Managing the Analyzer

This chapter, which provides information for managing the Analyzer, contains the following sections:

- Accessing NetFlow Data Files—Explains why data stored on nonlocal file systems can be inaccessible to the Analyzer.
- Managing Analyzer Files and Directories—Describes directory maintenance tasks and provides tips on how to efficiently use disk space on your Analyzer host machine.
- Optimizing Analyzer Memory Use—Provides tips on how to efficiently use memory on your Analyzer host workstation.
- Controlling Analyzer Modules—Describes how to start, stop, and check the status of the modules comprising the Analyzer.
- Performance-Tuning Considerations—Explains how fast you can expect the Analyzer to run, given the volume of NetFlow data being processed and the platform on which the Analyzer is running.

Accessing NetFlow Data Files

The Display module of the Analyzer might not be able to display desired NetFlow data if that data is not stored on an accessible file system. For example, the DisplayServer can retrieve NetFlow data only from the following:

- Local disks attached to hosts on which the DisplayServer is running
- File systems with valid network file System (NFS) mounts to the host on which the DisplayServer is running

For NFS mounts to function properly in your Analyzer operating environment, such mounts must involve NetFlow operations using only the local disks of Solaris platforms. In other words, a Solaris-based system can read a disk only on another Solaris-based platform.

Note If the DisplayServer host uses an automounter facility, the facility can mount several shared file systems that may not be valid NFS mounting hosts. In this case, the DisplayServer might not be able to retrieve NetFlow data stored on such file systems.

If you store NetFlow data in a file system called “my_database_WS,” for example, you can issue the **showmount -e** command shown below to list the available NFS mounting hosts in your network that have NFS mounting permission for this file system:

```
$ /usr/sbin/showmount -e my_database_WS
 export list for my_database_WS:
/u0                                [list_of_hosts_having_permission]
/u1                                [list_of_hosts_having_permission]
```

You can run the DisplayServer and service user requests for NetFlow data stored on disks anywhere in the network, provided that such disks are associated with a platform of the same type (such as that being used by the “my_database_WS” file system).

To obtain valid NFS mounts to file systems not listed in the output of the **showmount** command, you will need to consult with your system administrator.

Note To optimize the performance of the DisplayServer, use file systems that are local to the host on which the DisplayServer is running.

Managing Analyzer Files and Directories

This section describes the files and directories used by the Analyzer and explains how to manage and maintain these storage entities.

Maintaining UtilityServer Output Directories

To monitor and maintain the UtilityServer, you should periodically perform the following tasks:

- Check the log files for errors
- Delete old log files

You can view the UtilityServer log files in the /opt/CSCOnfa/NFAUtility/logs directory. These log files are numbered sequentially, as indicated below:

- NFAU.log
- NFAU1.log
- NFAU2.log
- And so on

Each time the UtilityServer is started, a new log file is created.

In addition to the /opt/CSCOnfa/NFAUtility/logs directory, the UtilityServer writes output to the /opt/CSCOnfa/NFAUtility/data directory. This directory contains various data files used by the UtilityServer and the DisplayServer and is for internal use only by the Analyzer.

Maintaining DisplayServer Exported Files Directory

Periodically, you should delete files that are no longer needed from the DisplayServer /opt/CSCOnfa/NFAServer/exported_files directory.

Files exported from the DisplayServer when you invoke the Export option of the Files menu are stored in the following directory:

```
/opt/CSCOnfa/NFAServer/exported_files
```

Maintaining the DisplayServer Cache Directory

This section describes how to define parameters in the NFADS.resources file to limit the size and number of data files stored in the /opt/CSCOnfa/NFAServer/Cache directory.

The NoWait option of the DisplayServer enables you to:

- Request a background job
- Service a NetFlow command that generates voluminous output—The NoWait option enables the command output to be stored in the DisplayServer's /opt/CSCOnfa/NFAServer/Cache directory. Data can then be retrieved from this directory in smaller segments.

For instructions on editing the NFADS.resources file, see the "Customizing the NFADS.resources File" section on page 2-7 in Chapter 2.

The parameters in the NFADS.resources file that you can specify for managing the storage of NoWait command response files in the /opt/CSCOnfa/NFAServer/Cache directory are described briefly below:

- Max_Stored_NoWait_MB—Maximum aggregate disk space allotted for storing the NoWait command response files in the /opt/CSCOnfa/NFAServer/Cache directory.
- Max_Stored_NoWait_Files—Maximum number of NoWait command response files that can be stored in the /opt/CSCOnfa/NFAServer/Cache directory.
- Tgt_Stored_NoWait_Percent—Percentage of NoWait command response files that will be deleted from the /opt/CSCOnfa/NFAServer/Cache directory when the value of either the Max_Stored_NoWait_MB parameter or the Max_Stored_NoWait_Files parameter is exceeded.

