Secure TNP

TCP/IP NetEx Proxy

Release 1.0

Software Reference Manual

Revision Record

Revision	Description
1.0	• Product Release

You may submit written comments to:

Network Executive Software, Inc.
Publications Department
6450 Wedgwood Road N Suite 103
Maple Grove, MN 55311
USA

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

support@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 TCP/IP NetEx™ Proxy (TNP) software for supported operating systems.

The current Secure TNP products are:

- TNP804 for the Linux/Oracle Linux operating system on x86 platforms.

“Chapter 1: Introduction”, Chapter 2: Overview”, and “Chapter 3: Installation are intended for all readers.

“Chapter 4: Secure TNP Error Codes” includes a list and description of the error messages and codes issued by Secure TNP.

Readers are expected to be familiar with NetEx/IP before using this manual.

Reference Material

The following manuals contain related information.

Number	Title and Description
Man-Ref-H367I	*NetEx Requester for HP NonStop on Integrity/Blade Platforms*
Man-Ref-H297	*NetEx Requester for GCOS8 on Bull Systems*
Man-Ref-H267	*NetEx Requester for HP OpenVMS on Integrity Platforms*
Man-Ref-H804	*Secure NetEx/IP for Linux or Windows OS on x86 Platforms*

Notice to the Reader

The material contained in this publication is for informational purposes only and is subject to change without notice. Network Executive Software is not responsible for the use of any product options or features that are not described in this publication, and assumes no responsibility for any errors that may appear in this publication. Refer to the revision record (at the beginning of this document) to determine the revision level of this publication.

Network Executive Software does not by publication of the descriptions and technical documentation contained herein, grant a license to make, have made, use, sell, sublicense, or lease any equipment or programs designed or constructed in accordance with this information.

This document may contain references to the trademarks of the following corporations:

Corporation Trademarks and Products

Network Executive Software	**NetEx, NetEx/IP, BFX, PFX, USER-Access, eFT**
The Open Group	**UNIX**
Linus Torvalds	**Linux**
HP Corporation	**HP, NonStop, Integrity, OpenVMS**
Bull Groupe	**Bull**

These references are made for informational purposes only.

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

Notice to the Customer

Installation information contained in this document is intended for use by experienced System Programmers.

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 `this font`.
`user entry`	*`This font`* 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.
bold	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.

Glossary

buffer: A contiguous block of memory allocated for temporary storage of information in performing I/O operations. Data is saved in a predetermined format. Data may be written into or read from the buffers.

host: A data processing system that is connected to the network and with which devices on the network communicate. In the context of Internet Protocol (IP), a host is any addressable node on the network; an IP router has more than one host address.

Internet Protocol (IP): A protocol suite operating within the Internet as defined by the *Requests For Comment* (RFC). This may also refer to the network layer (level 3) of this protocol stack (the layer concerned with routing datagrams from network to network).

ISO: Acronym for International Standards Organization.

Secure NETwork EXecutive (NetEx): A family of software designed to enable two or more application programs on heterogeneous host systems to communicate. Secure NetEx is tailored to each supported operating system, but can communicate with any other supported Secure NetEx, regardless of operating system.

Secure NetEx can reside on the host or another host with Secure TNP.

NetEx is a registered trademark of Network Executive Software.

Open Systems Interconnection (OSI): A seven-layer protocol stack defining a model for communications among components (computers, devices, people, and etcetera) of a distributed network. OSI was defined by the ISO.

path: A route that can reach a specific host or group of devices.

TCP/IP: An acronym for Transmission Control Protocol/Internet Protocol. These communication protocols provide the mechanism for inter-network communications, especially on the Internet. The protocols are hardware-independent. They are described and updated through *Requests For Comment* (RFC). IP corresponds to the OSI network layer 3, TCP to layers 4 and 5.

