



---

**PAM Distribution Utility (PAMDU)**  
for  
**Linux, Solaris, HP-UX (PA-RISC),  
Windows Server, and z/OS Systems**

**Release 1.0**

---

**User Guide**

# Revision Record

<b>Revision</b>	<b>Description</b>
01 (01/2011)	Initial release.

© 2011 by Network Executive Software. Reproduction is prohibited without prior permission of Network Executive Software. Printed in the U.S.A. All rights reserved.

The U.S. Department of Commerce restricts the distribution of technical information contained in this document when exported outside the U.S. Therefore, careful attention should be given to compliance with all applicable U.S. Export Laws if any part of this document is to be exported.

You may submit written comments using the comment sheet at the back of this manual to:

Network Executive Software, Inc.  
Publications Department  
6420 Sycamore Lane, Suite 300  
Maple Grove, MN 55369  
USA

Comments may also be submitted over the Internet by addressing e-mail to:

[pubs@netex.com](mailto:pubs@netex.com)

or, by visiting our web site at:

<http://www.netex.com>

Always include the complete title of the document with your comments.

# Preface

This manual describes the PAM Distribution Utility (PAMDU) software for NetEx systems. PAMDU is used in conjunction with TCP/IP allowing the user to easily distribute NetEx PAM files across the network.

This manual is intended for all users of PAMDU, and contains all of the information necessary to expand the user's ability to the fullest extent of the software.

The manual is divided into four parts plus two appendices:

“PAMDU Overview” gives a basic description of the PAMDU utility.

“*pamdu* (Hub Component)” explains the operation of the hub component.

“*pamlstn* (Spoke component)” explains the operation of the spoke component(s).

“Installation Procedures” explains the installation steps by operation system.

“Appendix A. *pamdu* Error Messages” lists the error messages generated by *pamdu* (hub).

“Appendix B. *pamlstn* Error Messages” lists the error messages generated by *pamlstn* (spokes).



# Notice to the Reader

This product is intended for use only as described in this document. Network Executive Software cannot be held responsible for use of features or options in any undocumented manner. This manual is subject to change without notice.

This document and the programs described in it are furnished under a license from Network Executive Software and may be used, copied, and disclosed only in accordance with such license.

This document contains references to the trademarks of the following corporations.

## Corporation Trademarks and Products

<b>Network Executive Software</b>	<b>NetEx, NetEx/IP</b>
<b>The Open Group</b>	<b>UNIX</b>
<b>International Business Machines</b>	<b>IBM</b>
<b>Oracle</b>	<b>SUN, Solaris</b>
<b>The University of California, Berkeley</b>	<b>Berkeley 4.x BSD</b>

These references are made for informational purposes only.

The diagnostic tools and programs described in this manual are **not** part of the products described.

# Document Conventions

The following notational conventions are used in this document.

Format	Description
displayed information	Information displayed on a CRT (or printed) is shown in <i>this font</i> .
user entry	<i>This font</i> is used to indicate the information to be entered by the user.
UPPERCASE	The exact form of a keyword that is not case-sensitive or is issued in uppercase.
MIXedcase	The exact form of a keyword that is not case-sensitive or is issued in uppercase, with the minimum spelling shown in uppercase.
<b>bold</b>	The exact form of a keyword that is case-sensitive and all or part of it must be issued in lowercase.
lowercase	A user-supplied name or string.
value	Underlined parameters or options are defaults.
<label>	The label of a key appearing on a keyboard. If “label” is in uppercase, it matches the label on the key (for example: <ENTER>). If “label” is in lowercase, it describes the label on the key (for example: <up-arrow>).
<key1><key2>	Two keys to be pressed simultaneously.
No delimiter	Required keyword/parameter.

# Contents

<b>Revision Record .....</b>	<b>ii</b>
<b>Preface.....</b>	<b>iii</b>
<b>Notice to the Reader.....</b>	<b>v</b>
Corporation Trademarks and Products.....	v
Document Conventions.....	vi
<b>Contents .....</b>	<b>vii</b>
Figures.....	viii
<b>PAMDU Overview .....</b>	<b>1</b>
Introduction.....	1
Definitions.....	1
System Requirements.....	1
<b><i>pamdu</i> (Hub Component) .....</b>	<b>3</b>
NCT Changes.....	3
<i>pamdu</i> (HUB) Configuration File Parameters .....	4
<i>pamdu</i> (Hub) Command Line Options.....	6
Program Termination .....	7
<i>pamdu</i> (Hub) Other Files .....	7
NCT File .....	7
NCT_DIRECTORY.....	8
Checkpoint File.....	8
Configuration Manager Output.....	8
Lock File.....	8
TCP Connections .....	8
<i>pamdu</i> (Hub) Security Considerations .....	8
<i>pamdu</i> Sample Scenarios .....	9
Send everyone a network configuration .....	9
Testing a new network configuration .....	9
Recovering a computer system .....	9
Determine which spokes have not received the current network configuration .....	9
An error occurred when <i>pamdu</i> was processing and the lock file remains.....	9
<i>pamdu</i> Debugging.....	9
Additional Messages.....	9
CM Output Files .....	10
<b><i>pamlstn</i> (Spoke component) .....</b>	<b>11</b>
<i>pamlstn</i> (Spoke) Configuration Parameters .....	11
<i>pamlstn</i> - Configuration File.....	11
Configuration Parameters for <i>pamlstn</i> .....	11
<i>pamlstn</i> - Configuration File User Exits .....	12
Configuration Defaults for <i>pamlstn</i> by Operating System.....	13
<i>pamlstn</i> (Spoke) Command line Options .....	14

