NJE38

(NJE for MVS 3.8)

Installation and Use Guide

Version 2.2.1

15 February 2021

Contents

Introduction to NJE38

NJE38 provides a mechanism to bring NJE data transmission into MVS 3.8. NJE38 can connect to other NJE capable hosts whether they are MVS-, VM-, or DOS-based, and receive file data from these systems and send file data to them.

NJE38 provides connectivity to other systems via binary-synchronous communication (BSC) lines. These lines are emulated in a Hercules environment by the use of the TCPNJE device driver in Hercules. The system on the other end of the connection also uses emulated BSC lines, or for certain operating systems that appeared after MVS 3.8, a choice of emulated BSC lines or NJE over TCP/IP in order to connect back to NJE38.

NJE38 does not provide a JES-based NJE environment like that of later MVS releases. The NJE capability provided by NJE38 is similar to JES-based NJE, but remains a subset of later capabilities. There are a variety of reasons for only having a subset of the possible capabilities and a major one is that system security mechanisms employed by MVS 3.8 do not provide the necessary features to fully implement certain NJE capabilities. Nevertheless, NJE38 provides significant capability and will serve to get data files into and out of MVS 3.8 over an NJE network, including job streams, listings, sequential and partitioned datasets, and more.

NJE38's core networking support is provided partly by an old component of VM/370 RSCS, a communication product associated with that operating system. The old RSCS is freely available in currently circulating Hercules-related VM/370 distributions. RSCS contains a line driver that communicates with remote job entry (RJE) devices. This RJE line driver was considerably modified by Peter Coghlan into a new RSCS line driver that provides NJE capability. The modified line driver uses emulated BSC lines in conjunction with the TCPNJE device driver in Hercules. The TCPNJE device driver was also created by Mr. Coghlan. It was created out of a modified version of the Hercules emulated BSC line driver. As a result of Mr. Coghlan's efforts, many in the VM community are presently enjoying NJE capabilities between their systems across the internet.

NJE38 capitalizes on this prior work to bring NJE into MVS 3.8. A copy of Mr. Coghlan's modified NJE RSCS line driver was brought over to MVS and further modified by myself to operate in an MVS environment. Not all of RSCS was brought into MVS; just the NJE line driver. The rest of NJE38's programming provides the environment, operator control, and user capabilities.

With the exception of the components identified above (namely, the VM/370 RSCS line driver and its subsequent modifications for NJE capability), all of the NJE38 programming is my creation and I am the sole author.

Bob Polmanter
February 2020

Requirements

The following are the requirements that must be met in order to install and operate NJE38:

- A working MVS 3.8 operating system. The MVS 3.8J TK4- system is recommended because it contains a variety of relevant system utilities and security modifications. However, any working MVS 3.8J system with the appropriate BSC lines generated is acceptable.

- A Hercules emulator built with the TCPNJE device support is required. Any Hercules or Hyperion release that runs MVS 3.8J is acceptable, but the emulator must contain the TCPNJE device support. You can build the emulator yourself to include TCPNJE, or you can use pre-built binaries which have TCPNJE already installed. Information on how to build a Hercules emulator to include TCPNJE device support is beyond the scope of this document. To build the emulator yourself, please refer to a related PDF document within the NJE38 distribution containing information about installing TCPNJE support into Hercules. If you wish to use pre-built emulator binaries, see the sidebar on page 7.

- Another NJE capable operating system with which to connect. This can be another MVS 3.8 system running on a different platform, or multiple MVS 3.8 systems running as guests of VM, or the VM system itself via its own RSCS, and so on.

- A minimum of approximately 50 cylinders of available 3350 or 3380 DASD space to contain NJE38 and its components.

- It is assumed that the reader has knowledge of how to add device statements to the Hercules configuration file, or to a live system via the ‘attach’ command, as well as how to edit certain MVS system datasets such as SYS1.PARMLIB and SYS1.PROCLIB.

Installation Steps

Below is a summary of the installation steps required:

1. Load the NJE38 distribution tape to your system
2. Authorize the NJE38 load library with APF
3. Define BSC connections in Hercules to connect with a remote NJE partner.
4. Re-IPL your system for APF authorization to take effect
5. Define the NJE38 spool dataset
6. Format the spool dataset
7. Set up the NJE38 NJE local node and links configuration
8. Add the NJE38 started task procedure to a system PROCLIB
9. Start NJE38

Installing NJE38 for the first time?

Perform all of the installation steps starting on the next page.

Installing NJE38 over an existing NJE38 installation?

Many of the install steps can be skipped, depending on the choices you make. Generally, if you use the same dataset names and intend to use the same network configuration, most of the steps can be skipped. Consult the paragraph and table below for a guide to installing NJE38 over an existing installation.

When upgrading from any prior NJE38 release, you will need to remove or rename your existing NJE38.AUTHLIB and NJE38.SAMPLIB datasets; back them up first if you wish. New versions of these datasets will be created during the upgrade. You do not need to remove the NJE38.NETSPOOL dataset.

When upgrading to NJE38 v2.2.0 or later, you will need to convert your old configuration that specifies your local node name, links, and routes, into the new configuration format. See the sample configuration member in NJE38.SAMPLIB(CONFIG00) and refer to the section NJE38 Configuration File later in this document. Because of the configuration change, the NJE38 start-up proc has also changed.

Consult the table below to determine which steps must be performed and which steps can be skipped.

Step	Consideration	Action
1	Loading the distribution tape must always be performed.	Perform step
2	If you are not changing the NJE38 AUTHLIB dataset name:	Skip step
3	If you are not changing the Hercules BSC connections or addresses:	Skip step
4	Re-ipling system; if step 2 is skipped, then:	Skip step
5	Your existing NJE38 spool dataset is reusable. Skip step unless you wish to change its size.	Skip step
6	Format the spool; perform this step if you performed step 5, otherwise skip.	Decide
7	Run JOB030 then convert your existing configuration file into the new text format.	Perform step
8	The NJE38 PROC has changed in NJE38 v2.2.0; replace the existing NJE38 proc.	Perform step
9	Start NJE38.	Perform step

Detailed Step-by-Step

1. Load the NJE38 distribution tape to your system.

The distribution tape consists of five tape files on a standard label .aws tape image. The first file is a JCL dataset containing sample members and installation JCL. The second file is the NJE38 load library. The third file contains load modules to be placed in SYS2.CMDLIB for TSO use. Files 4 and 5 contain the source materials (optional, not required for use).

To get started, there is a small job stream contained with the **nje38v221.zip** file. The job stream is called **job000.txt**. Use this job to load contents of the **nje38v221.aws** tape image. The JCL is also shown in Appendix B of this document.

Cut and paste the JCL within **job000.txt** into an appropriate editor on your system, and review the JCL for any changes you may wish to make. Specify a specific volume for the NJE38 datasets to reside because you will need to know the volume name in the next step. You may change any dataset names that are created if you wish, but using the default names is strongly recommended for consistency. Submit the job to create the datasets and load the tape.

If for some reason you are unable to view or cut/paste the **job000.txt** file, please see Appendix B.