Contents

Revision Record ..... **ii**

Preface ..... **iii**

Reference Material ..... **v**

Notice to the Reader ..... **vii**
Corporation Trademarks and Products vii
Notice to the Customer vii
Document Conventions viii
Glossary ix

Contents ..... **xi**
Figures xiii

Chapter 1: Introduction **1**
External Interface 1
Basic I/O Flow 1
Secure NetEx (Host Based) 1
Secure NetEx/IP via TNP 1

Chapter 2: Overview **3**
Secure TNP Configuration 3
TNP PROGRAM 3
tnp.cfg 3
Secure TNP Log File 4
Resolving NetEx/IP Requester Hosts to Use TNP 5
Requester Configuration 5

Chapter 3: Installation **7**

Chapter 4: Secure TNP Error Codes **9**
Secure TNP Startup Exit Codes 9

Appendix A: Messages **11**

Appendix B: TNP804 Linux Installation **13**
Prerequisites 13
Hardware Installation 13
Accessing the TNP804 software distribution 13
Getting the NESi Public Key 13
Importing the NESi Public Key 13
Verifying Signatures 13
Software Installation 13
Upgrading TNP804 14
Removing TNP804 14
Removing the NESi Public Key 14

Starting, Stopping & Verifying Install of Secure TNP 14
Post Installation Considerations 15
Configuring TNP804 15
1. Verify Secure NetEx/IP is configured and operational 15
2. Configure Secure TNP 15
3. Optionally, setup syslog file definition 15
4. Start Secure TNP 15
5. Optionally, manage the Secure TNP logs 15

Figures

Figure 1. Basic I/O Flow.....1
Figure 2. Example TNP custom configuration file.....4

Chapter 1: Introduction

Network Executive Software's NetEx/IP™ allows two or more application programs (which may be on different host computers) to communicate with each other at multi-megabit speeds. The NetEx/IP family of software consists of different versions of NetEx/IP for use with different operating systems, such as the versions for use with the various UNIX operating system hosts. All of these versions provide a common high-level interface to simplify programming requirements. NetEx/IP utility programs are also available, such as the Bulk File Transfer (BFX™), Print File Transfer (PFX™), and USER-Access® utilities. Refer to the NetEx/IP or NetEx Requester documents for more details on NetEx/IP.

External Interface

The NetEx/IP external interface for the application programmer is common for all versions of NetEx/IP. NetEx/IP provides requests for use in the programs that call NetEx/IP. These calling programs may be written in assembler, C or other high-level languages. NetEx/IP programs written in high-level languages may be transported from one host to another, with some changes to account for different word sizes and other machine architecture variations.

Basic I/O Flow

Figure 1 shows the basic I/O flow between two programs using host based NetEx/IP. The calling program communicates with NetEx/IP through the NetEx/IP user interface. NetEx/IP then uses the available network hardware to communicate with the calling program on the other processor.

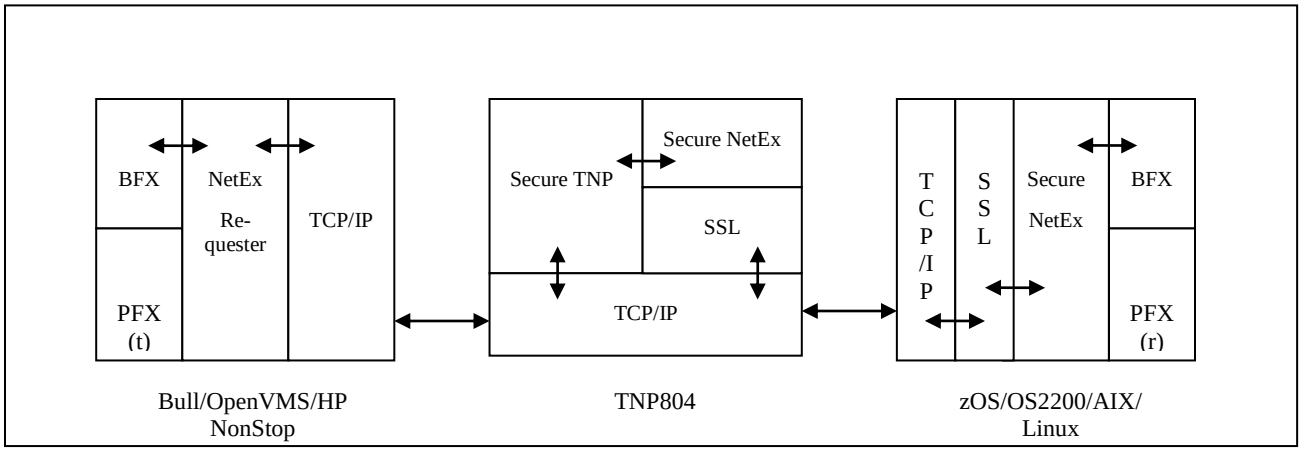

Figure 1. Basic I/O Flow

Secure NetEx (Host Based)

Secure NetEx exists on a machine and is part of the NetEx application. User tasks produce a NetEx request that is delivered to Secure NetEx. Data is moved so it is present in the NetEx program and the I/O is performed by NetEx.