<i>pamlstn</i> (Spoke) Other Files .....	14
NetEx Default file .....	14
PAM_OLD .....	14
<i>hostname</i> .PAMDUKM.RLOG .....	15
<i>hostname</i> .PAMDUKM.RLOGE.....	15
<i>hostname</i> .CMINPUT.....	15
<i>pamlstn</i> (Spoke) Security Considerations.....	15
<b>Installation Procedures .....</b>	<b>17</b>
SuSE Linux.....	17
HUB Component.....	17
Spoke Component .....	17
Solaris .....	17
Spoke Component .....	17
HP-UX (PA-RISC).....	18
Spoke Component .....	18
z/OS .....	18
Spoke Component .....	18
Windows Server .....	19
Spoke Component .....	19
<b>Appendix A. <i>pamdu</i> Error Messages .....</b>	<b>21</b>
<b>Appendix B. <i>pamlstn</i> Error Messages.....</b>	<b>25</b>

## Figures

Figure 1. Visual overview of PAMDU implementation.....	2
--	---



# PAMDU Overview

## Introduction

The PAM Distribution Utility (PAMDU) is a utility which runs in tandem with NetEx on several platforms. The purpose of this utility is to insure each NetEx host has a consistent view of the NetEx network configuration (specifically the Physical Address Maps or PAMs). This allows the customer to centrally manage the NCT and distribute changes throughout the NetEx configuration in a timely fashion, insuring each NetEx host has a consistent view and can communicate with the other NetEx systems.

This is a client/server implementation, where the clients listen on a well-known port and the server connects to them. However, to minimize confusion it is described using the terms hub and spokes. The hub controls the master copy of the NCT. This is where all NCT changes are made. The spokes listen on a well-known (configurable) TCP port. When changes are made to the NCT, the hub (*pamdu*) attempts a TCP connection with each spoke (*pamlstn*). The master copy of the NCT is transmitted across the link (performing ASCII/EDCDIC translation if necessary). The spoke, (*pamlstn*) then backs up the current PAM, and runs the Configuration Manager to generate the new PAM. If the file is successfully generated, a notification is sent back to the hub acknowledging a successful transfer. If the PAM is not generated, an error status is returned to the hub, and the previous PAM file is restored. The hub records which spoke systems have successfully generated the new PAM. The hub does not know if the spoke system has activated the new network configuration.

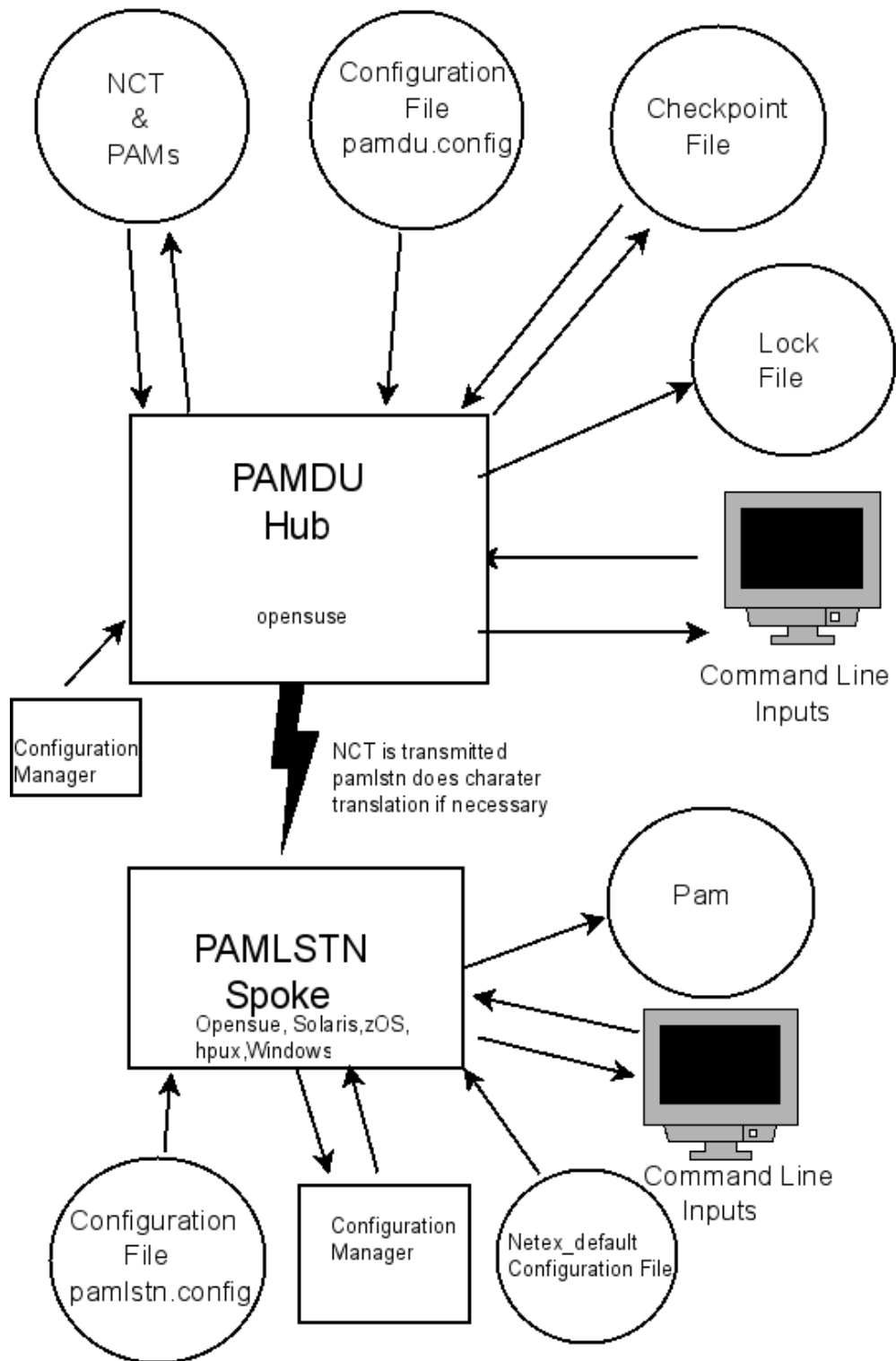
## Definitions

PAMDU	PAM Distribution Utility for tracking, generating and distributing PAM tables to the other host (spoke) systems in the NetEx configuration
<i>pamdu</i>	The hub component (server) of the PAMDU system.
<i>pamlstn</i>	The spoke component (client) of the PAMDU system. This will run on all NetEx systems including the hub.
NetEx™	NETWORK EXECUTIVE software; a proprietary transport protocol.
NCT	The Network Configuration Table for the NetEx system. This is used by the configuration manager to generate a PAM for each NetEx system.
PAM	A Physical Address Map (PAM) file used by the NetEx software to locate other hosts in the NetEx system.