2. Authorize the NJE38 load library.

Edit the IEAAPFxx member of SYS1.PARMLIB in order to add the dataset name of the NJE38.AUTHLIB to the authorization list. On most systems, this member is named IEAAPF00.

Add the new name anywhere within the list, specifying the dsname and volume of where the dataset was created. Here is an example of a correctly added entry to IEAAPF00:

```
SYS1.VTAMLIB MVSRES,
SYS2.LINKLIB PUB000,
NJE38.AUTHLIB PUB000,
EXH.EXHLIB PUB012,
EXH.ESPLIB PUB012
```

Your IEAAPF00 member may appear differently than that shown above.

3. Define BSC connections in Hercules to connect with a remote NJE partner.

BSC lines are required to use NJE38 in MVS 3.8. These lines must be defined to Hercules so that they will appear as 2703 devices to MVS. The TK4- version of MVS 3.8 has proper BSC lines generated into the system at these addresses:

090-097
690-697

Note that some of these addresses are already defined for use by JES2 as RJE lines. However, if the RJE devices are not activated in JES2, then the lines are available to other tasks in the system. If you use RJE lines, you may wish to find an address from the list above that does not conflict with any JES2 RJE devices.

In order to set up a connection, you will need to provide the following information for each side of the connection: NJE node name, IP address, port number, and BSC line address. In addition, **your Hercules emulator must have the TCPNJE device driver support built into the emulator**. If you wish to build the emulator yourself, please refer to a separate document titled "Installing TCPNJE Support into Hercules.pdf". Alternately, you may wish to use a pre-built version of the Hercules emulator (SDL Hyperion) that contains the TCPNJE driver already installed; please refer to the sidebar at right.

For the purposes of providing an example of setting up the BSC connection to a remote NJE partner, let us make the following assumptions: two MVS 3.8 systems to be connected together via NJE. Let's call them MVSA and MVSB.

System MVSA
NJE node name: MVSA
IP address of the host computer that Hercules is running: 192.168.1.20
Port number assigned to TCPNJE: 1175
BSC line address: 090

System MVSB
NJE node name: MVSB
IP address of the host computer that Hercules is running: 192.168.1.21
Port number assigned to TCPNJE: 1175
BSC line address: 094

Given the above information, here are the Hercules device definitions required; these should be placed in each Hercules configuration file:

Hercules A, where system MVSA will run:

```
0090 tcpnje 2703 lnode=MVSA  rnode=MVSB  lport=1175 rport=1175 rhost=192.168.1.21
```

Hercules B, where system MVSB will run:

```
0094 tcpnje 2703 lnode=MVSB  rnode=MVSA  lport=1175 rport=1175 rhost=192.168.1.20
```

Make the appropriate changes to your Hercules configuration and proceed to the next step.

Obtaining Hercules Emulator Binaries with TCPNJE Already Installed

You may wish to use a pre-built version of the Hercules emulator (SDL Hyperion) that contains the TCPNJE driver already installed. This version of the emulator was produced by TK4- maintainer Jürgen Winkelmann in May 2020 and it is referenced here with his permission. The download contains builds of the emulator for several different host operating system platforms. The download can be found below (copy both lines and paste into your browser):

https://polybox.ethz.ch/index.php/s/pz5Ed1pDPB0PFJz/download

NOTE: If you are using TK4- and before you start installing the package, please read the file within:
README_SDL_Hercules_Hyperion_4.3.9999_build_for_TK4-_Snapshot_1010-05-24.txt.
Follow particularly the instructions in the important notices that are given in this file. Failure to do so may damage your TK4- system.

4. Re-IPL your system for APF authorization to take effect.

You must re-IPL MVS in order for the APF authorization of NJE38 to take effect. If you added BSC connections to Hercules in its configuration file, this would be a good time to restart Hercules as well.

5. Define the NJE38 spool dataset.

JOB010 in the SAMPLIB dataset contains JCL to create the spool dataset. NJE38 requires a VSAM dataset to be used as a spooling mechanism. This is a dataset that is used to hold files queued for transmission to another system, or to hold files that have been received while waiting for a user to retrieve them.

The size of the NJE38 spool dataset in terms of DASD space requires some consideration. At a minimum, the size must be slightly larger than the largest file you expect to transmit or receive. Beyond that however, be aware that there may be other files in the spool awaiting transmission. In a configuration with more than two nodes, there could also be file(s) of unknown size transiting *through* your system on the way to another node. These will occupy spool space until the transmissions are completed.

A good start would be something in the 50 to 100 cylinder range for a spool size. The absolute minimum recommended size would be about 30 cylinders. Create the spool on a 3380 volume (recommended) or a 3350 volume. If the spool size proves insufficient later, you can always delete the dataset and define a larger one.

Review the JCL for changes as well as your desired cylinder size and submit the job.

6. Format the spool dataset

Use JOB020 in the SAMPLIB dataset to format the new spool space. VSAM RRDS datasets require formatting before general use. Formatting also builds the spool's internal directory and free space map.

Review the JCL for dataset name changes and submit the job. The job may run for several seconds especially if the dataset is large or if you have a slower host computer. Review the output for any errors; otherwise a completion code of 0 is success.

If you attempt to format the spool dataset at any time after it has been previously formatted, you will be given a chance to confirm on the system console, as formatting will erase and remove all existing data, if any, in the spool dataset. Only run the formatting program when the NJE38 started task is not active.

7. Set up the NJE38 NJE local node and links configuration

Run **JOB030** to create the configuration dataset and place the example CONFIG00 member into it so that you can customize your own configuration. Edit the CONFIG00 member in the resulting NJE38.CONFIG dataset to match your desired configuration and node name.

An example NJE38 configuration is provided to help you get started preparing your own configuration. This example configuration member is set up to operate with the two-system example configuration from step 3. For comparison, the figure below shows the configurations required for system MVSA and system MVSB.

NJE38 Configuration for MVSA	**NJE38 Configuration for MVSB**
LOCAL MVSA LINK MVSB LINE 090 AUTO YES	LOCAL MVSB LINK MVSA LINE 094 AUTO YES

The configuration member must consist of a single LOCAL entry, which must be first. Then add as many LINK entries as required for the configuration. Follow the link entries with any ROUTE statements to define indirect nodes.   In a two-node configuration like the one above, routes are meaningless and there are none specified.   These statements and the entire configuration are discussed later in this document; refer to the section titled '**NJE38 Configuration File**'.

A more advanced configuration example is discussed in Appendix A.