Secure NetEx/IP via TNP

The Secure NetEx/IP program may reside in another host running TNP. Only the Hxx7IP NetEx Requester user interface program resides on the local host. In this implementation, H267IP the OpenVMS TCP/IP

product, H367IP the HP Guardian TCPIP product, or H297IP the Bull product, is used for transport of NetEx requests and buffers between the H267IP (H367IP or H297IP) host and the Secure TNP NetEx.

Chapter 2: Overview

The Secure TNP product serves as a NetEx/IP "proxy", which can be used by other hosts for which a host-based Secure NetEx/IP is not available. A NetEx/IP Requester, residing on an HP NonStop, OpenVMS, Bull, or Stratus server, works with BFX, USER-Access, eFT, PFX (or other NetEx/IP applications running on the Requester server), and reads and writes the application's NetEx/IP requests over a TCP/IP connection to Secure TNP, which then passes the request to Secure NetEx/IP, on behalf of the Requester's NetEx/IP application. In effect, Secure TNP serves as a "proxy" NetEx/IP application for the Requester's NetEx/IP applications.

Secure TNP is a separately licensed product. Secure TNP is started after Secure NetEx/IP initialization on the same host, and is ready to accept connections from NetEx/IP Requester applications running on different servers. When a NetEx/IP request comes in from a Requester, Secure TNP becomes the local Secure NetEx/IP application, acting on behalf of the Requester. Secure TNP uses the NetEx/IP API's to communicate with the local NetEx/IP, just like any other local NetEx/IP applicaion. The Requester application is able to esablish Secure NetEx/IP sessions with other remote Secure NetEx/IP hosts on the network, as well as with local Secure NetEx/IP. Connections established between Requester applications and local NetEx/IP effectively appear as NetEx/IP intrahost connections.

Secure TNP Configuration

TNP PROGRAM

When the TNP program is installed and the default tnp.cfg file is put in /usr/share/nesi/tnp. However, this file can also be edited and changed, if needed, anytime after the installation. The TNP program must be started after the Secure NetEx/IP initialization. The TNP program then waits for work requests from remote NetEx/IP Requesters.

tnp.cfg

tnp.cfg is the TNP configuration file, and is shown in Figure 2. The default tnp.cfg file is installed during installation. However, this file can also be edited and changed, if needed, anytime after the TNPxx4 installation.

PORT keyword specifies the port number used with NetEx/IP Requester hosts, and must be the same as the port specified by the 'TCP' parameter in the NetEx/IP Requester host configuration file.

DEBUG specifies a debugging level (on/on2/off default is off).

TRACE is the path to the TNP log (default is /usr/share/nesi/tnp/tnp.log).

MSGSYSLOG specifies where TNP messages will be output. A value of 1 means output to the tnp.log. A value of 2 means output to the system log. A value of 3 means output to both logs. A value of 0 means no syslog and minimal log output.