## System Requirements

The PAMDU Hub component (*pamdu*) runs on Linux (specifically SuSE Linux).

The PAMDU spoke component (*pamlstn*) runs on SuSE Linux, z/OS, Windows Server, HP-UX (PA-RISC), and Solaris operating systems which are also running NetEx (*pamlstn* can simultaneously run on the hub).



**Figure 1. Visual overview of PAMDU implementation.**

# *pamdu* (Hub Component)

## NCT Changes

Depending on your host names and network configuration, the PAMDU utility may need additional information added to the NCT. In order to minimize the impact on the existing software and keep the NCT compatible with older software version, this information should be added as comments in the current NCT. Comments in NetEx are defined as: “**any line where the first character is one of the following characters “# ; \* ! “, followed by a white space. If these characters are located on a statement, preceded by at least one white space character, the rest of the line will be regarded as a comment.**”

(NOTE: Older configuration Managers may not interpret the # sign as a comment character).

*pamdu* introduces two new parameters to the HOST statement. These parameters must be added as comments if they are needed. They may be on the end of the host statement, by adding a # IP= on the statement, or the next line after the host statement, again by adding a # IP= and/or IP-PORT= . If these are not comments, Configuration Manager will report them as an unknown parameter and fail to generate the PAM file.

The first parameter that may be required is the IP=. This parameter tells the *pamdu* software how to contact this host. This parameter may be an IP address in xxx.xxx.xxx.xxx notation (dotted-decimal), or a host name that is resolvable through DNS services. **This parameter defaults to the NetEx host name.** If the NetEx host name is YELLOW, and YELLOW is the host name in your network, a change will not be required. If the NetEx host YELLOW exists on the host YELLOWSTONE, then the IP=YELLOWSTONE (in our example) must be added as a comment card behind the host statement for YELLOW in the NCT.

The second parameter that maybe required is IPPORT=. This specifies the TCP port number that the host will use to communicate with this specific spoke. This parameter is required only if this spoke requires a different port number than the global port number used by *pamdu*.

## ***pamdu* (HUB) Configuration File Parameters**

This configuration file (`pamdu.config`) contains fixed name tags and values which allow the site to customize the operation of the PAMDU Hub. It will be located in the directory where the PAMDU utility was installed (`/opt/netex/conf`). This file must exist. It may be empty if you wish to use the defaults. This is a text file of character strings in the local host character set and edited manually. Lines with a `#` in the first column will be interpreted as comments and be ignored. *pamdu* currently installs on an SuSE Linux operating system. The defaults for the system are:

SuSE Linux	<p>NCT_DIRECTORY /opt/netex</p> <p>PAMDU_PORT 6950</p> <p>VERBOSE 2</p> <p>NOTIFY_INTERVAL 5</p> <p>PAM_RETRIES 3</p> <p>STOP_ON_ERROR YES</p> <p>MAX_SESSIONS 6</p> <p>CHECKPOINT /opt/netex/conf</p> <p>IGNORE_HOST defaults to all hosts are processed</p>
------------	---

<b>Fixed Name Tag</b>	<b>Value</b>	<b>Description</b>
NCT_DIRECTORY	<i>DirectoryName</i>	The directory name where the NCT is stored; the format of each subdirectory is CURRENT_VERSION/NetExHostname/pamfile. Due to system differences these PAM files may not be compatible with the system they were generated for. These are used only as a validity check.
NOTIFY_INTERVAL	<i>TimeInMinutes</i>	The interval at which the PAMDU Hub attempts to reconnect to the PAMDU Spokes after a connection failure or no response.  <i>TimeInMinutes</i> – decimal value in minutes from 0 to 65535. 0 means to retry until all PAMs are acknowledged by the spoke systems.

<b>Fixed Name Tag</b>	<b>Value</b>	<b>Description</b>
PAM_RETRIES	<i>Number of tries</i>	The number of times the PAMDU Hub will attempt to contact the host at NOTIFY_INTERVAL before PAMDU Hub will give up and terminate. Any hosts not contacted will be left in pending status in the checkpoint file. The range is 0 through 99. A PAM_RETRIES of 0 means to run until all hosts are in receive status.
VERBOSE	<i>Level</i>	Verbosity level of activity which is output to SYSLOG. Level is an ASCII number where: <b>0</b> – no logging <b>1</b> – catastrophic errors logged only <b>2</b> – failures logged <b>3</b> – all activity logged <b>4</b> – not used <b>5</b> – Extensive diagnostic logging
PAMDU_PORT	<i>Portnumber</i>	This is the well-known default TCP port number used to connect to the spokes. This is the port number that the spoke will be contacted on.
CHECKPOINT	<i>FileName</i>	The name and location of the checkpoint file.
STOP_ON_ERROR	YES/NO	This specifies the action to be taken when an error is encountered while generating the PAM files. <b>YES:</b> <i>pamdu</i> will stop generating PAMs and terminate. Contact with the spoke systems will not be made. <b>NO:</b> <i>pamdu</i> will continue generating PAMs and distribute NCTs to all spoke systems for PAM generation.
IGNORE_HOST	<i>hostnames</i>	The hostnames specified will not be contacted by <i>pamdu</i> . The PAM will be generated, it will not be distributed. Hostnames may be comma separated. Multiple IGNORE_HOST lines maybe specified.
MAX_SESSIONS	<i>Decimal number</i>	This is the maximum TCP connections that will be started when the NCTs are being distributed for PAM generation. Range is 1 through 10.

