



**H897IPV NetEx/IP[®] Requester
for Stratus OpenVOS Operating Systems**

Release 2.1

Memo to Users

December 14, 2011

© 2011 Network Executive Software, Inc.
6420 Sycamore Lane N #300
Maple Grove, MN 55369

Introduction

This “Memo to Users” document contains information for H897IPV, and should be reviewed before beginning the installation process.

H897IPV is a new product that provides NetEx/IP Requester for Stratus V Series platforms. It is functionally equivalent to H897R, which was the comparable NetEx Requester product that ran on the Stratus Continuum platform. H897IPV uses a TNP component that is part of NetEx/IP for z/OS (H210IPZ Release 7.1), rather than using a DXE HYPERchannel adapter or NESiGate-LO adapter.

H897IPV is available by download.

The following manuals are available for the H89x Stratus products:

H892TV Print File Transfer (PFX) for Stratus OpenVOS Systems Software Reference Manual

H893V USER-Access for Stratus OpenVOS Systems Software Reference Manual

H897IPV NetEx/IP Requester for Stratus OpenVOS Systems Software Reference Manual

Note: these manuals may not be available at time of release, but will be available shortly thereafter. Since the OpenVOS versions of these products are logically equivalent to the Continuum versions, the Continuum versions of the manuals can be used as a reference until the OpenVOS versions are available. However, the complete installation instructions for the OpenVOS versions are contained in this “Memo to Users”.

These manuals, along with this “Memo to Users” can be downloaded from www.netex.com by selecting the ‘Support’ tab then navigating to the ‘Products, and then selecting the ‘Stratus’ link on the left side of the window, then clicking on ‘Docs’ in the ‘NetEx Requester [H897IPV]’ for the appropriate version.

Support

Support for NetEx/IP and its associated products is available via:

- Telephone (24x7) at (800) 854-0359 (+1 763-694-4350)
- The worldwide web at <http://www.netex.com>
- E-mail to support@netex.com

Notice to the Customer

Comments about this documentation may be submitted via e-mail to support@netex.com or by visiting our website, <http://www.netex.com>. Always include the complete title of the document with your comments.

Information on Network Executive Software's general software support policy (e.g., alternate contact methods, support severity level descriptions, and service status definitions) may be found at <http://www.netex.com/support/software-support-policy>.

Software Modification Policy

Modifications to H897IPV that are not specifically authorized by NESi are prohibited.

Any unauthorized modifications to H897IPV may affect its operation and/or obstruct NESi's ability to diagnose problems and provide corrections. Any work resulting from unauthorized modifications shall be paid by the customer at NESi's then-current support rates and may result in the immediate termination of warranty/support coverage.

New Features

Release 2.1

OpenVOS support on Stratus V Series

This release supports the Stratus V Series platform running the OpenVOS Release 17 operating system.

Installation

Installation Notes

A product release bundle for H897IPV is available as a downloadable distribution. The file has the name “H897IPV_Rel2.1.save.evf.gz”. Download instructions can be obtained by contacting support@netex.com.

The following are prerequisites for installing H897IPV Release 2.1 NetEx/IP Requester:

- A Stratus V Series platform
- OpenVOS Operating System Release 17
- The installer must have modify access to (*master_disk*)>*system* and the directory into which NetEx/IP Requester will be installed.
- Refer to the Software Reference Manual for “H897R (Rel 1.0) DX NETEX Requestor” until the “H897IPV NetEx/IP Requester for Stratus OpenVOS Systems Software Reference Manual” is available.

To communicate with another processor over NetEx/IP, the customer must have:

- The V Series module must be running STREAMS TCP/IP (STCP), a STREAMS-based implementation of the standard TCP/IP family of protocols for Stratus modules.
- The module must be configured with the appropriate Ethernet PCI adapter, cables, and connectors to support STCP. Additionally the STCP hardware requires the installation of appropriate drivers (STREAMS Data-Link Multiplexer communications driver and a lower-level device driver).
- The STCP application API that includes:
 - the socket library that comprises a group of C functions;
 - the STCP object modules that reside in subdirectories of the directory:
(master_disk)>*system*>*stcp*>*object_library*;
 - the STCP include files that reside in subdirectories of the directory:
(master_disk)>*system*>*include_library*
- NetEx/IP on z/OS (H210IPZ) that is licensed for the TNP feature.