8. Add the NJE38 started task procedure to a system PROCLIB

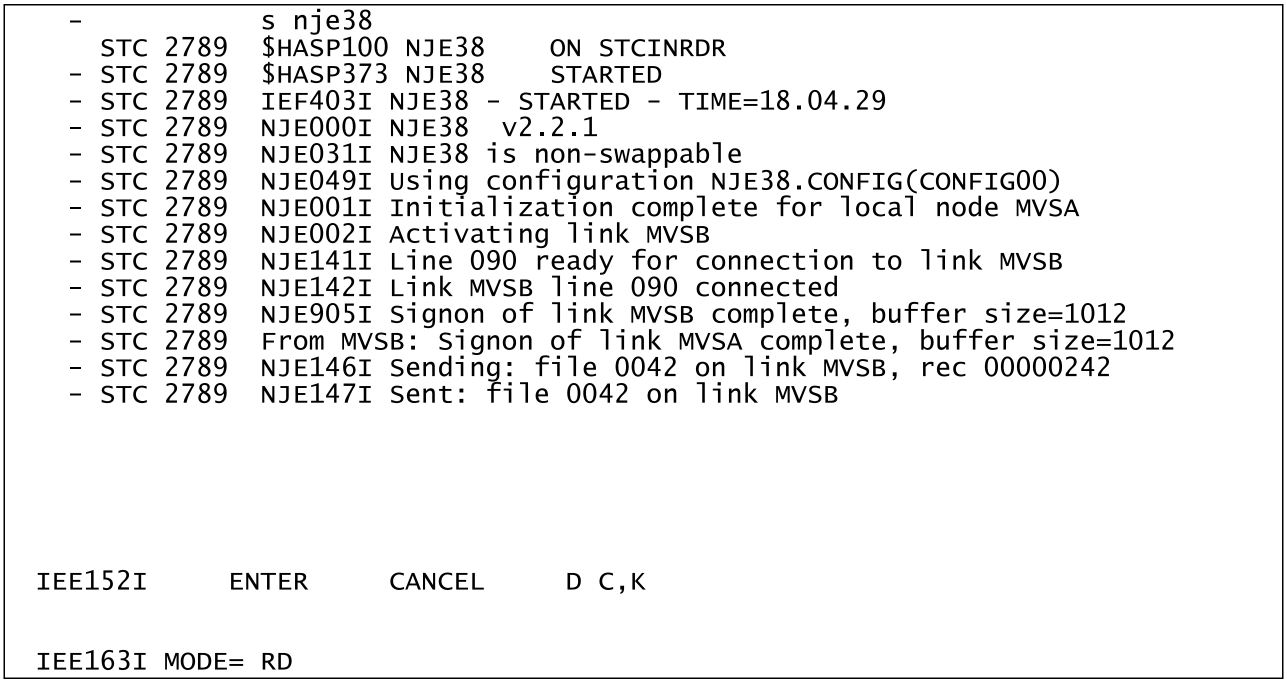
Use member NJE38 of the SAMPLIB dataset for the started-task JCL procedure. Review the procedure JCL for dataset name changes, and then copy the procedure to a system procedure library, such as SYS1.PROCLIB. The recommended member name of the procedure is NJE38.

Then proceed to step 9 on the next page.

9. Start NJE38

NJE38 is started at the MVS system console by an operator START command: S NJE38

NJE38 or equivalent NJE capable software (such as RSCS) should also be started on the other end of the connection. Assuming that MVSB is on the other end as per the examples, below is an example of NJE38 start-up from the console messages, and a file transmission:



```
  -           s nje38
    STC 2789  $HASP100 NJE38    ON STCINRDR
  - STC 2789  $HASP373 NJE38    STARTED
  - STC 2789  IEF403I NJE38 - STARTED - TIME=18.04.29
  - STC 2789  NJE000I NJE38  v2.2.1
  - STC 2789  NJE031I NJE38 is non-swappable
  - STC 2789  NJE049I Using configuration NJE38.CONFIG(CONFIG00)
  - STC 2789  NJE001I Initialization complete for local node MVSA
  - STC 2789  NJE002I Activating link MVSB
  - STC 2789  NJE141I Line 090 ready for connection to link MVSB
  - STC 2789  NJE142I Link MVSB line 090 connected
  - STC 2789  NJE905I Signon of link MVSB complete, buffer size=1012
  - STC 2789  From MVSB: Signon of link MVSA complete, buffer size=1012
  - STC 2789  NJE146I Sending: file 0042 on link MVSB, rec 00000242
  - STC 2789  NJE147I Sent: file 0042 on link MVSB





IEE152I     ENTER     CANCEL     D C,K


IEE163I MODE= RD
```

NJE38 supports a suite of commands to control links and spool files; these are discussed in the Commands and Control chapter.

In order to shut down and stop NJE38, issue a standard MVS operator stop command: P NJE38
NJE38 should terminate all of its links and come down in 2-30 seconds depending on various link status. As a general rule, NEVER cancel NJE38 by an operator cancel command. It is very important that NJE38 come down normally an be allowed to close the spool file properly and update internal pointers.

The installation of NJE38 is completed. However, please review the Appendix C: Security Considerations section of this document for any important operational information.

Commands and Control

NJE38 presently supports several commands to control its operation and to provide information. All of the commands can be issued from the MVS operator console. Subject to authorization restrictions, some or all of the commands may be issued by TSO users from the local node, or from remote users at other nodes.

NJE38 Supported Commands

The commands on the next two pages are presently supported by NJE38.

Command	**AUTH**	**Function**
D TIME		Display the local time and date
D FILES		Display the files in the NJE38 spool
D *filenum*		Display information about a specific NJE38 spool file
D NODES		Display the node connection status
D ROUTES		Display the routing table
D AUTH	Yes	Display the authorized user table
D ACTIVE		Display the active address spaces
D *nodeid*		Display the link and/or route status of the specified *nodeid*
C *filenum*	Yes	Cancel (purge) spool file number *filenum*
C *filenum1-filenum2*	Yes	Cancel (purge) all spool files from *filenum1* through *filenum2*
RESET *filenum*,DEST=*nodeid.userid* E *filenum*,D=*nodeid.userid*	Yes	Reset *filenum* to a new destination *nodeid* and *userid*
CPQ CPLEVEL		Display system level
CMD *nodeid command-text*		Issue a command to a remote node
MSG *nodeid userid message-text* M *nodeid userid message-text*		Send a message to *userid* at *nodeid*

NJE38 Supported Commands (continued)

Command	AUTH	Function
LINK *nodeid* LINE *cuu* BUFF *size* AUTO *val*	Yes	Add a new link for *nodeid* using line *cuu*, optional BUFF/AUTO
LINK *nodeid* OFF	Yes	Remove existing link for *nodeid*
ROUTE *nodeid* TO *link-id*	Yes	Add a new route for *nodeid* via *link-id*
ROUTE *nodeid* OFF	Yes	Remove route for *nodeid* from routing table
AUTH *userid* AT *nodeid*	Yes	Add a userid at the specified node to the authorization table
AUTH *userid* AT *nodeid* OFF	Yes	Delete a userid at the specified node from the authorization table
S *nodeid*	Yes	Start the link to another node
P *nodeid*	Yes	Stop (drain) the link to another node

Commands that appear with a "Yes" in the AUTH column require that the userid/nodeid of the issuer be present in the authorization table (see NJE38 Configuration). The system operator console of the local node is always authorized for every command.

Whether commands are issued from the system console or from a TSO user, the command-text format is the same. Commands are always issued to the local node unless the command text is prefixed by 'CMD'. Operator messages can be issued to remote users only and the full nodeid/userid specification is required for the MSG command. The MSG command may be abbreviated as M.