MSGSYSLOGFAC specifies how syslog messages will be output. A default syslog facility code of "local3" will be used unless overridden. The valid facility codes must be used if overridden. It is recommended only the local facility codes be used. Check with your system administrator for more information.

```
# ****************************************************************************
# *
# *     COPYRIGHT (C) 1999-2018, Network Executive Software, Inc. (NESi)
# *
# *     Network Executive Software, Inc., Maple Grove, MN
# *
# *     THIS SOFTWARE FURNISHED UNDER A LICENSE FOR USE ONLY ON A SINGLE
# *     COMPUTER SYSTEM AND MAY NOT BE COPIED EXCEPT FOR THE PURPOSE OF
# *     CREATING A BACKUP COPY FOR THE SYSTEM FOR WHICH IT WAS LICENSED.
# *     TITLE TO AND OWNERSHIP OF THIS SOFTWARE SHALL AT ALL TIMES REMAIN
# *     WITH NETWORK EXECUTIVE SOFTWARE, INC.
# *
# *     THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
# *     AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY NETWORK EXECUTIVE
# *     SOFTWARE INC.
# *
# ****************************************************************************
#
# TNP configuration file
#
# All keywords and boolean values are case-insensitive.
# Boolean values: on, off
# Comment indicators: # * ! " /
# Format: TNP <keyword> <value>
#
# Keyword       Default    Description
# -------       -------    -----------
# PORT          5001       Port number used with NetEx/IP requesters
# DEBUG         OFF        Debug messages
#                          OFF = No debug messages
#                          ON  = Enable debug messages, NRB tracing and
#                                limited user data tracing (16 bytes maximum)
#                          ON2 = Enable full user data tracing
# TRACE                    Path to TNP message log
#                          Default = /usr/share/nesi/tnp/tnp.log
# MSGSYSLOG     1          Where TNP log messages will be sent
#                          0 = None (minimal message log output)
#                          1 = Message log only (set by TRACE keyword)
#                          2 = Syslog only
#                          3 = Message log and syslog
# MSGSYSLOGFAC  local3     Syslog facility
#

TNP port 5001
TNP debug OFF
TNP trace /usr/share/nesi/tnp/tnp.log
TNP msgsyslog 1
TNP msgsyslogfac local3
```

Figure 2. Example TNP custom configuration file

Secure TNP Log File

Secure TNP log file (defined by the keyword TRACE in tnp.cfg) is not managed (i.e. unlimited growth) and must be monitored and managed manually.

The TRACE value ‘console’ will write messages to stdout. This can be used on Linux systems that have systemd-journald to allow a system service to store and manage TNP messages.

The log file contains messages from Secure TNP as specified by the MSGSYSLOG and DEBUG keywords in tnp.cfg. To disable all logging set the following in tnp.cfg:

TNP MSGSYSLOG 0

TNP DEBUG OFF

Resolving NetEx/IP Requester Hosts to Use TNP

For TNP configurations, the NetEx/IP Requester host will need to resolve to use TNP (i.e. add the Netex Requester Hostname and TNP IP address of the to DNS of the NetEx/IP on the other end of the session). In the event the requester host name used by TNP is the same as a IP host name, NTXhostname can be used in DNS to properly route traffic to the TNP host. Secure NetEx will try to resolve NTXhostname before hostname when establishing a connection.

Requester Configuration

The NetEx/IP Requester host also has a configuration file. Refer to the appropriate NetEx/IP Requester manual for a description of the configuration file. It contains the IP address and port number used when communicating with HXX4 TNP.

Chapter 3: Installation

Please refer to the appropriate appendix for installation and OS specific information for your operating system:

- Appendix B: TNP804 Linux Installation

Chapter 4: Secure TNP Error Codes

Refer to the Secure NetEx/IP manual for the platform of the host that is running Secure TNP/Secure NetEx for the NetEx Error Codes. Only Secure TNP error codes are described here.

Secure TNP Startup Exit Codes

If an error occurs while starting Secure TNP, a non-zero exit code and message will be produced. The message will be logged in the Secure TNP log if logging is enabled. The following table shows the possible codes and the appropriate Secure TNP log messages (where %s will be substituted with a character string and %d by a decimal number):

1 : error reading config file '%s'

2 : cannot specify daemond and use console as trace

3 : error opening trace file '%s': %s

4 : req socket() failed: %s

5 : bind() sock=%d port=%d failed: %s

6 : listen() sock=%d failed: %s

7 : daemon() failed: %s

8 : select() failed: %s

9 : accept() sock=%d failed: %s

Exit code 1 is a general Secure TNP configuration file error and there will be additional messages detailing the specific error. These messages start with "config : ".

Appendix A: Messages

This section contains a description of the messages issued by Secure TNP. These messages are displayed in the ‘tnplog’ or in the log or trace file which was set up at install time.

Depending on the logging service, each message may be prefixed with a date and timestamp. The messages are intended to be self-explanatory. Some messages are informational showing flow, while others are errors.

The following are examples of some of the messages:

```
error opening trace file '%s'
%s version %s
bind() sock=%d port=%d failed
tcp connect request from %s:%d (sock=%d)
```

Appendix B: TNP804 Linux Installation

Prerequisites

The following are hardware and software prerequisites for installing the TNP804 product.