Service Notes

This release does not contain source code modules. Therefore any attempt to recompile source code will result in an error and abnormal termination of the script. Since object code modules are provided, -bind requests are still permitted.

Installation Steps

The reader is advised to read through this section in its entirety before performing the installation.

The installation of the NetEx/IP Requester, whether the initial installation or a subsequent update, consists of the following steps:

1. Create the release directory
2. Load the distribution file set
3. Install the distribution file set
4. Verify IP connectivity
5. Create the DXNRL Configuration file
6. Create a DXNRL Environment file
7. Customize NetEx/IP Configuration and Parameters
8. Perform additional verification testing

The steps required to install the H897IPV NetEx/IP Requester product on Stratus V Series are as follows:

Step 1. Create the Release Directory

In order to install H897IPV, the contents of the product distribution file set must be obtained and moved to disk. A release directory must be created first. It is recommended that *>nsc>ntxr* be used as the target directory.

To create a new “ntxr” directory on your system, enter the following commands:

```
create_dir (master_disk)>nsc
change_current_dir (master_disk)>nsc
create_dir ntxr
```

Step 2. Load the Distribution File Set

After creating the release directory, load the contents of the product release bundle to the release directory created in Step 1:

```
change_current_dir (master_disk)>nsc>ntxr
```

Copy the H897IPV product release bundle to the current directory:

```
copy_file [H897IPV_Rel2.1.save.evf.gz from ftp server]
-keep_dates
```

Extract the files from this bundled file to the current directory:

```
unbundle H897IPV_Rel2.1.save.evf.gz -keep_dates
```

Three files will be restored to the current directory:

H897IPV_dist.V	contains the full distribution file set, in OpenVOS save format, for this V Series release of NetEx/IP Requester.
H897IPV_load.cm	command macro that will extract all files from H897IPV_dist.V into the current directory.
install.cm	command macro that will perform commands to finish the installation of files and subdirectories under the “ntxr” directory.

Load the full NetEx/IP Requester file set and associated subdirectories into the current “ntxr” directory by invoking the following command macro:

```
H897IPV_load
```

The full NetEx/IP Requester file set that includes command macros, object files, executable program modules, and sample “dxnrl” configuration files will be restored to the current directory and associated subdirectories.

Step 3. Install the Distribution File Set

After loading the full H897IPV file set into the current “ntxr” directory, NetEx/IP will be fully installed by issuing the following command:

```
install -form
```

This will cause the form as shown in Figure 1 to appear on the terminal screen:

```
-----install-----  
string:  all  
-bind:  no
```

Figure 1. NetEx/IP Installation Screen

string

This parameter defines the scope of the install request relative to the total set of modules that comprise the NetEx/IP Requester product. The allowed values are:

ALL This is the default value. If specified, the installation covers the entire set of NetEx/IP Requester modules (the library modules residing in *>nsc>ntxr>lib*, the test modules residing in *>nsc>ntxr>tests*, and the utility modules residing in *>nsc>ntxr>utils*).

TESTS	If specified, installation tasks are restricted to only test modules residing in <i>>nsc>ntxr>tests</i> .
UTILS	If specified, installation tasks are restricted to only utility modules residing in <i>>nsc>ntxr>utils</i> .
CM	Installation of “cm” modules residing in <i>>nsc>ntxr>config>conf</i> is ignored by the V Series release of this product.
NCTL	Installation of “nctl” modules residing in <i>>nsc>ntxr>config>conf</i> is ignored by the V Series release of this product.

-bind

This parameter is a switch with the following allowed values:

YES

If this value is selected, the object code in the source code directories covered by the scope of the install as determined by the first parameter will be bound into program modules with debugging tables included.

NO

This is the default value. If this value is selected, bind operations will not be performed on any of the object code covered by the scope of the install as determined by the first parameter.

The default parameters in the “install.cm” command macro are recommended for normal installation. Tasks performed by the install macro include the creation of the bin subdirectory (*>nsc>ntxr>bin*), to which the H897IPV utility programs *ntxoper* and *ntxverify* are copied.

