



AMX UNIX / Linux Administrator's Guide

Version 2.3

Copyright © 2019 by Naviga.
All Rights Reserved.

Naviga provides this documentation without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.

AMPS, AMPS Lan, AMPS Net, AMX, COGENT and ENQUE are trademarks of Naviga.

Other brands and product names used in this manual may be registered trademarks or trademarks of their respective holders.

Table of Contents

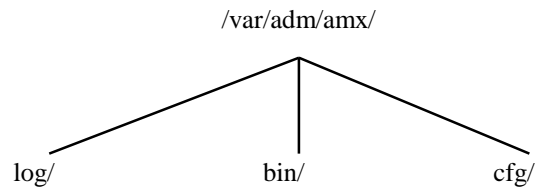
1	INTRODUCTION.....	2
1.1	DIRECTORY LAYOUT – ALL COMPONENTS	2
1.2	DIRECTORY LAYOUT – REQUIRED COMPONENTS.....	2
1.3	AMX UTILITIES / PROCESSES	4
1.4	DIRECTORY LAYOUT – OPTIONAL COMPONENTS	4
2	SYSTEM CONFIGURATION	5
2.1	AUTO BOOT CONFIGURATION - /ETC/RC.D/INIT.D/AMX.INIT (LINUX)	5
2.2	SYSTEM CONFIGURATION FILE - /NWSYS/RELEASE/CONF/.SYSCFG	5
2.3	TRANSPORT CONFIGURATION - /NWSYS/RELEASE/CONF/PORT.CONF	5
2.4	INTERCEIVER CONFIGURATION - /NWSYS/RELEASE/CONF/INTCVR.CONF.....	5
2.5	EMAIL CONFIGURATION - /NWSYS/RELEASE/CONF/NLOGD.CONF	5
2.6	FACILITIES - USER TOOLSET.....	5
3	APPLICATION STARTUP AND SHUTDOWN	6
4	APPLICATIONS MONITORING.....	7
4.1	PROCESS STATUS DISPLAY	7
4.2	PROCESS ALERT LOGGING	7
5	REAL-TIME ACTIVITY MONITORING.....	11
5.1	ALERTS OVERVIEW	11
5.1.1	Use of amxmonitor.....	11
5.1.2	Sample amxmonitor command.....	11
5.1.3	CRITICAL Conditions.....	12
5.1.4	WARNING Conditions	13
6	MAINTENANCE ACTIVITIES	15
6.1	DAILY ACTIVITIES	15
6.2	TROUBLESHOOTING PROCEDURES	15
6.3	SOFTWARE/PATCH INSTALLATION	15
7	ADVANCED SYSTEM CONFIGURATION	16
7.1	PROCESS CONFIGURATION	16
7.2	EMAIL TRIGGER CONFIGURATION.....	16
8	AMX MIGRATION.....	18
9	AMX REMOVAL	19
9.1	REQUIRED COMPONENTS.....	19
9.2	OPTIONAL COMPONENTS	19
9.3	FINAL CLEANUP – ALL COMPONENTS	20

1 Introduction

The AMX server software is controlled by various utilities and processes. All processes will automatically be started at boot time. Utilities are provided to serve as an interface to these processes. Tools used for checking process states, stopping and restarting, and report generation are provided.

1.1 Directory Layout – all components

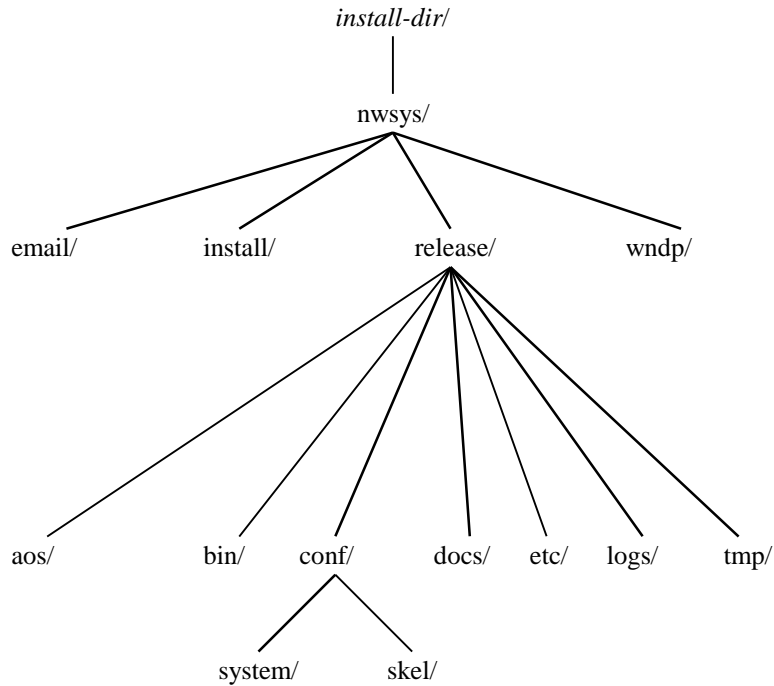
The directory `/var/adm/amx` is used to hold general configuration files that pertain to all components of the AMX software.