## ***pamdu* (Hub) Command Line Options**

The hub component of the PAM Distribution Utility is *pamdu*. This utility will write a lock-file to prevent more than one copy from running a transfer command concurrently. A user may run a check command or an exception command while *pamdu* is processing. All messages will be written to stdout and the system log.

The command line interface supports the following commands:

<b>Command</b>	<b>Description</b>
<code>pamdu -n filename [host host ...]</code>	<p>New NCT File</p> <p>The NCT specified by the filename will be retrieved using the NCT_DIRECTORY parameter from the configuration file. If the version number is the same as the last PAM distributed, an error message will be issued and the program will terminate, otherwise all PAMS will be generated. If <i>hostnames</i> are coded, only those systems will receive the appropriate updated NCT, other systems will be scheduled to receive it otherwise all systems will receive the NCT for PAM generation, unless the host was specifically excluded by the configuration file. Each host name will be separated by a space.</p>
<code>pamdu -f filename [host host ...]</code>	<p>Force an existing NCT to be retransmitted</p> <p>This is the same as “<code>pamdu -n</code>” except the version number check will not be made. If <i>hostnames</i> are specified, only they will receive the NCT to regenerate PAM. All hosts will be scheduled to receive the file on the next “<code>pamdu -r</code>” command.</p>
<code>pamdu -c [host host ...]</code>	<p>Check Status</p> <p>This command will display the current checkpoint file. This shows the current status of the spoke systems. It does not show if the user has activated the PAM. If you specify a host, only those hosts will be displayed.</p>
<code>pamdu -e</code>	<p>Display Exception Status</p> <p>This command displays the checkpoint file for all hosts not in receive status.</p>
<code>pamdu -s host [host host ...]</code>	<p>Send Command</p> <p>This will send the current (last NCT supplied) to each host specified, unless it is excluded by the configuration file. It will not affect the status of other systems.</p>

Command	Description
<code>pamdu -r</code>	<p>Restart Command</p> <p><i>pamdu</i> will read the checkpoint file and attempt to connect with each spoke that has not acknowledged the successful generation of the PAM file. If one or more hostnames were specified on the last “<code>pamdu -n</code>” or “<code>pamdu -f</code>” command, the NCT will be sent to the rest of the spoke systems for PAM file generation.</p>
<code>pamdu -u</code>	<p>Unlock Command</p> <p>Only one instance of <i>pamdu</i> may be attempting data transmission at a time. A lock file is created to record that <i>pamdu</i> is transmitting data. <i>pamdu</i> may run check commands while a second copy is transmitting data.</p> <p>This command will remove the current lock file and allow the <i>pamdu</i> program to run. The user <b>MUST</b> verify that another copy is not running.</p>
<code>pamdu -d -cmd parameters</code>	<p>The <code>-d</code> option (debug) must follow the <i>pamdu</i> program name, and before the command option you wish to execute. This changes the default VERBOSE level to 5 at program start time for this execution only. This allows for debugging messages that occur before the config file is processed.</p>

## Program Termination

The *pamdu* program will terminate, when the retry count has been exceeded, or when all the hosts scheduled to receive the updated information have received it. If the user wishes to terminate the program prior to those conditions, the user may enter a Cntl-C in the window running *pamdu*, or enter a `kill -1 pid`, `kill -2 pid`, or `kill -3 pid`. *pamdu* will trap these signals and terminate. The current checkpoint file will be written, so the program can continue where it left off. Terminating the program in other ways, may cause some updates to the checkpoint file to be lost, and the lock to remain, preventing *pamdu* from being restarted, until the “`./pamdu -u`” command is entered.

## *pamdu* (Hub) Other Files

### NCT File

This is a read-only file. *pamdu* extracts the NCT version number, any `IP=` statements that were added as comment cards to the host statements to obtain the IP address, and any `IPPORT=` statements in case a specific spoke could not use the system-wide default port. *pamdu* invokes the configuration manager to validate PAM files can be generated for each spoke. If the PAM file for one spoke fails, *pamdu* will not contact any spokes if the configuration parameter `STOP_ON_ERROR` equals YES. If `STOP_ON_ERROR` equals NO, *pamdu* will contact each spoke with the new NCT.

## NCT\_DIRECTORY

*pamdu* must have read/write/create access to this directory. *pamdu* will create a subdirectory called “CURRENT\_VERSION”. In this subdirectory will be a directory for each NetEx host. In the host directory, will be the PAM file that was generated for that host. These PAM files may not work if they are copied to the target system. These files are used for validation only.

## Checkpoint File

The checkpoint file is created as a result of a “*pamdu -n filename*” being entered. It is a permanent file. For each host system, an entry will be created. This entry will contain:

- The name of the NCT to be distributed.
- The version number of the NCT to be distributed.
- The IP address of the system, if it was specified.
- The current status (pending, received, excluded ...) of that system.
- The current number of attempts to contact the system

## Configuration Manager Output

The output from the NetEx Configuration manager is stored in the same directory as the PAMDU program is installed. The file name is *pamduCM.log*. This may be useful in determining why a PAM file was not generated.

## Lock File

The *pamdu* program will create a lock file when it is started for transferring data. It will delete the file when the program terminates. *pamdu* will catch the CNTL-C signal and attempt to gracefully terminate. If this file exists when *pamdu* is started, it will terminate immediately (check commands are excluded from this condition). The “*pamdu -u*” will delete the lock file. It is the user's responsibility to insure a second copy of *pamdu* is currently not running.

## TCP Connections

*pamdu* attempts to establish a TCP connection to each of the PAMDU spokes. In the initial handshake, the software versions and buffer sizes will be established. When the spoke system has successfully received and written the PAM file to disk, the spoke will return a string of “OK” to the hub system. If a return code is not received, or the return code is not “OK”, *pamdu* will leave the status of the host system as pending, close the connection and move on to the next system.

## *pamdu* (Hub) Security Considerations

*pamdu* must be run with read, write and create access to the NCT\_DIRECTORY.

PAMDU\_PORT must be accessible through any firewalls.

The *pamdu* program must have access rights to use TCP functions if your security product requires this.