A few commands may be abbreviated. Some commands have RSCS-like equivalents for compatibility with VM users. The available abbreviations and compatibility commands are shown below:

Command	Abbreviation	Compatible command
D TIME	D T	CPQ Time
D FILES	D F	Q Files
D ACTIVE	D A	CPQ Names
D NODES	*None*	Q SYStem
D ROUTES	*None*	Q SYStem ROutes
MSG	M	Msg

Special note on the "C" Cancel command: Users whether local or remote can always cancel their own files, even if they are not explicitly authorized by the authorization table. To be considered a file owner, a user (whether local or remote) must either be the sender of the file (the origin user and node), or must be the recipient of the file (destination user and node). If the user does not meet the owner criteria, then that user must be explicitly authorized in order to cancel spool files.

Using the "RESET" command: Use the RESET or "E" command to alter the destination of a file in the NJE38 spool. The sender of the file may have inadvertently sent the file to the wrong userid, or perhaps you wish to reroute the file onward to another node. You may also use a '*' character in place of the local node name; for example to change the destination userid on the local node: e 37,d=*.HERC03

Using the LINK command: Use the LINK command to add a new link to another host via a specified line. The nodeid and line cuu are required. The buffer size if omitted will default to 1012. The automatic start specification if not specified defaults to NO. Once a new link is added, it can be started immediately using the "S nodeid" command, if the line is varied online and defined to the Hercules emulator.

You may also define two or more nodeids to use the same line cuu address. Only one can be started at a time of course. This may be useful in certain situations where remote hosts are changing.

Use the OFF parameter with LINK nodeid to remove an existing link. Before the link will be removed in must be stopped and in an inactive state.

Issuing Commands from the MVS System Console

Commands entered from a system console must be entered via the MVS operator command MODIFY. The short form of the MODIFY command is F. The format of the command is:

MODIFY *procname*,text
  or
F *procname*,text

Where *procname* is the name of the NJE38 started task procedure, and *text* is the command that you wish NJE38 to process. The command itself and the text are not case sensitive. Blanks in the command text are acceptable and no apostrophes or quotes should be entered. Responses to a command entered on the console will be displayed on the console.

Examples of possible operator commands:

F NJE38,D NODES	Display node status at the local node
F NJE38,CMD MVSB D NODES	Display node status of the MVSB remote node
F NJE38,S MVSB	Start the local node's link to MVSB
F NJE38,M MVSB HERC01 Hello from MVSA!	Send a message to user HERC01 at node MVSB
F NJE38,ROUTE MVSC TO MVSB	Add a routing table entry to MVSC via link MVSB
F NJE38,ROUTE MVSD OFF	Nullify (turn off) the route to MVSD
F NJE38,AUTH OP AT MVSB	Make MVSB operator console an authorized user
F NJE38,C 48-53	Delete NJE38 spool files 48 through 53
F NJE38,RESET 117,D=*.HERC01	Reset the destination of file 117 to user HERC01 on the local node
F NJE38,E 62,DEST=MVSB.HERC06	Reset the destination of file 62 to node MVSB user HERC06
F NJE38,LINK CEARN LINE 097 BUFF 800	Dynamically define a new link named CEARN using line address 097

Issuing Commands and Messages from TSO Terminals

TSO users can issue commands and receive responses at their terminal via the TSO command 'NJE38'. The NJE38 command can be issued from the TSO ready prompt or from inside RPF via the TSO command line option 6. ISPF users have two choices: command line option 6 or via the TSO prefix from other panels. TSO users also have the option to use an asterisk character in place of the local node name where a node name is required.

When using the NJE38 command, if you receive a message that says "The NJE38 command is not APF-authorized" please refer to Appendix C: Security Considerations in this document.

The format of the TSO NJE38 command is:

NJE38 *command-text*

Examples of the TSO NJE38 command usage:

NJE38 D NODES	Display node status at the local node
NJE38 CMD * D NODES	Same as above; another way of specifying
NJE38 CMD MVSB D NODES	Display node status of the MVSB remote node
NJE38 M * HERC02 please resubmit your job	Send a message to user HERC02 on the local node
NJE38 M MVSB HERC01 Hello !	Send a message to user HERC01 at node MVSB
NJE38 C 71	Delete spool file 71*
NJE38 ROUTE MVSF TO MVSB ALT MVSC	Add a dynamic route **

* Issuing user must be the file sender or file recipient; or must be explicitly authorized.
** Issuing user must be explicity authorized to issue the ROUTE command.

For users that prefer a consistent command format (for example, to use NJE38 inside a CLIST), you may use the CMD command to send commands to the local node as well, thereby allowing all use of the NJE38 command to have the same format: e.g., "NJE38 CMD *nodeid command-text"* regardless of whether the command is destined for the local or remote node. Optionally, specify '*' for the local nodeid to avoid the need to know what the local nodeid is.

Users should note that when commands are issued there is a slight delay while the command is transmitted, executed by the destination node, and the results transmitted back to your terminal. Therefore, the user should wait a few moments and press Enter at the TSO terminal in order to receive the results.

RSCS Users

Remote RSCS users can issue a command to a remote NJE38 via the usual method familiar to RSCS NJE users:

CP SMSG RSCS CMD *nodeid text*
CP SMSG RSCS MSG *nodeid userid text*

For example:

```
CP SMSG RSCS CMD MVSA  D NODES
CP SMSG RSCS MSG MVSA  HERC02 I sent your file
```

Example Commands and Responses

The system operator wants to query the connection status for NJE38:

```
-           f nje38,d nodes
- STC 2800  NJE012I  Node status
- STC 2800  Name     Status    Type     Addr
- STC 2800  MVSB     Connect   NJE       090
- STC 2800  MVSC     Inactive  NJE       091
```

A number of files are queued for destination MVSC but the link is currently inactive. A file has arrived from node MVSB destined for MVSA (this node), but the user ‘HERC02’ on the MVSA system has not yet retrieved it. Display the file status:

```
-           f nje38,d f
- STC 2800  NJE014I  File status
- STC 2800  File  Origin   Origin    Dest     Dest
- STC 2800   ID   Node     Userid    Node     Userid    CL  Records
- STC 2800  0035  MVSB     HERC02    MVSC     HERC01    A       193
- STC 2800  0036  MVSB     HERC02    MVSC     HERC01    A       193
- STC 2800  0037  VM1      MAINT     MVSC     HERC01    A     2,351
- STC 2800  0038  MVSA     HERC01    MVSC     HERC01    A        22
- STC 2800  0040  MVSB     HERC01    MVSA     HERC02    K        77
- STC 2800  Spool 04% full
```

The MVSA operator realizes files 0035 and 0036 are duplicates, so he deletes one of them.

```
-           f nje38,c 36
- STC 2800  NJE015I  File(0036) purged
```

A TSO user at node MVSB issues a NJE38 style ‘D F’ command to see if the HERC02 user at MVSA has retrieved his file yet:

```
NJE38 CMD MVSA D F
***
From MVSA: NJE014I  File status
From MVSA: File  Origin   Origin    Dest     Dest
From MVSA: ID    Node     Userid    Node     Userid    CL  Records
From MVSA: 0035  MVSB     HERC02    MVSC     HERC01    A       193
From MVSA: 0037  VM1      MAINT     MVSC     HERC01    A     2,351
From MVSA: 0038  MVSA     HERC01    MVSC     HERC01    A        22
From MVSA: 0040  MVSB     HERC01    MVSA     HERC02    K        77
From MVSA: Spool 03% full
```

The MVS operator wants to stop the link to node MVSB:

```
-           f nje38,p mvsb
- STC 2800  NJE570I Link MVSB now set to deactivate
- STC 2800  NJE143I Link MVSB line 090 disconnected
- STC 2800  NJE010I Line 090 is drained
```

NJE38 Configuration File

The NJE38 configuration file specifies the node name for the system as well as all of your remote connections and routing information used to reach nodes beyond your connections, if any.