- `/var/adm/amx/` – This directory is created by the install utility. All subsequent installations
 - `log/` – This directory contains a record of all AMX installations that have been performed on this machine. This includes optional products as well as the required components.
 - `bin/` – This directory contains the start/stop/status utilities used for all installed optional products.
 - `cfg/` – This directory is used to hold information for each of the installed products (required components and optional). It contains version information as well as information pertaining to each product such as where the software was installed and under which user account it belongs to.

1.2 Directory Layout – required components

The directory structure for the files which comprise the required components of AMX is broken down into sub-directories below the root directory `/nwsys`. During the initial installation of the software, these directories will be created. The symbolic link `/nwsys` will be created to point to `install-dir/nwsys`. Note that `install-dir` can be located on any file partition that has sufficient disk space.



- *install-dir/* – This directory is chosen by the user at the time of initial AMX installation.
- */nwsys/email/* – This directory is used by the daemon process email as a staging area for outgoing messages. In the event a message cannot be sent, a sub-directory called *failure* will be used to hold the contents of the message.
- */nwsys/install/* – This directory is used exclusively by the install utility as a staging area used for the installation of a new software release or patch that has been automatically downloaded.
- */nwsys/wndp/* – This directory tree is populated by the daemon process wndp to hold all media content received by the AMPS Interceiver.
- */nwsys/release/* – This directory tree is used to hold the software release and appropriate documentation for the installed release.
 - *aos/* - This directory is used for **Add On Software**. The intent is a work area used for any system specific utilities that will NOT be included in a subsequent installation.
 - *bin/* - This directory holds all AMX utilities required for system administration, such as starting, stopping etc.
 - *conf/* - This directory holds all AMX configuration files used by processes in the *bin* and *etc* directories.
 - *conf/system* - This sub-directory holds state files created by the running daemons. File in *conf/system* should not be modified except by the running software.
 - *conf/skel* - This sub-directory holds example configuration files that get installed into the *conf* directory.
 - *docs/* - This directory contains all AMX documentation in Microsoft Word (.doc) and plain ASCII text (.txt) formats.
 - *etc/* - This directory holds AMX daemon processes.
 - *logs/* - This directory holds all log files generated by AMX.

- *tmp/* - This directory is used strictly as a run time scratch area where information may be transferred from one application to others via disk.

1.3 AMX Utilities / Processes

Utilities: (/nwsys/release/bin)

- *amxstart* - start AMX system
- *amxkill* - stop all AMX processes
- *amxstatus* - program for displaying the status of each AMX process. Use ***amxstatus -h*** to learn about optional switches.
- *amxmonitor* - alert log monitor program
- *windiag* - program used for configuring and diagnosing the connection to the AMPS server. See the *AMPS UNIX/Linux Diagnostic User Manual* for more information.

Daemons: (/nwsys/release/etc)

- *nlogd* - used for logging all alerts generated by all other applications
- *wndp* - caches the data (news) files created by the AMPS Interceiver and provides a generic TCP/IP interface whereby one or more external programs can receive and process the news using the WNDP protocol.
- *emaild* - used to provide outgoing email capabilities for the AMX server
- *intcvr* - maintains a connection between the client system and the AMPS server in order to receive news and other information.

1.4 Directory Layout – optional components

Refer to AMX UNIX/Linux Solutions Guide.

2 System Configuration

Unless you are directed by a Naviga engineer and the consequences of the changes have been explained in detail, *do not* change any of these parameters.