The DisplayServer:

- Checks the state of the /opt/CSCOnfa/NFAServer/Cache directory upon completion of each NoWait command.

  If the value of either the Max_Stored_NoWait_MB parameter or the Max_Stored_NoWait_Files parameter is exceeded, the DisplayServer deletes files in chronological order until both the target storage value (in MB) and the target file count are reached.

- Multiplies the value of the Tgt_Stored_NoWait_Percent parameter by the value of both the Max_Stored_NoWait_MB parameter and the Max_Stored_NoWait_Files parameter to determine the amount of file space and the number of files, respectively, to be deleted.

  The file relating to the most recently completed NoWait command is never deleted, even if its size exceeds the value of the Max_Stored_NoWait_MB parameter. Hence, there is always at least one file in the /opt/CSCOnfa/NFAServer/Cache directory.

Optimizing Analyzer Memory Use

This section presents procedures that enable the Analyzer to use available memory resources efficiently.

Calculating Available Memory and Adjusting the MaxMB Value

This section describes how to calculate the logical memory capacity of the host on which Solaris software is running and how to fine-tune the system to use memory efficiently.

If your Analyzer host meets the recommended memory guidelines (256 MB of physical memory and 400 MB of free logical memory), you might not need to perform the procedures in the following section.

However, if your system does not meet the recommended memory guidelines, or if you experience Analyzer performance problems, perform the following procedure for your Solaris system software.

Calculating Available Memory for Solaris Software

Use the procedure in this section to calculate the available memory for Solaris software when all of the Analyzer modules (except the DisplayServer) are running.

For instructions on starting and stopping the DisplayServer, see the “Controlling Analyzer Modules” section on page 4-5.

To calculate the amount of logical memory (swap space) available on your Analyzer host, perform the following procedure:

Step 1 Run the vmstat program with “5” as an argument (which causes command output to be refreshed every 5 seconds). The format of this command and the output that it generates are shown below:

```
$ vmstat 5

 procs     memory            page      ...
 r b w   swap  free  re  mf pi po fr de...
 0 0 0  74312 51536   0  16 64 20 46  0... (this line yields no valid output)
 0 0 0 420056 223296  0   1  6  0  0  0...
 0 0 0 420056 223280  0   0  4  0  0  0...
 0 0 0 420056 223272  0   0  0  0  0  0...
 0 0 0 420056 223272  0   0  0  0  0  0...
 0 0 0 420056 223264  0   0  1  0  0  0...
...
```

Note that the first line of output from the vmstat program is invalid.

The column labeled “swap” shows the number of kilobytes (420056) of swap space available on your Analyzer host. To calculate the amount of memory available (in MB), divide 420056 by 1024, which yields approximately 410 MB of swap space.

Another way to determine the amount of logical memory available for swap space on your Analyzer host is to issue the **swap -s** command, as shown below:

```
$ swap -s
total: 117808k bytes allocated + 60160k reserved = \
 117968k used, 550152k available
```

Step 2 Use the MaxMB parameter in the NFADS.resources file to configure the DisplayServer to use the maximum amount of memory (MaxMB).

The MaxMB parameter in the NFADS.resources file limits the amount of memory to be used by the Analyzer to store data when processing a command.

See the "Customizing the NFADS.resources File" section on page 2-7 in Chapter 2 for instructions on changing the parameters in the NFADS.resources file.

Observe the following rules in calculating the largest reasonable starting value for the Analyzer's MaxMB parameter:

Rule 1—Ensure that the MaxMB value is less than or equal to the following:

Workstation's_Physical_Memory - 32

Rule 2—Ensure that the MaxMB value is less than or equal to the following:

SWAP_AVAILABLE - 100

Thus, the value of the MaxMB parameter should be no larger than either of the following:

(a) The host's actual physical RAM (256 MB in this case), minus 32 MB (see Rule 1)

(b) The host's available swap space (410 MB in this case), minus 100 MB (see Rule 2)

The formula for calculating physical memory is:

MaxMB—Less than or equal to 256 – 32 = 224 MB

The formula for calculating swap space is:

MaxMB—Less than or equal to 410 – 100 = 310 MB

Therefore, the value of the MaxMB parameter for the Analyzer should not be larger than 224 MB, which is the smaller of the two results calculated above.

In setting the MaxMB value, if you violate:

- Rule 1—Severe performance degradation can result, including disk thrashing.
- Rule 2—The system software can run out of swap space, in which case, active processes are killed.