The configuration file format has changed from a series of macros that are assembled and linked, into a text-based configuration file that can be a sequential file or a member of a PDS (recommended). In this way, multiple configurations can be kept in the PDS and you can choose which one is to be used by NJE38 when it is started. No longer are configuration assembly jobs required when the configuration is changed.

The configuration file presently consists of four main statement types: LOCAL, LINK, ROUTE, and AUTH. These define the parameters for the local node, the remote links or nodes, routes to yet other nodes, and a list of NJE38 authorized users. The configuration file must consist of as many 80-byte fixed length records as are needed to define your configuration. It is recommended that the configuration be kept in a PDS so that multiple configurations can be stored there.

All of the main statement types must begin in column 1 of each record. You may also have comment records throughout the configuration; comment records must have an asterisk character in column 1.

Configuration Example / Overview

Below illustrates an example configuration file:

```
*
*-- Local parameters
LOCAL MVSA     DEFUSER HERC01
*
*-- Physical links to remote nodes
*
LINK   MVSB    LINE 091    AUTO NO     BUFF 1012
LINK   VM1     LINE 092    AUTO YES    BUFF 800
*
*-- Routes to indirect nodes
*
*     Indirect    via
*     Node     TO Link
*     -------- -- --------
ROUTE MVSD     TO MVSB
ROUTE MVSC     TO VM1
*
*-- Authorized users
*
*     Userid   AT Node
*     -------- -- --------
AUTH  HERC01   AT MVSA
AUTH  HERC01   AT MVSB
AUTH  OP       AT MVSB
AUTH  CMSUSER  AT VM1
```

The example configuration defines connections that might look like this:

MVSC-----VM1--------**MVSA**------MVSB------MVSD

LOCAL Statement
The configuration member must consist of a single LOCAL entry, which must be the first statement encountered.

The format of the LOCAL statement is:

LOCAL *nodeid* DEFUSER *userid*

The LOCAL statement defines the node name of your NJE38 and as it will be known by other host which you connect with. Choose the *nodeid* carefully, because it is rather painful to have every remote connection change their configurations as well to accommodate a later nodeid name change. All node names in this and other statements in the configuration must be valid NJE node names, which consist of 1-8 uppercase characters containing the characters A-Z, 0-9. Again, the LOCAL statement must be first in the configuration, but may be preceded by comments.

The optional DEFUSER parameter defines the default userid for NJE38 use if and only if your MVS system does not have any security product enabled. If DEFUSER is not specified, the userid will default to HERC01. **If your MVS 3.8 system has no security product installed or enabled (such as RAKF) please refer to Appendix C: Security Considerations.**

LINK Statement
After the LOCAL statement, specify one or more LINK statements to define your remote connections. The remote connections can be other MVS systems, or VM systems, or other types of NJE communications capable hosts.

The format of the LINK statement is:

LINK *remotenodeid* LINE *cuu* BUFF=*size* AUTO=*yes|no*

Where:

r*emotenodeid* is the node name of the system on the remote end of the connection.

LINE *cuu* is the device address of the MVS defined 2703 communication line. This address corresponds to the same device address of a Hercules TCPNJE configuration statement also defining this connection. Exactly three hexadecimal digits are required.

The remaining parameters on the LINK statement are optional and may be omitted:

BUFF *size* is the maximum buffer size that NJE38 is to use in negotiations with the remote node during connection. The allowable value for *size* is 300 to 1012. The default value if BUFF is not coded is 1012. The default value is also the recommended value. Sizes up to 4020 may be specified, however values above 1012 are not specifically supported. Please refer to the Limitations section for more information on larger BUFF sizes.

AUTO defines whether or not the link is automatically activated by NJE38 upon start up. Of course, the other end of the connection must also be started (via AUTO or by command) before there can be a successful connection. Specify a YES or NO value. Default if not coded: NO
Links may also be dynamically added or removed from the configuration while NJE38 is active. This can be done at the operator console via the F NJE38,LINK command. Link alterations done dynamically take effect immediately. However, remember to update the permanent configuration file, as dynamic modifications will fall off if NJE38 is restarted.

ROUTE Statement
Routes are required to reach nodes that may be beyond any links you have direct connections with. Specify as many routes as are needed or desired to enable paths to other nodes. These routed nodes will also need routes defined in *their configurations* that described the routing back to your system to be effective. The ROUTE statement(s) must follow the last LINK statement. If you have no routes, remove or comment out any ROUTE statements.

Routes may also be dynamically added, updated, or removed from the routing table that is defined by the ROUTE statements. This can be done at the operator console via the F NJE38,ROUTE command. Routing alterations done dynamically take effect immediately. However, remember to update the permanent configuration file, as dynamic modifications will fall off if NJE38 is restarted.

The format of a ROUTE statement is:

ROUTE *remotenodeid* TO *link-name* ALT *alt-link-name*

Where:

remotenodeid is the name of a distant remote node (that is, not a node directly specified by your LINK statements). *Remotenodeid* may contain an asterisk character to define a wildcard route (see Wildcard Routes, below).

link-name is the name of a node that is a LINK within your configuration which is along the path to *remotenoteid.*

alt-link-name is the optional name of an alternate node that is a LINK within your configuration which provides an alternative path to *remotenoteid.* Up to three *alt-link-names* may be specified.

As an example, assume a network that looks like this one:
MVSA—MVSB—MVSC—MVSD

System MVSA only has a direct connection to system MVSB via a LINK statement. Routes will be required in order to send commands or files to nodes MVSC and MVSD. Correspondingly, systems MVSC and MVSD must have routes back to MVSA in their configurations. System MVSB also needs routes to reach MVSD. These example routes help to illustrate:

System MVSA	**System MVSB**	**System MVSC**	**System MVSD**
ROUTE MVSC TO MVSB ROUTE MVSD TO MVSB	ROUTE MVSD TO MVSC	ROUTE MVSA TO MVSB	ROUTE MVSA TO MVSC ROUTE MVSB TO MVSC

For connection configurations that could provide alternative paths, let’s assume that in addition to the example network above, MVSA also defines a direct connection to MVSC:

```
MVSA—MVSB—MVSC—MVSD
  |             |
  ---------------
```

This means that network traffic leaving MVSA could get to any of the other nodes via MVSB or MVSC. The route statements in MVSA could therefore be used define these alternate paths:

System MVSA
ROUTE MVSD TO MVSC ALT MVSB ROUTE MVSC TO MVSB ROUTE MVSB TO MVSC

The ROUTE MVSD provides two paths to MVSD. The alternate MVSB path would only be used if the more direct link from MVSA to MVSC is down. The ROUTE MVSC provides a different path to MVSC via MVSB in the event that the direct link from MVSA to MVSC is down. And finally, the ROUTE MVSB provides another path to MVSB in the event that the direct link from MVSA to MVSB is down. The routing in the other nodes would be similar to these, in an opposite form.