2.1 Auto boot configuration - /etc/rc.d/init.d/amx.init (Linux)

The amx.init startup/shutdown script was automatically installed during software installation. This file is required so that the AMX software will be automatically started during system boot-up and gracefully stopped during system shutdown. The file has the following startup and shutdown links:

```
/etc/rc.d/rc0.d/K90amx  
/etc/rc.d/rc5.d/S90amx
```

2.2 System configuration file - /nwsys/release/conf/.syscfg

The file .syscfg contains system configurations for your AMX server. These parameters are set when the system is configured during the initial installation. *Once these parameters are set, any changes may cause loss of data or unnecessary downtime.*

2.3 Transport configuration - /nwsys/release/conf/port.conf

The file port.conf contains the configuration data needed by the AMPS Interceiver to contact the AMPS server. These parameters are set when the system is configured during the initial installation.

2.4 Interceiver configuration - /nwsys/release/conf/intcvr.conf

The file intcvr.conf contains the configuration data needed for the AMPS Interceiver to successfully login and begin receiving data from the AMPS server. These parameters are set when the system is configured during the initial

2.5 email configuration - /nwsys/release/conf/nlogd.conf

The file nlogd.conf contains the configuration data needed for the AMX daemons to automatically generate email messages under certain conditions. In order to modify, create or delete existing triggers, please refer to the email configuration section (p. 16).

2.6 Facilities - User toolset

A standard toolset is provided as an interface to system operation. This includes starting and stopping the system, maintenance, status checks, etc. The toolset resides in /nwsys/release/bin. The path /nwsys/release/bin should be added to the PATH environment variable for the administration account so that the toolset can be referenced without the use of a full pathname.

3 Application Startup and Shutdown

Use `amxstart` to start all configured AMX processes. If processes are already running, these commands will have no effect on the system. Note: The file `/nwsys/release/conf/.feedcron` is submitted to the crontab to provide a permanent monitor for all processes. If any one of them crashes, it will be automatically be restarted.

```
{nadmin} amxstart
Logging process nlogd is currently not running
  Starting nlogd ... done
  Starting wndp ... done
  Starting emaild ... done
  Starting intcvr ... done
```

Use `amxkill` to stop the AMX server.

```
{nadmin} > amxkill
Are you sure you want to stop intcvr? (y/n) y
  Stopping intcvr ... done

  intcvr has been halted

Are you sure you want to stop wndp? (y/n) y
  Stopping wndp ... done

  wndp has been halted

Are you sure you want to stop emaild? (y/n) y
  Stopping emaild ... done

  emaild has been halted
```

4 Applications Monitoring

4.1 Process status display

The `amxstatus` command is used to display the current status of all configured AMX processes. The display output will indicate either *running* or *DOWN*. When the process is *running*, the output will also contain the process ID's (PIDs) for all that processes with the relevant process name.

```
# amxstatus

Release: AMX 2.0.0

      nlogd      running      4669

      wndp       running      4718
      emaild     running      4677

      intcvr     running      4692 4722
                          NZ:UP

**** All processes are up and running ****
```

4.2 Process alert logging

All AMX processes can and will log alert messages to the `/nwsys/release/logs/nlog` file. The messages written to the log are used to indicate the health, status and informational conditions of any and all processes executing on the server. One week of historical logs is preserved by renaming all existing logs at midnight. Each log file is renamed to indicate how many prior days the information in the log corresponds to. Thus, `nlog` becomes `nlog1`, `nlog1` becomes `nlog2`, etc. In the case of system problems or failures, the `nlog` file should always be checked immediately.

The `nlogd` daemon is the process responsible for creating, logging and maintaining the log files. The alert logging capabilities of `nlogd` include alert level filtering. Alert levels that get written to the `nlog` file are as follows:

- **CRITICAL** - This alert is meant to be used when a process has encountered an un-recoverable condition. The process cannot continue and must exit.
- **WARNING** - This alert is used to indicate that a condition exists that may require operator intervention or analysis. The condition was not catastrophic, thereby allowing the process to continue.
- **STATUS** - This alert is used to indicate informational type messages that require no action but may be useful for monitoring of system health and performance.
- **INFO** - This alert is used to log informational messages that do not require any action.
- **DEBUG** - This alert is intended for verbose logging of messages.