The log file *INSTALL_2.n.log* created by the install macro will contain all the messages displayed during the installation. On completion, the following message is displayed:

```
H897IPV program installation complete
```

To invoke the H897IPV utility programs after the install macro has completed, issue the following command at the operating system command level:

```
add_library_path command >nsc>ntxr>bin
```

The above command may also be added to the “start_up.cm” files for various users to take effect during login, or to the “module_start_up.cm” file to take effect during boot.

Step 4. Verify IP connectivity

Since the NetEx/IP Requester uses a TCP connection between the Stratus host and z/OS, verification that IP connectivity exists between the Stratus and z/OS hosts should be performed. Verification should be done using the IP addresses that will be used for the Requester/TNP connection. This can easily be verified with the IP “ping” utility.

Step 5. Create the DXNRL Configuration file

A DXNRL configuration file defines the NetEx/IP Requester configuration. The following describes the contents of this file and where the file may reside.

DXNRL Configuration File Contents

The NetEx/IP Requester uses the DXNRL configuration file to select the H210IPZ TNP component used for a NetEx/IP session. When a NetEx/IP application is started and an SOFFR or SCONNECT request is issued, the NetEx/IP Requester will examine the DXNRL configuration file to select an H210IPZ TNP. The configuration file associates site-determined TNP names with the IP address of the Ethernet interface of the TNP. The IP address is used to establish a TCP connection between the NetEx/IP Requester and the TNP.

An example of a simple configuration file is shown in Figure 2.

```
*
*   Simple DXNRL configuration file example
*
local  myhost
#
netex  tnp1    0.0.0.0      zos5.network.com  tcp  5001
```

Figure 2. DXNRL Configuration File Example

Statements beginning with an “*” or “#” are comments. “local”, “netex”, and “tcp” are keywords, with “local” and “netex” being directive identifiers. Fields may be separated by blanks or tabs.

The “local” directive specifies that the local host name (for NetEx/IP purposes) is “myhost”. This name will be associated with offers (SOFFR) originating from the local host and will allow the TNP to select the correct SFFR if the same application name is offered (SOFFR) by applications on multiple NetEx/IP Requester hosts.

Each “netex” directive identifies a TNP. Note that multiple TNP’s may be defined.

The second field in the “netex” directive specifies a name for the TNP. Note that several TNP components may have the same name, in which case they will be treated as a group.

The third field in the “netex” directive specifies the address of a local network interface which should be used for communication with TNP. This address is used to “bind” the local TCP connection endpoint before connecting. An address of 0.0.0.0 indicates that no explicit bind will be done and the system should select an interface. This field is intended for a configuration where the local host has multiple interfaces to a single network.

The fourth field in the “netex” directive specifies the IP address of the TNP Ethernet interface. It can be specified in either numeric (dotted decimal) or domain name format.

The fifth field in the “netex” directive is optional. It specifies the protocol type, and if specified, it must be specified as “tcp”.

The sixth field in the “netex” directive is optional. It designates the TCP port number used by TNP in the remote NetEx/IP. By default, this port is 5001.

Refer to “>nsc>ntxr>dxnrlrc.base” for an example DXNRL file.

To create a file for editing, issue the following command:

```
copy_file >nsc>ntxr>dxnrlrc.base >nsc>ntxr>dxnrlrc
```

Then edit the “dxnrlrc” file to set up your desired configuration.

DXNRL Configuration File Locations

When a NetEx/IP application using the NetEx/IP Requester is started and the first session call is invoked, the NetEx/IP Requester attempts to locate and validate a DXNRL configuration file. This file is selected as follows:

1. If the OpenVOS “attach_port” command has been issued to establish a logical connection between the port DXNRLRC and a DXNRL Configuration file, then the specified file is used. If this port is detached, the selection process continues with the next step.
2. If the environment variable DXNRLxx (see Step 6) is set, its value is taken to be the path name of the configuration file. If it is not set, the selection process continues with the next step.
3. If a file named “dxnrlrc” is readable in the current directory, it is used. If not, the selection process continues with the next step.
4. If a file named “dxnrlrc” is readable in >nsc>ntxr, it is used. If not, the selection process continues with the next step.
5. If the selection process gets to this point, this is an error. The NetEx/IP Requester displays a message to stdout and exits.