## ***pamdu* Sample Scenarios**

### **Send everyone a network configuration**

```
./pamdu -n nct02
```

This will cause each spoke to be contacted. They will receive NCT02 and create a new PAM file

### **Testing a new network configuration**

```
./pamdu -n nct03 host1 host2
```

This will send the new network configuration to host1 and host2 only. The other hosts will not be contacted.

When host1 and host2 have tested the configuration,

```
./pamdu -r
```

This will restart *pamdu* and send NCT03 to all hosts except host1 and host2. (They already have a copy).

### **Recovering a computer system**

```
./pamdu -s host3
```

This will ship the current network configuration to host3 only. Other hosts will not be contacted. This would be used in the case of an accidental file deletion or in a disaster recovery situation.

### **Determine which spokes have not received the current network configuration**

```
./pamdu -e
```

This will show the status of all spokes that have not yet acknowledged the new network configuration

### **An error occurred when pamdu was processing and the lock file remains**

```
./pamdu -u
```

This will cause *pamdu* to erase its lock file, and become available for transferring data to the spokes. It is the user's responsibility to insure that another instance of *pamdu* is not running.

## ***pamdu* Debugging**

### **Additional Messages**

If you set the VERBOSE level to 5, more informational messages are recorded. Before the *pamdu.config* file is processed however, the default value of 2 is in effect. To set the VERBOSE level to 5 for this execution of the *pamdu* program, add a “-d” option after the program and before your run options:

```
./pamdu -d -n nct02
```

This will allow lower level messages to be recorded from the start of the program. The VERBOSE level from the configuration file will be ignored.

## CM Output Files

The information written to stdout from the NetEx configuration manager is written to pamduCM.log in the same directory as the *pamdu* program (/opt/netex/conf). This may help you resolve problems when CM fails to generate one or more PAM files.

The output from the NetEx Configuration Manager is also written to the NCT\_DIRECTORY. In this directory, a sub-directory is created called "CURRENT\_VERSION". This has a subdirectory for each NetEx host found in the NCT. In these subdirectories are the PAM files generated for each host. If the PAM file is missing, the Configuration Manager did not create one for this NCT-HOST combination. These PAM files may not be compatible with the host they were generated for. These are just validation files.

# ***pamlstn* (Spoke component)**

## ***pamlstn* (Spoke) Configuration Parameters**

### ***pamlstn* - Configuration File**

This configuration file (*pamlstn.config*) contains fixed name tags and values which allow the site to customize the operation of *pamlstn*. It will be located in the directory where the *pamlstn* utility was installed (see operating system defaults). If it is not available when *pamlstn* is started, *pamlstn* will terminate. This file may be empty. The default values are described below. This is a text file of character strings in the local host character set and edited manually. Lines with a # in the first column will be interpreted as comments and be ignored. Tags that are not known by *pamlstn* will be flagged as errors. The PAM\_FILE is received from the NetEx\_DEFAULT file. If it was not coded, the default listed below will be used.

### **Configuration Parameters for *pamlstn***

<b>Fixed Name Tag</b>	<b>Value</b>	<b>Description</b>
PAM_FILE	Path/filename	This is the complete path and filename to which the PAM file will be written. This should be the filename NetEx is using to read the PAM from.  Default: NetEx Default Configuration file-> PAM file
PAM_OLD	Path/filename	This is the complete path and filename to which the PAM file will be saved, in case it needs to be restored.
NETEX_DEFAULT	Filename	The directory name and file name of the default configuration file used by NetEx, when NetEx is started. <i>pamlstn</i> looks for the NetEx hostname and PAM file location.
PAMDU_PORT	Portnumber	This is the well-known default TCP port number to use to connect to the hub. This is the port number that was specified in the NCT for the HUB to contact this spoke on.

Fixed Name Tag	Value	Description
VERBOSE	Level	Verbosity level of activity which is output to the system log. All information is logged to stdout or sysout for z/OS.  <b>0</b> – no logging <b>1</b> – catastrophic errors logged only <b>2</b> – failures logged <b>3</b> – all activity logged <b>4</b> – not used <b>5</b> – Extensive diagnostic logging
PAM_RECEIVED	command	Command the spoke utility will execute when a new PAM has been successfully received and can be loaded into NetEx. This is not available on z/OS.
PAM_IS_LOADED	Command	This command is executed after the PAM_RECEIVED command has been executed. This is not available on z/OS.

## ***pamlstn* - Configuration File User Exits**

The spoke facility allows the user to execute two separate commands when a PAM file has been successfully received. These commands are not available on z/OS. z/OS users should use their normal procedures for taking an action when a message is written to the system log. (PAMLS023 PAM file Received - Connection free) For Unix systems, you must include the ./ for the current directory, see example below. For Windows Server systems, you must include the .\ for the current directory, and the “.bat” extension must be included (see below). These scripts must be in the same directory that *pamlstn* was installed into.

	Unix Systems	Windows Systems
Config Statement	PAM_RECEIVED=./cmdrec	PAM_RECEIVED=.\cmdrec.bat
Command to execute	cmdrec	cmdrec.bat
File contents:	#!/bin/bash echo “pam is received from cmdrec”	echo “hello” dir

## Configuration Defaults for *pamlstn* by Operating System

HP-UX	<p>PAM_FILE /etc/opt/netex/ntx_pam  PAM_OLD /etc/opt/netex/PAMOLD  NETEX_DEFAULT /etc/opt/netex/ntx_default  PAMDU_PORT 6950  VERBOSE 2  PAM_RECEIVED “ “  PAM_IS_LOADED “ “</p>
SOLARIS	<p>PAM_FILE /usr/nesi/netex/site/ntx_pam  PAM_OLD /usr/nesi/netex/site/PAMOLD  NETEX_DEFAULT /usr/nesi/netex/site/ntx_default  PAMDU_PORT 6950  VERBOSE 2  PAM_RECEIVED “ “  PAM_IS_LOADED “ “</p>
SuSE Linux	<p>PAM_FILE /etc/opt/netex/ntx_pam  PAM_OLD /etc/opt/netex/PAMOLD  NETEX_DEFAULT /etc/opt/netex/ntx_default  PAMDU_PORT 6950  VERBOSE 2  PAM_RECEIVED “ “  PAM_IS_LOADED “ “</p>
WINDOWS	<p>PAM_FILE C:\NESI\netex\ntx_pam  PAM_OLD C:\NESI\netex\PAMOLD  NETEX_DEFAULT c:\NESI\netex\site\ntx_default  PAMDU_PORT 6950  VERBOSE 2  PAM_RECEIVED “ “  PAM_IS_LOADED “ “</p>

z/OS	PAM_FILE //DDN:PAMFILE PAM_OLD //DDN:PAMOLD NETEX_DEFAULT //DDN:NETEX PAMDU_PORT 6950 VERBOSE 2 PAM_RECEIVED “ “ PAM_IS_LOADED “ “
------	--