Using a tiered level of configurable logging is extremely useful and flexible for daily monitoring and system diagnosis. This is achieved by dynamically setting a global log level for nlogd to use. The log level is configurable to any one of the 5 categories defined above. All alerts that have equal or greater severity to the current level are logged to disk. For example, if the log level is set to STATUS, all STATUS, WARNING, and CRITICAL messages will be logged to disk and all INFO and DEBUG messages will be filtered out. STATUS is the typical level a production server will run in. It is important to note that setting the log level to INFO or DEBUG may cause degradation in system performance due to an increased level of disk I/O. Disk space should also be carefully monitored when running a server at INFO or DEBUG level for extended periods of time.

The dynamic logging level is set by using the *amxloglvl* utility. In the event that nlogd cannot keep up with all incoming alerts, all WARNING and CRITICAL level messages will automatically be written to the console. *This is an extremely rare condition, not likely to occur.*

Here is an example of a typical alert message from the nlog file:

```
01:38:45 STATUS:    intcvr: Interceiver has been authorized (ID: 0x5f)
```

Use *amxloglvl* to dynamically set the log level for one or more specified running processes.

amxloglvl -h

```
amxloglvl [warning|status|info|debug]
          [-a|-r|-s|-m|-p|-n|-o|-c|-v|-x]

level          - when just level is specified, all processes
                  that have not been explicitly set will be
                  assigned this new level
-a level       - all processes will be set to this level
                  regardless of any explicit setting
-r -[p|n|o]    - reset specified process to global level.
  [pid|name]    this will clear any prior setting.
                  specify pid, log name or process name
-s             - show current process table
-m             - show process table for process that have been set
-p pid lvl     - set plog level for pid
-n name lvl    - set plog level for process 'log' name
-o name lvl    - set plog level for process name of all
                  associated processes. For example, using
                  s_disk would set sdisk10, sdisk20... etc.
-c             - clear all slots in process table that are
                  currently used by processes that are no
                  longer running
-v             - verbose output where applicable
-x slot        - display raw data for this slot
```

```
Current Level: STATUS
```

The following examples illustrate some of the *amxloglvl* capabilities:

1. Display the current process table entries:

```
# amxloglvl -s
```

Slot	Process	Log Name	PID	Level
****	*****	*****	***	*****
1	nlogd	nlogd	16683	STATUS
2	wndp	wndp	16691	STATUS
3	emaild	emaild	16698	STATUS
4	intcvr	intcvr	16699	STATUS

2. Set the wndp process to debug using process name:

```
# amxloglvl -n wndp debug
```

```
Adjusting log level for wndp
Log level --> DEBUG
```

3. Display the current process table entries: (*Asterisk indicates process has been set to desired level*)

```
# amxloglvl -s
```

Slot	Process	Log Name	PID	Level
****	*****	*****	***	*****
1	nlogd	nlogd	16683	STATUS
2	wndp	wndp	16691	DEBUG*
3	emaild	emaild	16698	STATUS
4	intcvr	intcvr	16699	STATUS

4. Set the intcvr process to INFO using process PID:

```
# amxloglvl -p 16699 info
```

```
Adjusting log level for intcvr
Log level --> INFO
```

5. Display the current process table entries: (*Asterisk indicates process has been set to desired level*)

```
# amxloglvl -s
```

Slot	Process	Log Name	PID	Level
****	*****	*****	***	*****
1	nlogd	nlogd	16683	STATUS
2	wndp	wndp	16691	DEBUG*
3	emaild	emaild	16698	STATUS
4	intcvr	intcvr	16699	INFO*

6. Set all processes to new level: (*Note, this does not affect any prior processes that were set by name or pid*)

```
# amxloglvl info
```

```
Log level --> INFO
```

7. Display the current process table entries: (*Asterisk indicates process has been set to desired level*)

```
# amxloglvl -s
```

Slot	Process	Log Name	PID	Level
****	*****	*****	***	*****
1	nlogd	nlogd	16683	INFO
2	wndp	wndp	16691	DEBUG*
3	emaild	emaild	16698	INFO
4	intcvr	intcvr	16699	INFO*

8. Reset all processes to status level: (*Note, this clears any prior individual process settings*)

```
# amxloglvl -a status
```

```
Log level --> STATUS
```

9. Display the current process table entries: (*Asterisk indicates process has been set to desired level*)

```
# amxloglvl -s
```

Slot	Process	Log Name	PID	Level
****	*****	*****	***	*****
1	nlogd	nlogd	16683	STATUS
2	wndp	wndp	16691	STATUS
3	emaild	emaild	16698	STATUS
4	intcvr	intcvr	16699	STATUS

5 Real-time activity monitoring

5.1 Alerts overview

This section will address a subset of CRITICAL and WARNING alerts that happen most frequently. Please refer to the appropriate section (4.1.3 or 4.1.4) based on alert type. These sections will specify the alert, a short description of what is happening and the actions necessary to correct the problem.