Use extreme caution in defining alternate path routes. It can become very complex very quickly and it is easy for one of the nodes to have an error in the routing specifications that can cause routing loops. This causes a file or a message to travel from a node to another node, and be returned if that is what the routing specifies. Upon return, the node sends it back again, because that’s what the routing says to do. The only way to stop a routing loop is to stop the link by command, and then purge the file from the NJE38 spool in whichever node the file resides at the time the link was stopped. Then correct the routing specifications allowing the loop.

Using Wildcard Routes

Wildcard routes allow you to specify groups of similar remote node names in a single route statement. For example, use this example network:

```
MVSA—MVSB—NYC1—NYC3
            |     |
          NYC2—NYC4
```

Rather than coding four route statements for the NYC nodes, the MVSA node’s configuration could simply code:
ROUTE NYC* TO MVSB

Node MVSB might code:
ROUTE NYC* TO NYC1

When a node, such as MVSA in the above example, has only a single path to the outside, all routes *must* go through MVSB since that is the only defined path. Node MVSA might use a single ROUTE statement to cover every possibility:
ROUTE * TO MVSB

Wildcard Routing Special Considerations

1. When a file or message is to be transmitted to a remote node via a route, non-wildcard names are selected over wildcard names if they match exactly. For example, assume these statements:

   ```
   ROUTE NYC* TO MVSB
   ROUTE NYC8 TO MVSX
   ```

   If a user sends a file to node NYC8, the direct *explicit* route via MVSX will be chosen even though the wildcard route of NYC* would qualify the name NYC8 for selection.

2. When a file or message is sent to a named link node (i.e., a direct connection) but the link is inactive, any wildcard routes that would qualify that link name will <u>not</u> be selected. For example:

   ```
   LINK  ABC   LINE  097
   ROUTE  A*  TO MVSB
   ```

   When a file is sent to ABC but ABC is down, the file will not be routed to MVSB even though the wildcard of A* certainly qualifies the name ABC as a suitable routing candidate. The file will remain in the spool until link ABC becomes active.

   If you want link ABC to have special routing, you must code a separate explicit route defining that path:

   ```
   ROUTE ABC TO MVSB
   ROUTE A* TO MVSB
   ```

   This rule only applies to names that are also link names in your node's configuration.

3. Use caution when defining wildcard routes in multiple nodes. It is very easy to cause routing loops.
4. When using wildcard routes, be aware that mis-typed node names could cause a file to be sent somewhere that you didn't intend. For example, a file wants to transmit a file to node ABC but as he is typing in the node name into the TRANSMIT command, he mistypes the name as ABV. This would be covered by the wildcard route example A* above. If node ABV actually exists in MVSB, it will be sent there even though that was not the intention. If node ABV doesn't exist, then the file will just sit in MVSB's spool until someone at that node can reset it to another destination or purge it.

AUTH Statement

Optional AUTH statements may be coded in order to authorize remote users at other nodes, or TSO users at the local node, so that they will be able to issue restricted NJE38 commands. Code as many AUTH statements as needed to define the users you wish to be authorized.

The format of the AUTH statement is:

AUTH   *userid   AT   nodeid*

Where:

userid is the userid of a local or remote user you wish to have authorized status.

nodeid is the node name of the authorized user. If *userid* specifies a TSO userid for the local node, specify the local node name for *nodeid.*

AUTH Statement Special Considerations

- The MVS operator console of the local node is always authorized for all commands; no AUTH statement is required.
- MVS operator consoles belonging to remote nodes must explicitly be authorized if you wish them to have command authority on your local node. To specify an operator console, code *userid* as OP at the desired node. For example:

  ```
  AUTH   OP    AT  MVSC
  ```

- If you wish to authorize a remote VM system console, specify that system's operator userid in the *userid* parameter of AUTH.
- If you wish to authorize a remote RSCS virtual machine console, you must code $RSCS for *userid* followed by AT and the node name of that remote system. For example, to authorize commands issued to your local node from the RSCS virtual machine on remote node VMSYS1, code:

  ```
  AUTH  $RSCS   AT  VMSYS1
  ```

- Otherwise, code all other AUTH statements as needed with actual userids and node names.

Authorizations may also be dynamically added or removed from the authorization table that is defined by the AUTH statements. This can be done at the operator console via the F NJE38,AUTH command. Authorizations done dynamically take effect immediately. However, remember to update the permanent configuration file, as dynamic modifications will fall off if NJE38 is restarted.

Transmitting and Receiving Files with NJE38

NJE38 is not JES2 based, nor is it like RSCS running in a VM environment. But its internal data transmission mechanisms are compatible with NJE on other operating systems thereby allowing NJE38 to connect with these systems.

If instead of NJE38 you were using a JES2-based NJE connection, or a RSCS-based NJE connection, whatever data file you wish to be transmitted must first be written to the JES2 spool, or the VM spool, respectively, before it can be transmitted. This is what the TSO TRANSMIT, or VM PRINT/PUNCH/SENDFILE commands do. Once the file is fully contained within the spool, the file will be transmitted by JES2 or RSCS as the case may be, over NJE. Correspondingly, inbound files are placed in the respective spool as they are received and made available to the destination user once the receiving transmission has completed.

With NJE38, the concept is the same. The file must be placed into the NJE38 spool before it can be transmitted. Files being received will be placed in the NJE38 spool awaiting retrieval by the destination user. The mechanisms used to get files into and out of the spool are the NJE38 TRANSMIT and RECEIVE commands from TSO, or alternately, simple batch jobs.

The TRANSMIT and RECEIVE commands distributed with NJE38 are TSO commands. For those familiar with more modern MVS-based operating systems, these commands will operate very similarly. These commands convert data being transmitted to/from a format called NETDATA, which is a standard documented format. This allows your MVS 3.8 system to send datasets to and receive files from more advanced operating systems that already use the NETDATA format, as well as to other MVS 3.8 systems using NJE38. The TRANSMIT and RECEIVE commands and their use are described in the next section.

The batch utilities NJ38XMIT and NJ38RECV perform the basic function of placing files into the NJE38 spool for transmission, or retrieving them from the spool after receipt, respectively. These utilities can be paired with other utilities not associated with NJE38 but nevertheless useful such as XMIT370 and RECV370. These utilities are discussed later in the section titled Using the NJ38XMIT and NJ38RECV Batch Utilities.

The NJE38 TRANSMIT and RECEIVE commands must run APF-authorized. **If you receive a message that says "The command is not APF-authorized" please refer to Appendix C: Security Considerations in this document.**

TRANSMIT Command

Use the NJE38 TRANSMIT command within a TSO environment in order to prepare and transmit files via NJE. Files that can be transmitted are non-VSAM sequential and partitioned datasets.

The TRANSMIT command format is similar to that used on modern TSO systems although it has fewer parameters. The command may also be specified as XMIT.