## ***pamlstn* (Spoke) Command line Options**

The command line interface will support the following commands:

<code>pamlstn</code>	This will invoke the <i>pamlstn</i> utility. It will read in a configuration, if necessary, and post a TCP listen for the <i>pamdu</i> to connect. Messages will be sent to the system log and to stdout.
<code>pamlstn -r</code>	This will restore the file before the last PAM was received and listen for a new PAM file.
<code>pamlstn -f pathname</code>	The path name points to a directory where the <code>pamlstn.config</code> file is located.
Ctrl-c, Kill -1 Kill -2 Kill -3	<code>pamlstn</code> will trap these signals and attempt to gracefully terminate.

## ***pamlstn* (Spoke) Other Files**

### **NetEx Default file**

This is a read-only file. *pamlstn* reads in NetEx configuration files and extracts two values. The first value is the “local” parameter. This is the NetEx host name for this system. This is used by the Configuration Manager to generate the PAM. The second item is the PAM file location. This specifies the location to write the updated PAM file to.

### **PAM\_OLD**

Before *pamlstn* invokes the Configuration Manager to write the new PAM file, *pamlstn* stores the current PAM file at this location. If *pamlstn* is started with the `-r` option, the file pointed to by PAM\_OLD will be restored to the PAM-file location, before posting a TCP listen for an updated NCT file. This allows the remote user to recover to a previous configuration, without an action from the hub. The Hub will still show the spoke has received the new network information, but has no knowledge that the spoke has recovered to a previous version.

## ***hostname.PAMDUCM.RLOG***

This file is in the `pamdu_work` directory. This directory is located in the same directory as the `pamlstn.config` file. This contains the stdout output from the NetEx Configuration Manger. It may help determine why a PAM file was not generated.

## ***hostname.PAMDUCM.RLOGE***

This file is in the `pamdu_work` directory. This directory is located in the same directory as the `pamlstn.config` file. This contains the stderr output from the NetEx Configuration Manger. It may help determine why a PAM file was not generated.

## ***hostname.CMINPUT***

This file is in the `pamdu_work` directory. This directory is located in the same directory as the `pamlstn.config` file. This contains the stdin input to the NetEx Configuration Manger. It may help determine why a PAM file was not generated.

## ***pamlstn (Spoke) Security Considerations***

*pamlstn* must be run with read, write and create access to the PAM file used by NetEx and read/write access to the directory where the NetEx Configuration Manage is installed.

PAMDU\_PORT must be accessible through any firewalls.

*pamlstn* program must have access rights to use TCP functions if your security product requires this.





# Installation Procedures

## SuSE Linux

### HUB Component

The hub component installs both the hub and the spoke components. The user will receive a file labeled “suse\_pamdu.ms”. It may be placed in any directory on your system. This file should be executed with root authority (`./suse_pamdu`). This is a self-extracting file that when executed will create a temporary directory and extract all software into this directory. It will invoke an install script to copy the files into the correct locations on your system.

The user will be prompted for the directory into which the software will be installed. This directory must be the same directory that the NetEx configuration manager is installed (`/opt/netex/conf`). The script will check for `cm`. The script will also prompt for the man page directory. The default is `/usr/local/man/man1`.

### Spoke Component

The spoke component installs *pamlstn* as a daemon. The user will receive a file labeled “suse\_pamlstn.ms”. It may be placed in any directory on your system. This file should be executed with root authority (`./suse_lstn.ms`). This is a self-extracting file that when executed will create a temporary directory and extract all software into this directory. It will invoke an install script to copy the files into the correct locations on your system.

The user will be prompted for the directory into which to install the software. This directory must be the same directory that the NetEx configuration manager is installed (`/opt/netex/conf`). The script will check for `cm`. The script will also prompt for the man page directory. The default is `/usr/local/man/man1`.

The script will then copy all modules to the correct location. *pamlstn* will be installed as a daemon that is started by the boot process.

Daemon commands are executed from `/etc/init.d`

```
./pamlstnd start
```

```
./pamlstnd status
```

```
./pamlstnd stop
```

All output for the *pamlstn* daemon goes to the system log (`/var/log/messages`).

## Solaris

### Spoke Component

The spoke component installs *pamlstn* as a daemon. The user will receive a file labeled “solaris\_pamlstn.ms”. It may be placed in any directory on your system. This file should be executed with root authority (`./solaris_lstn.ms`). This is a self-extracting file that when executed will create a temporary directory and extract all software into this directory. It will invoke an install script to copy the files into the correct locations on your system.

The user will be prompted for the directory into which the software is to be installed. This directory must be the same directory that the NetEx configuration manager is installed (`/usr/nesi/netex/conf`). The script will check for `cm`. The script will also prompt for the man page directory. The default is `/usr/local/man/man1`.

The script will then copy all modules to the correct location. *pamlstn* will be installed as a daemon that is started by the boot process.

Daemon commands are:

```
svcadm -v enable pamlstnd    (start)
svcadm -v disable pamlstnd   (stop)
svcs -x pamlstnd             (display problems)
svcs pamlstnd                (status command)
```

All output for the *pamlstn* daemon goes to the system log (`/var/adm/messages`).