All CRITICAL alerts are used to indicate that a process cannot continue and must exit. If a CRITICAL alert occurs that is not documented in this section, it is important to determine why this has happened. If the system cannot be restarted, the problem should be escalated to the next level of support. WARNING messages indicate a non-fatal condition or problem. WARNING messages not documented in this section most likely do NOT require support escalation. If the alert appears to be serious based on the text within the alert, escalate to the next level of support.

5.1.1 Use of *amxmonitor*

The *amxmonitor* command is used to monitor all incoming alerts as they are being generated. The utility will prompt for the different level of alerts, (CRITICAL, WARNING and STATUS), and filter appropriately. *amxmonitor* provides a filter to the /nwsys/release/logs/nlog disk file which contains all alerts, based on user input. It is important to note that the nlog file is a plain ASCII file, therefore it can be viewed with standard utilities such as *more*, *vi*, *grep*, etc.

5.1.2 Sample *amxmonitor* command

```
# amxmonitor

*****
*
*   Starting amxmonitor: Wed May  4 15:13:27 EDT 2000
*
*   Enter ^C at any time to abort amxmonitor
*
*****

Monitor CRITICAL alerts? (y/n) y <Enter>
Monitor WARNING alerts? (y/n) y <Enter>
Monitor STATUS alerts? (y/n) n <Enter>

13:12:45 WARNING: : intcvr: Successfully opened new socket connection
^C
#
```

5.1.3 CRITICAL Conditions

In response to any critical condition you must restart the system in accordance with the restart procedures documented in this manual.

hh:mm:ss CRITICAL: intcvr: Failed to open /nwsys/release/conf/intcvr.conf (error)

Meaning: The AMPS configuration file could not be opened.

Action: Check that file permissions match that of the user account under which the system is running. To reset all permissions automatically, run as root
`/nwsys/release/bin/rc.news -u account` where account is the user account to be used.

hh:mm:ss CRITICAL: intcvr: Authorization denied

Meaning: AMPS server denied login access.

Action: Assuming that intcvr was able to successfully connect in the past, check that `/nwsys/release/conf/intcvr.conf` has not been corrupted. Verify file against last known good backup copy. If intcvr.conf is OK, run the Interceiver connection test (documented in the *AMPS UNIX/Linux Diagnostic User Manual*) to determine at what point the connection process fails.

hh:mm:ss CRITICAL: intcvr: Memory failure (system error)

Meaning: Failed to allocate enough memory for normal data processing.

Action: Verify all hardware requirements as specified in the installation notes. If hardware requirements are acceptable, check for system memory failure.

hh:mm:ss CRITICAL: intcvr: Cannot open socket for reading (system error).

Meaning: Failed to establish TCP connection back to AMPS server.

Action: Verify internet connectivity and all hardware required for the AMX service. Run the Interceiver connection test (documented in the *AMPS UNIX/Linux Diagnostic User Manual*) to determine at what point the connection process fails.

5.1.4 WARNING Conditions

hh:mm:ss WARNING: intcvr: Failed to save stats to /nwsys/conf/system/intcvr.state (system error)

Meaning: Unable to write state information to disk.

Action: Check for proper permissions and available disk space.

hh:mm:ss WARNING: intcvr: Ping timeout on seqno #####
hh:mm:ss WARNING: intcvr: cvidrSocket::CheckConn - Ping Timeout, closing connection (AMPS_MSG)

Meaning: AMPS server failed to respond to ping request within allocated time interval.