If a file name is selected by the above procedure, and is readable, the DXNRL initialization is complete. Otherwise, the application process is terminated with a fatal error.

Typically, a global configuration file (applicable to all users and all applications) is installed by placing it in the >nsc>ntxr directory and making it world-readable.

DXNRL Environment Options

When the application issues a NetEx/IP SCONN or SOFFR request and a TCP connection to a TNP needs to be established, the NetEx/IP Requester determines a group of TNPs that are candidate servers for this application. By default, all TNPs defined in the configuration file are in the group. The NetEx/IP Requester attempts to establish a TCP connection with each TNP in the group in turn, until a CONNECT is successful. If no connect is successful, the session request is terminated with an error.

Refinement of the TNP selection is possible by using the DXNRLSS environment variable. If the application process’ DXNRLSS environment variable is set to a TNP name, the group of candidate TNPs is composed of all entries from the configuration file (“dxnrlrc”) that match the specified name.

Step 6. Create the DXNRL Environment file

This step enables optional *DXNRLxx* environment variables to be defined as described in the preceding steps; if none are needed, proceed to the next step.

The Stratus OpenVOS operating system does not support the definition of environment variables in the manner supported on such OS platforms as UNIX. Since the NetEx/IP Requester on other platforms supports the use of *DXNRLxx* variables, a mechanism has been implemented to enable H897IPV to support this feature as well.

To define one or more *DXNRLxx* environment variables for H897IPV NetEx/IP applications, a DXNRL environment file must be created. Each environment variable is simply defined as a record in the file. Statements beginning with “*” or “#” are comments. A sample environment file (“dxnrlxx.base”) is provided with H897IPV. To create a file for editing, issue the following commands:

```
change_current_dir >nsc>ntxr
copy_file dxnrlxx.base dxnrlxx
```

When a NetEx/IP application using the NetEx/IP Requester is started, the Requester attempts to locate a DXNRL environment file. This file is selected as follows:

1. If the OpenVOS “attach_port” command has been issued to establish a logical connection between the port *DXNRLxx* and a DXNRL environment file, then the specified file is used. If this port is detached, the selection process continues with the next step.
2. If a file named “dxnrlxx” is readable in the current directory, it is used. If not, the selection process continues with the next step.
3. If a file named “dxnrlxx” is readable in *>nsc>ntxr*, it is used. If not, the selection process continues with the next step.
4. If the selection process gets to this point, the NetEx/IP Requester proceeds with no environment variables defined.

Step 7. Customize NetEx/IP Configuration and Parameters

Refer to the “TNP” section in the “*H210IPZ NetEx/IP for IBM z/OS Operating Systems Release 7.1 Installation Reference Manual*” for a description of the H210IPZ TNP and NCT configuration requirements.

Step 8. Perform additional verification testing

To verify that the NETEX Requester Library and the NETEX TNP are working, make sure that the path *>nsc>ntxr>bin* has been added to the command library (see Step 3 above). Then enter the following command:

```
ntxverify
```

This utility will retrieve the current local NETEX host name and execute a simple intrahost session test. Results similar to the following will be displayed by ntxverify:

```
VERIFY 2.1 local interface and adapter Tue Dec 13 13:44:28 2011
NETEX Requestor: H897IPV Release 2.1
```

```
Get local hostname ... TNPSTRAT
Test session intrahost ... successfully completed 20 loops of 512 bytes
VERIFY 2.1 complete Tue Dec 13 13:44:32 2011
ready 13:44:32
```

Update Summary

Release 2.1

This section describes the updates in Release 2.1 that have been made since the initial Beta Release 2.0.1. Updates in this release include:

- The product was rebuilt after the version level was changed to Release 2.1.
- Shortened the `destination_file` component of the H897IPV release bundle file name in order to ensure that the length of the entire bundle file name does not exceed 32 characters. This was done to prevent a problem copying the H897IPV release bundle to an OpenVOS system on which extended-names support has been disabled.

Known Problems

There are no known problems at the time of this release.