- An Intel compatible system running a supported Linux OS. Review the website for supported OS distributions.
- At least one other processor on the network running Secure NetEx/IP software. This processor should be connected with another Secure NetEx/IP (not required for intra-host test/evaluation).
- Secure NetEx/IP software installed and operational on this processor (H804)

All requirements for the equipment listed above must be met before proceeding with the installation.

Hardware Installation

Install and verify proper operation of the appropriate operating system.

Accessing the TNP804 software distribution

The TNP804 software is available as an RPM which may be downloaded from NESi. Contact NESi Customer Support to request the download link.

Getting the NESi Public Key

The RPM software distribution package is signed to ensure integrity and authenticity. It is recommended to install the NESi public key and verify the signature of any software packages before installation.

You can download the key by visiting the documentation page for your version of NetEx/IP at http://www.netex.com/.

Importing the NESi Public Key

Install the public key as super user with the command:

```
# rpm --import RPM-GPG-KEY-netex.txt
```

Verifying Signatures

You can verify the RPM signature to ensure that a package has not been modified since it has been signed. Verification will also check that a package is signed by the vendors or packagers key.

To verify the signature, use the -K or –checksig option to the rpm command:

```
# rpm -K TNP804-1.0.xxxx.rpm
```

Software Installation

If this is an initial installation, install the software as super user with the command:

```
# rpm -i TNP804-1.0.xxxx.rpm
```

If the NESi public key has not been installed use the command:

```
# rpm –i --nosignature TNP804-1.0.xxxx.rpm
```

Upgrading TNP804

Using the “rpm -U” command preserves any customized files in this package and the replacement files are installed with extensions of “.rpmnew”. Any files that are not in the package but in package directories will also be preserved. Upgrade the software as super user with the command:

```
# rpm –U TNP804-1.0.xxxx.rpm
```

If the NESi public key has not been installed use the command:

```
# rpm –U --nosignature TNP804-1.0.xxxx.rpm
```

Removing TNP804

During RPM removal, any modified configuration files and log files will not be deleted. Remove the software as super user with the command:

```
# rpm –e TNP804
```

Removing the NESi Public Key

To remove the NESi public key, as super user issue the command:

```
# rpm –e gpg-pubkey-3d6b35d3-51bb5907
```

Starting, Stopping & Verifying Install of Secure TNP

The following commands should be used to stop, start and restart Secure TNP:

For System 5 init (ex. RHEL 6):

```
# service tnp stop
# service tnp start
# service tnp restart
```

For systemd (ex. RHEL 7):

```
# systemctl stop tnp.service
# systemctl start tnp.service
# systemctl restart tnp.service
```

The following command should be used to verify installation:

For System 5 init (ex. RHEL 6):

```
# chkconfig  --list  tnp
```

For systemd (ex. RHEL 7):

```
# systemctl list-units tnp.service
```

Post Installation Considerations

Configuring TNP804

Once the software package installation has been successfully completed, Secure TNP must be configured prior to execution. The following instructions address editing the configuration file for Secure TNP (**/usr/share/nesi/tnp/tnp.cfg**) and starting the Secure TNP process.

Setup of the TNP804 software is detailed in the following steps.

1. Verify Secure NetEx/IP is configured and operational

2. Configure Secure TNP

Refer to the section on TNP for pertinent information: Secure TNP Configuration beginning on page 3.

3. Optionally, setup syslog file definition

Secure TNP may be configured to send messages to syslog. The messages will have a syslog priority of err or info. An example syslog definition for creating a Secure TNP log is provided:

```
# syslog config for Secure TNP
local3.*    /var/log/tnp.log
```

4. Start Secure TNP

Refer to the section Starting, Stopping & Verifying Install of Secure TNP on page 14.

5. Optionally, manage the Secure TNP logs

Example logrotate definitions are also provided:

```
# Secure TNP log file
/usr/share/nesi/tnp/tnp.log {
      daily
      compress
      nocreate
      copytruncate
      nodateext
      maxage 365
      rotate 32
      missingok
      #notifempty
      ifempty
      create 644 root root
}

# Secure TNP syslog file
/var/log/tnp.log {
      daily
      compress
      nocreate
      copytruncate
      nodateext
      maxage 365
```

```
        rotate 32
        missingok
        #notifempty
        ifempty
        create 644 root root
    }
```