Action: None required. The Interceiver will automatically close its current server connection and establish a new connection to the AMPS server.

hh:mm:ss WARNING: intcvr: Received unknown control packet '#####' (AMPS_MSG)
hh:mm:ss WARNING: intcvr: Unknown session mgt flag ##### (AMPS_MSG)
hh:mm:ss WARNING: intcvr: Unknown presentation layer on TVC ##
hh:mm:ss WARNING: intcvr: Unknown download HIC ## (AMPS_MSG)
hh:mm:ss WARNING: intcvr: Invalid download file name HIC received.
hh:mm:ss WARNING: intcvr: Missing data blks (# to #) in file download.
hh:mm:ss WARNING: intcvr: Got download data block without download file name.

Meaning: The AMX server was unable to process incoming data content.

Action: It is extremely unlikely that any of these messages will occur. If they do, contact customer support.

hh:mm:ss WARNING: Unable to open...
hh:mm:ss WARNING: Failed fwrite (system error)
hh:mm:ss WARNING: Failed renaming...(system error)

Meaning: Failed basic disk operation.

Action: Check for proper permissions and available disk space.

hh:mm:ss WARNING: Retranmission ##### (current seqnum #####)
hh:mm:ss WARNING: VerifySeqNum: Missed msgs ##### - #####

Meaning: Retransmission indicates reception of duplicate content from head-end.
VerifySeqNum indicates data content gap.

Action: It is extremely unlikely that any of these messages will occur. If they do, contact customer support.

6 Maintenance Activities

6.1 Daily Activities

It is imperative that all processes remain up and running. Use of *amxstatus* will verify that processes are running. Use *amxmonitor* to view alerts as they are being generated. It is important to scan entire alert log (by using **grep**) for CRITICAL alerts along with any other conditions you may wish to check. Depending on the size of the log file, it may be best to scan the entire file to ensure that all processes are not only running, but that they have not encountered any situations that would cause system problems. Use *winddiag* to monitor the integrity of both the AMPS connection and incoming flow of data on a continuous basis. For example, use *winddiag* to display this status once every ten seconds.

Backup the following directories (which hold software and configuration files) so that in the event of a system crash, recovery is possible.

- /var/adm/amx
- /nwsys/release

Optionally, backup the following directories (which hold data received by the Interceiver). Note that this does not account for the directories used by the applications described in the AMX UNIX/Linux Solutions Guide.

- /nwsys/wndp

6.2 Troubleshooting Procedures

- run *amxstatus* to determine if any processes have been terminated.
- run *amxmonitor* to view current alert log.
- check /nwsys/release/logs/nlog for messages that may describe the problem. Check any WARNING or CRITICAL messages against *Real-time activity monitoring* section of this manual for meaning and action that needs to be taken.
- use *amxkill* and *amxstart* to cycle server processing if necessary.
- use the procedures described in the *AMPS UNIX/Linux Diagnostic User Manual* to examine the AMPS connection and data flow into the AMX application.
- check the file /var/adm/messages for system messages that have been logged to the system console that may have been overlooked.

6.3 Software/Patch installation

Please refer to Release Notes that will accompany all new releases and patches.

7 Advanced System Configuration

7.1 Process Configuration

```
#####
# .syscfg
#
# This is the system configuration file used for process configuration.
# All lines starting with '#' are considered comments and are ignored.
# All administration utilities will run based on the information
# contained within this file.  If any required information is
# missing, the utility will display an appropriate error.
# Use the following lines as templates for creating your .syscfg.
# Don't forget when using this skeleton file to rename it to
# .syscfg from syscfg
#
# NOTE: bourne shell scripts defined to run as Process: must end
#       with a '.sh' extension to be handled properly.
#
#####

Operating_system: Linux
Backup_device: /dev/rmt/0m

Logging_process: nlogd

Feed_process: wndp
Feed_process: emaild
Feed_handler: intcwr -mxp -lp

SERVER_TYPE: AMX
PRODUCTION_USER: nadmin
```

7.2 Email trigger configuration

The UNIX/Linux implementation of AMX can generate email messages to designated addresses based on the occurrence of a specific event. For example, let's take the most basic case, which has been automatically setup during the initial system installation; if a process encounters a fatal condition and exits unexpectedly, an email message will automatically be sent indicating that a problem has occurred.

When properly configured, triggers can be setup such that nlogd will monitor the contents of each incoming alert and perform a specified event when the trigger condition is satisfied. The trigger event is achieved by forking a child process to do the actual work. Allowable events include sending automatic email, automatic paging and running any number of operating system commands to include in the outgoing email. Useful commands such as *amxstatus* or *df -k* can have their output appended to the email.