TRANSMIT *node.userid* DAtaset(*input.dsname*) OUTdataset(*output.dsname*) Volser(*volume*)
Unit(*unitname*)
PDS | SEQuential
Quiet

Where:

node.userid – is the optional destination node and userid of where to transmit the file over NJE. If not specified, OUTDATASET must be specified. If node.userid is specified, it must be the first parameter after TRANSMIT.

DATASET – required. Specifies the name of the dataset that will be transmitted. Your TSO userid will be prefixed to the beginning of the dataset name unless you place the name in apostrophes. A member name may also be specified in parenthesis immediately following the dataset name, for example: DATASET(my.pds(mem1))

OUTDATASET – optionally specifies the name of a dataset that the prepared transmission should be written to in lieu of sending it via NJE. If the named dataset exists it will be overwritten. If it does not exist, it will be created. OUTDATASET may also specify the name of an existing PDS with a member name in parenthesis. Your userid will be prefixed to the beginning of the dataset name unless you place the name in apostrophes.

If you specify OUTDATASET, the *node.userid* values are placed in the encoded NETDATA file, but no NJE transmission occurs.

If you do not specify OUTDATASET, then TRANSMIT always attempts to send the file via NJE.

The dataset or member named by OUTDATASET will always contain NETDATA formatted records. These records can be processed by RECEIVE using INDATASET, or later sent for transmission using the NJ38XMIT batch utility.

VOLSER – optionally specify the name of a volume where you wish the target OUTDATASET to be created.

If you do not specify VOLSER, an available volume marked as public will be selected automatically.

UNIT – optionally specify the unit name of where you wish the target OUTDATASET to be created. If UNIT is specified but VOLSER is not specified, the volume selected will be a public or storage volume belonging the unit name’s grouping as defined to the system. In most cases, UNIT should be allowed to default.

If you do not specify UNIT, the default is SYSDA.

PDS – If specified, indicates that the member specified with DATASET is to be transmitted using IEBCOPY unload thereby transmitting the member’s user directory information along with the member.   PDS has no meaning if a member name was not specified in DATASET.

<u>**SEQUENTIAL**</u> – DEFAULT. Indicates that the member specified with DATASET is to be transmitted as a sequential file. No directory information is needed or preserved. SEQUENTIAL only has meaning if a member name is specified. SEQUENTIAL must be used if transmitting a PDS member to a VM system.

QUIET – If specified, indicates that informational messages will be suppressed. A transmit operation would therefore issue no messages. However, any errors encountered will still issue error messages. The QUIET option may be useful for TRANSMIT within TSO CLISTs.

Example TRANSMIT sessions

<u>Example 1</u>
A TSO user is logged on to userid HERC01. The user wishes to transmit a small PDS named HERC01.COBOL.PDS to user FRED at node MVSB.

```
transmit mvsb.fred da(cobol.pds)
NJE38 TRANSMIT v2.0.0

                                         IEBCOPY MESSAGES AND CONTROL
STATEMENTS
                               PAGE 0001
 IEB167I  FOLLOWING MEMBER(S) UNLOADED FROM INPUT DATA SET REFERENCED BY
SYS00003 -
 IEB154I  MYCOB    HAS BEEN SUCCESSFULLY UNLOADED
 IEB154I  CPYBLK   HAS BEEN SUCCESSFULLY UNLOADED
 IEB154I  WRTEX12  HAS BEEN SUCCESSFULLY UNLOADED
 IEB147I  END OF JOB -00 WAS HIGHEST SEVERITY CODE

File successfully queued to MVSB.FRED
READY
```

<u>Example 2</u>
A TSO user is logged on to userid BOB. He wishes to prepare a NETDATA formatted file and store it into another dataset for later disposition. No transmission is performed. The input dataset is a sequential file named BOB.PLI.PROGA. The formatted file is to be stored on DASD locally, in a dataset called SYS2.XMIT.SOURCE(PROGA).   User BOB uses the TRANSMIT command with the OUTDATASET parameter to prepare the file:

```
xmit da(pli.proga) out(‘sys2.xmit.source(proga)’)
NJE38 TRANSMIT v2.0.0

File successfully written to ‘SYS2.XMIT.SOURCE’
READY
```

The TRANSMIT command has a number of limitations; most of them are similar to the limitations in the modern TRANSMIT counterpart. It is suggested that the Limitations and Known Issues section be reviewed.

TRANSMIT always outputs NETDATA formatted records. If you must transmit data that is not in NETDATA format, you must use the NJ38XMIT utility.

RECEIVE Command

Use the NJE38 RECEIVE command within a TSO environment in order to retrieve files that were transmitted via NJE. Files that can be retrieved are those containing NETDATA, as well as simple 80/133-byte punch/print records. RECEIVE always creates a new cataloged dataset to contain the received results; RECEIVE will not receive into an existing dataset.

The RECEIVE command format is similar to that used on modern TSO systems. All parameters are optional; in its simplest form you can simply specify 'RECEIVE' with no parameters.

RECEIVE *nnn* DAtaset(*target.dsname*) Volser(*volser*) Unit(*unitname*) DIR(*dblks*) INdataset(*input.dsname*)
Purge | NOPurge
Prompt | NOPRompt
Quiet

Where:

nnn – is the optional NJE38 spool file number of the file you wish to retrieve. If no file number is specified, the next available file destined for your userid is retrieved. If a file number is specified, it must be the first parameter after RECEIVE. The available spool file numbers can be obtained by issuing a F NJE38,D FILES operator command, or by using the TSO NJE38 command to issue D FILES.

DATASET – optional. Specifies the name of the target dataset that will contain the retrieved file. Your userid will be prefixed to the beginning of the dataset name unless you place the name in apostrophes.

If you do not specify DATASET, the target dataset name is derived from the original dsname when it was transmitted. The first qualifier of the original dsname is changed to your userid.

Generally, it is recommended that you do not specify the DATASET parameters and let RECEIVE choose the target dataset name. During RECEIVE processing, the original dataset name will be displayed and you are given a chance to specify a different target dsname if desired.

VOLSER – optionally specify the name of a volume where you wish the target dataset to be created.

If you do not specify VOLSER, an available volume marked as public will be selected automatically.

UNIT – optionally specify the unit name of where you wish the target dataset to be created. If UNIT is specified but VOLSER is not specified, the volume selected will be a public or storage volume belonging the unit name's grouping as defined to the system. In most cases, UNIT should be allowed to default.
If you do not specify UNIT, the default is SYSDA.

DIR – optionally specify the number of directory blocks to be allocated to the target dataset when the target is partitioned. In most cases this parameter need not be specified and should not be specified. Its primary purpose is to be used when the incoming file being received was originally a PDSE data library dataset. PDSE datasets do not have directory blocks in the traditional sense and the number of directory blocks that would be required for a standard PDS are not accurately contained in the incoming file's attributes. Thus, RECEIVE cannot determine the number of directory blocks required for incoming PDSE datasets. This parameter provides a means to specify directory block quantity. Should you ever receive a dataset from a modern operating system and the IEBCOPY portion of the receive operation fails due to a lack of sufficient directory blocks, delete the target dataset that was created and re-RECEIVE the incoming file again with the DIR( ) option specifying an appropriate 'best guess' number of directory blocks based on what you know about the incoming data.

If DIR is specified and the incoming files's attributes do contain directory block quantity information, the specified value will override the incoming file's attribute.