## HP-UX (PA-RISC)

### Spoke Component

The spoke component installs *pamlstn* as a daemon. The user will receive a file labeled “`hpux_pamlstn.ms`”. It may be placed in any directory on your system. This file should be executed with root authority (`./hpux_lstn.ms`). This is a self extracting file that when executed will create a temporary directory and extract all software into this directory. It will invoke an install script to copy the files into the correct locations on your system.

The user will be prompted for the directory into which the software is to be installed. This directory must be the same directory that the NetEx configuration manager is installed (`/opt/netex/conf`). The script will check for `cm`. The script will also prompt for the man page directory. The default is `/usr/local/man/man1`.

The script will then copy all modules to the correct location. *pamlstn* will be installed as a daemon that is started by the boot process.

Daemon commands are executed from `/sbin/init.d`

```
./pamlstnd start
./pamlstnd stop
```

All output for the *pamlstn* daemon goes to the system log (`/var/adm/syslog/syslog.log`).

## z/OS

### Spoke Component

**NOTE: If FTP is used to transfer the two distribution files to z/OS, your FTP session must use BINARY mode. You must issue a ‘quote site lrecl=80 blksize=3200 recfm=fb’ command before you issue the PUT for NETEX.ZOS.PAMLSTN.JCL and NETEX.ZOS.PAMLSTN.PKG files.**

PAMLSTN for z/OS is distributed as two files. The file names are NETEX.ZOS.PAMLSTN.JCL (the installation job) and NETEX.PAMLSTN.PKG. Both files may be transferred with FTP to your system (see note above).

The user should edit NETEX.ZOS.PAMLSTN.JCL (the installation job). This job will create two datasets. The first, “NETEX.ZOS.JCL”, contains two members: one for allocating files and the other for running PAMLSTN. The second file is the NETEX.PAMLSTN.LOAD. This contains the executable code. If the dataset names are acceptable, you can submit the NETEX.ZOS.PAMLSTN.JCL (INSTALL1 job). This should complete with condition codes of 0.

The user should then edit the member NETEX.ZOS.JCL(XALLOC) in the dataset created. If you want the default HLQ of “NETEX.PAMLSTN”, you may submit the job. This allocates various work files used by PAMLSTN, and the NetEx configuration manager. This should complete with a condition code of 0.

The user should then edit the NETEX.ZOS.JCL(XPAMLSTN) member. This is JCL that will start the PAMLSTN utility. The user should point the NETEX DD statement to the dataset and member that contains the NetEx configuration parameters. (Look at your NetEx proc for the NTXINIT DD statement). The user should change the PAMFILE DD statement to point to the PAM file NetEx reads to determine the network configuration. (Look at your NetEx proc for the PAMFILE DD statement). The SYSOUT and SYSERR DD statements contain the output from the PAMLSTN program.

## Windows Server

### Spoke Component

The spoke component installs *pamlstn*. The user will receive a file labeled “pamlstn.zip”. This is a Windows .ZIP file. The user should copy this file into the directory that contains the NetEx configuration manager (c:\nesi\netex\conf). The user should right click on this .ZIP file and select “extract all”. In the popup window, the user should delete the *pamlstn* extension on the file name. The modules will then be extracted into c:\nesi\netex\conf (the location that has the cm executable).

The user may then select from the programs menu, the startup entry with a right click. This will open the startup folder. The user may select open, new shortcut and browse the C:\nesi\netex\conf directory for the pamlstn.exe module. Press OK. Press next, and Finish. A shortcut will be placed in the startup folder. Output from *pamlstn* goes to the system log. This may be browsed using the Event Viewer. When installing, it is often easier to open a command window, change to the C:\nesi\netex\conf directory and “exec pamlstn.exe”. Messages will be displayed to the screen and recorded in Window’s Event Viewer.



# Appendix A. *pamdu* Error Messages

This appendix is intended to give users more information about *pamdu* messages that may be seen during a session. All of the messages have been listed along with details describing them. Many of the messages are self-explanatory (e.g., “Invalid command”) and require no further discussion. At the end of this message table is a list and description of the messages that require further explanation.

PAMDU001	“Version %s, on Host=%s”
PAMDU002	“Exception command does not accept parameters”
PAMDU003	“Duplicate commands were entered”
PAMDU004	“Restart command does not accept parameters”
PAMDU005	“Unlock command does not accept parameters”
PAMDU006	“New requires a NCT file name”
PAMDU007	“Force requires a NCT file name”
PAMDU008	“Valid commands are:”
PAMDU009	“Send requires a NetEx hostname”
PAMDU010	“Hosts entered on the command line”
PAMDU011	“Configuration File is:”
PAMDU012	“Configuration File failed to open”
PAMDU013	“VERBOSE must be 1 through 5”
PAMDU014	“PAM_RETRIES must be 0 through 99”
PAMDU015	“CHECKPOINT file was not specified”
PAMDU016	“NCT_DIRECTORY was not specified”
PAMDU017	“PAMDU_PORT was not specified”
PAMDU018	“STOP_ON_ERROR was not specified”
PAMDU019	“STOP_ON_ERROR must be YES or NO”
PAMDU020	“MAX_SESSIONS was not specified”
PAMDU021	“IGNORE_HOST was not specified”
PAMDU022	“Configuration values in effect:”
PAMDU023	“NCT directory and file must be less than 200 characters”
PAMDU024	“NCT File open failure:”
PAMDU025	“Read error on NCT =”
PAMDU026	“Maximum number of NetEx Hosts is 200”
PAMDU027	“Address length exceeded”
PAMDU028	“Port number was not specified for host:”
PAMDU029	“PAMDU CHECKPOINT FILE:”

PAMDU030 “Checkpoint file failed open”

PAMDU031 “Write of checkpoint failed rc=”

PAMDU032 “Checkpoint failed to open for reading”

PAMDU033 “EOF on checkpoint file”

PAMDU034 “IGNORE\_HOST %s, not found in NCT”

PAMDU035 “Host %s entered on the command line was not found in the NCT”

