize NBA packet processing.
- If you set the object types to ALL, the NBA can detect all protocols **including** HTTPURL. This is in contrast to the <protocol> tag in nbapolicy.xml. For example:

```
<objtypes type="stringListType">
    <element value="ALL"/>
</objtypes>
```

- If you change the object types, you must restart the NBA service for the change to take effect.

<httpgetfiletypes>

(Applies only to application filters that include the HTTPGET protocol)

Specifies a list of document types, identified by their file name extension and separated by semi colons. When a user downloads a document from a Web site, the NBA only captures document types that match types in this list. It ignores all other document types. Even if the file extension for the downloaded document does not match this list, the NBA can determine its file type from the content of the document.

On Bivio 7000 appliances, the HTTPGET File Types setting on the policy screen of the NBA console has the same effect as this configuration tag.

Default: The list includes Microsoft Office file types such as DOC, DOCX, PPT and XLS, plus other common document types such as TXT and PDF. For the full list, see the default nbaconfig.xml; find this in the \config folder in the NBA FTP folder.

Examples: Add '.XML' to the list of file types to capture RSS feed data when browsing the web. Add '.HTM' to the list of file types to capture web page content.

<numberofbuffers>

Defaults to 20,000. This tag specifies the number of memory buffers available to each NBA application processor. The NBA uses these buffers to store data packets while they wait to be written to disk. When these buffers are full, the NBA cannot accept further packets until buffers become available again. The NBA will therefore miss any packets arriving while the buffers are full.

The default buffer size is 8KB, so the default number of buffers guarantees that 160MB of data can be buffered by each NBA application processor before it needs to discard additional incoming data. The default buffer size is defined by the <sizeofbuffers> tag.

This tag supports type and value attributes:

type

Always set to type="numberType".

value

Specifies the number of buffers. For example, to specify 30,0000 buffers, set this attribute to:

```
<numberofbuffers type="numberType value="30000"/>
```

Note: Using memory buffers to decouple the data capture process from the process of writing data to disk minimizes the risk of the NBA losing data packets.

<sizeofbuffers>

Defaults to 8,000. This tag specifies (in bytes) the default size of the memory buffers on the NBA. The number of buffers available to the NBA is set defined by the <numberofbuffers> tag.

type

Always set to type="numberType".

value

Specifies the buffer size. For example, to specify 10KB buffers, set this attribute to:

```
<sizeofbuffers type="numberType value="10000"/>
```

<policyenginespercpu>

Defaults to 1. This tag specifies the number of concurrent policy engine connections for each NBA application processor.

type

Always set to type="numberType".

value

Specifies the number of connections This is normally set to 1.

Note: See the Troubleshooting section of this manual for more information on how to use this setting to improve load balancing (see page 224).

<diskfullpercent>

Defaults to 80. The minimum value is 50.

This tag specifies the critical level of used disk space on the NBA, measured as a percentage of total disk space.

If the amount of used disk space rises above the specified percentage (causing a critical shortage of free disk space), the NBA stops saving captured emails or files to local folders on the NBA, if configured to do so. The NBA's ability to save local copies resumes as soon as the level of used disk space falls back (so that free disk space recovers).

Data analysis is unaffected and continues as normal, even if used disk space rises above the critical threshold. For example, the NBA will continue to forward emails and files to policy engines for processing.

<diskfullintervalsecs>

Defaults to 15. This tag specifies how often the NBA checks the level of free disk space. This tag operates in conjunction with the <diskfullpercent> tag.

<ssl>

Contains the SSL Decode configuration tags described below.

<validityperioddays>

Specifies the number of days that the SSL Decode master certificate is valid for.

Default: 730.

type

Always set to type="numberType".

value

Specifies the number of days.

Note: This setting is only used when the master certificate is generated, which is after installation and after expiry of the certificate. Install Master certificates in every browser whose traffic needs to be decoded. After generating the certificates, distribute them to all client machines.

<commonname>
<organizationname>
<localityname>
<provincename>
<countryname>

Specify SSL Decode master certificate properties. When presented with a certificate for a web site, the client sees the commonname property in the 'signed by' field, and the remaining properties in the site's parent certificate. Customize these tags for your organization.

type

Always set to type="stringType".

value

Specifies the text for the master certificate.

Note: These settings are only used when the master certificate is generated, which is after installation and after expiry of the certificate. Install Master certificates in every browser whose traffic needs to be decoded. After generating the certificates, distribute them to all client machines.

<ciphersuites>

Specifies the set of allowed SSL Cipher Suites that will be used when connecting to SSL servers. Low strength ciphers are disabled and the NBA is set to choose the highest strength cipher available to it. For more details, see http://www.openssl.org/docs/apps/ciphers.html.

type

Always set to type="stringType".

value

Defaults to value="ALL:!ADH:!LOW:!EXP:@STRENGTH"

More information:

XML Syntax: nbapolicy.xml (see page 155)

Example Configuration File

An XML configuration file must adhere to the format shown below. This example configures the NBA to run in active mode, and outputs captured objects to policy engines via a socket connection. It only detects files uploaded to or downloaded from Web sites, targeting specific types of Microsoft Office files. It also specifies the amount of buffer storage available to the NBA and the free disk space threshold.

```
<?xml version="1.0" encoding="UTF-16"?>
<wigan xmlns:xsi="http://www.w3.org/1999/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="wigan://NBAConfig v1.0">
    <networkagent>
        <debugflags type="simpleEnumDebugFlags" value="none" />
        <settings>
            <online type="booleanType" value="true" />
            <active type="booleanType" value="true" />
            <capture type="simpleNBAcaptureMode" value="socket" />
            <objtypes type="stringListType">
                <element value="HTTPPOST" />
                <element value="HTTPGET" />
            </objtypes>
            <httpgetfiletypes type="stringType"
             value=";DOC;DOCX;XLS;XLSX;PPT;PPX" />
            <numberofbuffers type="numberType" value="20000" />
            <sizeofbuffers type="numberType" value="8000" />
            <policyenginespercpu type="numberType" value="1" />
            <diskfullpercent type="numberType" value="80" />
            <diskfullintervalsecs type="numberType" value="15" />
            <settings/>
            <ssl>
                <validityperioddays type="numberType" value="730" />
                <commonname type="stringType" value="CA DataMinder Network" />
                <organizationname type="stringType" value="CA Technologies" />
                <localityname type="stringType" value="Islandia" />
                <provincename type="stringType" value="NY" />
                <countryname type="stringType" value="US" />
                <ciphersuites type="stringType"
                 value="ALL:!ADH:!LOW:!EXP:@STRENGTH" />
            </ssl>
     </networkagent>
</wigan>
```

Index

L

M

N

O

P

Q

R

S

T

U

V

W

X