There are system-imposed limitations however. There can only be three simultaneous nlogd children at a time. If there is a flurry of WARNING messages, only the first three are guaranteed to get processed. As the nlogd children complete their tasks, new incoming WARNING messages would get processed in the same manner. There is also a configurable limit of events per trigger per day. The counts for each defined trigger will be reset at midnight.

It is also important to note that when nlogd is configured for trigger events, it is assumed that the network on which the server is installed, and running is configured for email transmission. Paging is achieved in conjunction with an email/paging service. This is a special email account used through a paging service that provides an account that will trigger an outgoing page when email is received.

In addition to writing to the nlog file on disk, nlogd can also be configured to write to a serial printer. Proper hardware configuration must be in place before this can happen. When properly configured, all CRITICAL and WARNING messages will be automatically sent to the printer in addition to be logged on disk.

8 AMX Migration

It is important to note that if you are moving AMX to a different server you cannot just remove and reinstall the software. Configuration information contained on the first server must be copied to the second in order to establish a connection back to the AMPS head-end. It is also important to note that any historical data contained on disk should be copied.

All steps are to be executed by the root user account.

1. Determine what has been installed on this server.

```
# /var/adm/amx/bin/amxopt info
```

```
Current installation information Thu May 25 12:43:24 EDT 2003
```

```
Application : AMX
Release    : 1.1
User acct  : nadmin
Location   : /pool/amx

Application : File Archiver
Release    : 1.1
User acct  : nadmin
Location   : /pool/amx

Application : AmxViewer
Release    : 1.1
User acct  : nadmin
Location   : /pool/amx
```

2. Backup each unique directory specified under the **Location** from the output of amxopt. In the above example, copying /pool/amx to the new server would catch all software and data. The target directory on the remove server **MUST** match exactly. The permissions, owner and group should **MUST** also match exactly.
3. Backup the AMX registry /var/adm/amx and copy to new server.
4. If AMX (required component) is installed, you will need to execute the rc.news command in order to have all required processes automatically start at boot time.

```
# /pool/amx/nwsys/release/bin/rc.news install
```

```
Verifying installation state...
Installing /etc/rc.d/init.d/amx.init
Shutdown link /etc/rc.d/rc0.d/K90amx is installed
Startup link /etc/rc.d/rc5.d/S90amx is installed
Installation state complete
```

9 AMX Removal

This section should be used when it is desired to remove AMX and all related components from your server. It is assumed that the proper migration steps from the prior section were taken in order to maintain a fully functional AMX installation. If a migration to another server is planned and has not yet been completed - **STOP. DO NOT CONTINUE UNTIL MIGRATION IS COMPLETE AND FULLY OPERATIONAL.**

All steps are to be executed by the root user account.

9.1 Required components

1. Remove the auto-start mechanism for AMX.

```
# /etc/init.d/rc.news uninstall
```

```
Removing /etc/rc.d/rc0.d/K90amx
```

```
Removing /etc/rc.d/rc5.d/S90amx
```

```
Removing /etc/rc.d/init.d/amx.init
```

```
All automatic startup and shutdown processing has been removed
```

2. Remove all software and related data. (*this may take a while based on the amount of historical data*)

```
# rm -rf /nwsys
```

3. Clean out appropriate registry entries.

```
# rm /var/adm/amx/cfg/amx.*
```

9.2 Optional components

1. Determine all optional components that have been installed on server.

```
# /var/adm/amx/bin/amxopt info
```

```
Current installation information Thu May 25 12:43:24 EDT 2000
```

```
Application : AMX
Release    : 1.1
User acct: nadmin
Location   : /pool/amx
```

```
Application : File Archiver
Release    : 1.1
User acct: nadmin
Location   : /pool/amx
```

```
Application : AmxViewer
Release    : 1.1
User acct: nadmin
Location   : /pool/amx
```

2. Remove all software and related data from each designated Location from the above information. (*this may take a while based on the amount of historical data*). The number of

locations may vary based on which components have been installed. In this example, all components were installed in /pool/amx.

```
# rm -rf /pool/amx
```

9.3 Final cleanup – all components

1. Remove all registry information. **(NOTE: This step should only be performed after successfully completing the prior two sections.)**

```
# rm -rf /var/adm/amx
```