PAMDU036 “PAMDU.LOCK file exists. Only one pamdu may run at a time”

PAMDU037 “CM log is:”

PAMDU038 “File stat failed for”

PAMDU039 “The PAM file size for host is 0”

PAMDU040 “Kill received - Terminating”

PAMDU041 “LogUser= Remuser= Host= Thread=”

PAMDU042 “make\_connection failed for “

PAMDU043 “Initialization handshake failed phase 1 rc=”

PAMDU044 “Initialization handshake failed phase 2 rc=”

PAMDU045 “Initialization handshake failed phase 3 rc=”

PAMDU046 “Initialization handshake failed phase 4 rc=”

PAMDU047 “Buffer=”

PAMDU048 “File =” PAM file path

PAMDU049 “PAMDU CHECKPOINT FILE =”

PAMDU050 Checkpoint titles

PAMDU051 “Failed to Send Version rc=”

PAMDU052 “Version Number acknowledgement failed rc=%d”

PAMDU053 “Version Read failed rc=”

PAMDU054 “Open of NCT failed”

PAMDU055 “TCP write of PAM file failed”

PAMDU056 TBD

PAMDU057 “Failed to read return code”

PAMDU058 “Expected OK received:”

PAMDU059 “NCT transmitted connection is closed”

PAMDU060 “pamdu.lockfile has been deleted”

PAMDU061 “pamdu.lockfile could not be deleted”

PAMDU061 “Current NCT version found in checkpoint file”

PAMDU062 “NCT Version found in checkpoint file”

PAMDU063 “Host needs PAM file generated”

PAMDU064	TCP messages
PAMDU065	TCP messages
PAMDU066	TCP messages
PAMDU067	“MAXSESSIONS must be 1 through 10”
PAMDU068	“TCP messages”
PAMDU069	“TCP messages”
PAMDU070	“Back from read rc= buffer= hex format/character fmt”
PAMDU071	“Hub client Footprint does not match”
PAMDU072	“Read error on NCT =”
PAMDU073	“CM input file failed to open”
PAMDU074	“Temporary file failed to open – check file permissions”
PAMDU075	“Pamdu Version Copyright 2010 Network Executive Software”
PAMDU076	“lock file=”
PAMDU077	“command code=”
PAMDU078	“Opening PAM file=”
PAMDU079	“NCT version=”
PAMDU080	“NCT host=”
PAMDU081	“PAMDU terminating – STOP_ON_ERROR = YES”
PAMDU082	“Host name must be less than 9 characters”
PAMDU083	“Checkpoint file written records=”
PAMDU084	“parm1= ,parm2=”
PAMDU085	“Host entered on the command line and ignored:”
PAMDU086	“make_connection: Invalid network address –”





## Appendix B. *pamlstn* Error Messages

This appendix is intended to give users more information about *pamlstn* messages that may be seen during a session. All of the messages have been listed along with details describing them. Many of the messages are self-explanatory (e.g., “Invalid command”) and require no further discussion. At the end of this message table is a list and description of the messages that require further explanation.

PAMLS001	“Version %s, on Host %s”
PAMLS002	“Opening Configuration file”
PAMLS003	“Configuration file failed to open - using defaults”
PAMLS004	“Read error on the config file = %d”
PAMLS005	“Missing parameter in config file for %s”
PAMLS006	“VERBOSE values are 0 thru 5”
PAMLS007	“Configuration Values in Effect:”
PAMLS008	“TCP connection is coming up”
PAMLS009	“Unable to find TCP service for”
PAMLS010	“Bad Handshake to Hub services phase 1 rc=”
PAMLS011	“Bad Handshake to Hub services phase 2 rc=”
PAMLS012	“Bad Handshake to Hub services phase 4 rc=”
PAMLS013	“Kill received - terminating”
PAMLS014	“Read of Version failed rc=”
PAMLS015	“Expected VERSION=nn Found %s”
PAMLS016	“Send acknowledgement of version failed rc=”
PAMLS017	“CM failed to create a PAM file”
PAMLS018	“NCT open failed”
PAMLS019	“Receiving NCT failed rc=”
PAMLS020	“Receiving PAM file completed”
PAMLS021	“Write of final ACK failed rc=”
PAMLS022	“Connection to HUB is up”
PAMLS023	“PAM file Received - Connection free”
PAMLS024	“NCT= Version= Size=”
PAMLS025	“Unknown configuration input, input=”
PAMLS026	“Configuration Manager failed load”
PAMLS026	“Back from hand1 sock= rc=”
PAMLS027	“SPOKE is running Hub is running NO Translation”
PAMLS028	“RHOST= LOGUSER= REMUSER=”

PAMLS029 "NETEX\_DEFAULT file filed open %s,"

PAMLS030 "Read error on NETEX\_DEFAULT file rc=%d"

PAMLS031 "Missing NetEx configuration parameter for"

PAMLS033 "CM failed to generate a PAM file"

PAMLS034 "Open of CM temp file failed"

PAMLS035 "Receive of PAM file failed"

PAMLS036 "The old PAM file was not found. Pam not recovered"

PAMLS037 "PAM file restored to the previous version"

PAMLS038 "The current PAM file could not be opened. Pam not recovered"

PAMLS039 "Read error on the current PAM file. Pam not recovered."

PAMLS040 "Backup of pam failed - transfer aborted."

PAMLS041 "The old PAM file was not found. Pam transfer aborted"

PAMLS042 "The current PAM file was not found. Pam transfer aborted"

PAMLS043 "Connected sock="

PAMLS044 "Invalid parm passed to program parm="

PAMLS045 "Program installed in %s"

PAMLS046 "Using hostname= from the NetEx default file"

PAMLS047 "Using PAM file location= from the NetEx default file"

PAMLS048 "CM failed RC="

PAMLS049 "Couldn't load Sub program:"

PAMLS050 error messages from execlp

PAMLS051 error messages from execlp

PAMLS052 "pamlstn Version Copyright 2010 Network Executive Software"

PAMLS053 "PAM file not found:"

PAMLS054 "Create work directory failed"