For information about the expected performance of the Analyzer, see the ""Performance-Tuning Considerations" section on page 4-7."

Controlling Analyzer Modules

This section tells you how to individually start, stop, and check the status of the Analyzer modules.

To start all of the Analyzer modules simultaneously, see the "Starting the Analyzer" section on page 2-10 in Chapter 2.

To start the Analyzer modules individually, do so in the following order:

1. UtilityServer module
2. DisplayServer module
3. Display module

Running the UtilityServer Module

This section describes how you start, check the status of, and stop the UtilityServer.

Starting the UtilityServer Module

To start the UtilityServer, perform the following steps:

Step 1 Log in as root:

```
$ su root
password: <enter the password>
```

Step 2 Run the start.UtilityServer shell script, as indicated below:

```
# /opt/CSCOnfa/NFAUtility/bin/start.UtilityServer
```

Checking the Status of the UtilityServer Module

To check the status of an active UtilityServer process, run the check.UtilityServer shell script, as shown below:

```
$ /opt/CSCOnfa/NFAUtility/bin/check.UtilityServer
```

Stopping the UtilityServer Module

To stop an active UtilityServer process, perform the following steps:

Step 1 Log in as root:

```
$ su root
password: <enter the password>
```

Step 2 Run the stop.UtilityServer shell script, as shown below:

```
# /opt/CSCOnfa/NFAUtility/bin/stop.UtilityServer
```

Running the DisplayServer Module

This section describes how you start, check the status of, and stop the DisplayServer.

Starting the DisplayServer Module

To start the DisplayServer, perform the following steps:

Step 1 Log in as root:

```
$ su root
password: <enter the password>
```

Step 2 To start the DisplayServer, run the start.DisplayServer shell script, as shown below:

```
# /opt/CSCOnfa/NFAServer/bin/start.DisplayServer [server_logfile]
```

This command starts the DisplayServer and generates a log file of DisplayServer sessions. If you do not specify a <server_logfile>, the system uses the file name “server.out” by default.

If a server_logfile already exists, the log file output is stored in the lowest-numbered server_logfileNUM file (with “NUM” as a non-negative integer).

Caution The startup script, start.dsa, in the /opt/CSCOnfa/NFAServer/bin/ directory is used by Cisco engineering personnel for advanced troubleshooting. Do not run this script.

Checking the Status of the DisplayServer Module

To check the status of an active DisplayServer process, run the check.DisplayServer shell script, as shown below:

```
$ /opt/CSCOnfa/NFAServer/bin/check.DisplayServer
```

Stopping the DisplayServer Module

To stop an active DisplayServer process, perform the following steps:

Step 1 Log in as root:

```
$ su root
password: <enter the password>
```

Step 2 Run the stop.DisplayServer shell script, as show below:

```
# /opt/CSCOnfa/NFAServer/bin/stop.DisplayServer
```

Running the Display Module

This section describes how you start and stop the Display module.

Starting the Display Module

To start the Display, enter the following command:

```
$ /opt/CSCOnfa/NFADisplay/bin/start.Display
```

Stopping the Display Module

To exit from the Display, perform the following steps:

Step 1 To save any changes made to the data tree structure in the NetFlow Data area of the Display module window, invoke the Save option of the File menu.

Step 2 Select the Quit option of the File menu.

Performance-Tuning Considerations

At times, the Analyzer may run slowly due to the volume of data that it is processing.

In a busy network, traffic data can be amassed at a rate of hundreds of megabytes per hour. Searching or sorting through such a massive amount of data can be a huge and time-consuming task.

Table 4-1 shows the performance of the Analyzer during search operations on two different models of workstations. In each case, Analyzer performance is based on search operations on 100 MB of NetFlow data.

Table 4-1 Comparative Performance of Different Analyzer Workstations

Platform	Processing Speed	Search Time for 100 MB of Data
ULTRA-1 workstation	1 MB per second	96 seconds
SPARC-20 workstation	0.5 MB per second	206 seconds

Performance Tips

Consider the following in attempting to improve system performance:

- Minimize the amount of paging and context switching required in processing NetFlow data.

  If you need to change the value of the MaxMB parameter to improve Analyzer performance (see the“Optimizing Analyzer Memory Use” section on page 4-4), run the vmstat 60 program (refreshes output every 60 seconds) to monitor swap space and disk activity when the DisplayServer is processing large volumes of data for any of the “Detail” aggregation schemes.
- Run the perfmeter program to ensure maximum storage capacity for processed NetFlow data. Doing so reduces the likelihood of disk-thrashing.
- Store NetFlow data on a workstation that also serves as the DisplayServer host, as opposed to mounting a non-local file system from which desired NetFlow data is to be retrieved.