If you do not specify DIR, the default is to use what the incoming file's attributes specify and in most cases this is the recommendation.

INDATASET – optionally specifies the name of an existing sequential dataset containing NETDATA records that you wish to retrieve. INDATASET may also specify the name of an existing PDS with a member name in parenthesis. Your userid will be prefixed to the beginning of the dataset name unless you place the name in apostrophes.

If you specify INDATASET, the file id number *nnn* is ignored, if present.

If you do not specify INDATASET, then RECEIVE always attempts to retrieve from the NJE38 spool.

The dataset or member named by INDATASET <u>must</u> contain NETDATA formatted records. Ordinary 80/133-byte punch/print records are not accepted from INDATASET and will abort the RECEIVE process.

<u>PURGE</u> – If specified, indicates the spool file is to be purged from the spool upon successful retrieval into a target dataset. **This is the default**. PURGE has no meaning if INDATASET is specified.

NOPURGE – if specified, indicates the spool file is to be retained in the spool upon successful retrieval into a target dataset. NOPURGE has no meaning if INDATASET is specified.

<u>PROMPT</u> – Indicates that RECEIVE is to issue a prompt to the user for additional parameters. The user may optionally choose a different target dataset name or volume, or abort the RECEIVE process. **This is the default** and is the normal operation of RECEIVE.

NOPROMPT – if specified, indicates that RECEIVE is to issue no prompts to the user for additional parameters. The user should specify all needed parameters on the command line invocation. Any errors encountered will abort the receive process and there will be no opportunity to enter replacement parameters.

QUIET – If specified, indicates that informational messages will be suppressed. A successful receive operation would therefore issue no messages. However, any errors encountered will still issue error messages. QUIET also forces NOPROMPT; therefore, the use of QUIET means there will be no user prompts. The QUIET option may be useful for RECEIVE within TSO CLISTs.

RECEIVE Prompts

During RECEIVE processing, RECEIVE will display the dataset name from the original file. You will be prompted to enter any changes. In most cases, you can simply press enter knowing that RECEIVE will create the target dataset using the same dsname as the original except that the first qualifier will be replaced with your own userid. If this is not acceptable, specify DATASET on the RECEIVE command to specifically name your target dataset, or wait until the prompt. You can enter the DATASET parameter at the prompt to specify a target dsname instead of using the command line, or to change the target dsname again.

If the target dsname already exists, you will be given a second prompt (see example 1 below) and another chance to specify a unique target dsname.

The valid RECEIVE parameters that can be specified on the prompt are DATASET, VOLSER, UNIT, DIR, PURGE, and END.

On any prompt, specifying END means abort RECEIVE and no action is taken. The file is not retrieved, the target is not created, and nothing is purged.

On any prompt and only when receiving from the NJE38 spool, specifying PURGE means simply delete the spool file and exit RECEIVE with no further action. Use PURGE at the prompt when you have decided you do not wish to receive the file after all.

Example RECEIVE sessions

Example 1

A TSO user is logged on to userid HERC01. He wishes to receive the next file queued for him. The user knows nothing specific about the file other than it was sent to him. He uses the RECEIVE command to retrieve the file; shown is the sequence of events:

```
receive
NJE38 RECEIVE v2.0.0

Receiving 'A.TT.LISTING' from VMSYS1(CMSUSER)
Enter receive parameters or 'PURGE' or 'END' +
presses enter
Dataset 'HERC01.TT.LISTING' exists
Receiving 'A.TT.LISTING' from VMSYS1(CMSUSER)
Enter receive parameters or 'PURGE' or 'END' +
da(tt2.listing)

Receive into 'HERC01.TT2.LISTING' successful
File(0059) purged from NJE38 spool
READY
```

Example 2

A TSO user is logged on to userid BOB. He wishes to receive a NETDATA formatted file previously created by the XMIT370 utility. The formatted file is stored on DASD locally, in a dataset called SYS2.XMIT.SOURCE(PROG5MAC). User BOB uses the RECEIVE command with the INDATASET parameter to retrieve the file; shown in this sequence of events:

```
RECEIVE IN(‘SYS2.XMIT.SOURCE(PROG5MAC)’)
NJE38 RECEIVE v2.0.0

Receiving 'SAM.PROG5.MACROS' from MVSA(SAM)
Enter receive parameters or ‘END’ +
presses enter

                                 IEBCOPY MESSAGES AND CONTROL STATEMENTS
 IEB167I  FOLLOWING MEMBER(S)  LOADED FROM INPUT DATA SET -
 IEB154I  MYDCB    HAS BEEN SUCCESSFULLY  LOADED
 IEB154I  GETQ     HAS BEEN SUCCESSFULLY  LOADED
 IEB154I  PUTQ     HAS BEEN SUCCESSFULLY  LOADED
 IEB144I  THERE ARE 0000003 UNUSED TRACKS IN OUTPUT DATA SET
 IEB149I  THERE ARE 0000005  UNUSED DIRECTORY BLOCKS IN OUTPUT DIRECTORY
 IEB147I  END OF JOB -00 WAS HIGHEST SEVERITY CODE

 Receive into 'BOB.PROG5.MACROS' successful
 READY
```

Other RECEIVE Considerations

The RECEIVE command can receive an incoming file from the INDATASET or from the NJE38 spool. If INDATASET is used, the contents of the specified dataset must contain NETDATA formatted records. If you are receiving a spool file, the incoming file can be NETDATA formatted or just plain data, (e.g., a “flat” file).

When receiving a plain “flat” file from the NJE38 spool, the data will consist of 80- or 132-byte records per NJE convention. If the data is 80-byte records, it is received and written to the target dataset as is. If the data is 132-byte records (such as a PRINT-style listing from a VM system), it will always be received and written to the target dataset with LRECL=133 RECFM=FBA. The addition of the ASA carriage control character thus increases the target record length by 1 compared to the incoming file. If you must receive a PRINT-type file with another form of carriage control or record length sizing control, you must use the NJ38RECV utility to receive the spool file. These kinds of files are denoted in a NJE38 D *filenum* command output as ‘Type: PUNCH’ or ‘Type: PRINT’.

Since incoming files can come from various operating systems, it is possible that the original file was blocked with a block size larger than the DASD devices on your system can accommodate. For example, receiving a file blocked 23,400 would not fit on a 3350 device with a track size of 19069 bytes. RECEIVE will attempt to locate a volume with a larger track capacity to use for the target dataset. However, if you do not have a larger track size volume online or marked as public, or if you specify a particular volume to use that happens to be a smaller track size than needed, RECEIVE will issue error messages and abort the receive process. Your choices at that point are either to not receive the file, or to bring a larger track size DASD device online to your system and attempt to re-receive the file.

The RECEIVE command has a lengthy number of restrictions and limitations; it is recommended that you review them in the Limitations and Known Issues section of this document.

Using the TRANSMIT and RECEIVE Commands in Batch TSO

The TRANSMIT and RECEIVE commands may be used in a TSO batch environment. The parameters and options are identical.

For the TRANSMIT command in batch mode, no userid prefix is available. Therefore, always specify the fully qualified dataset name in any use of DATASET or OUTDATASET.

An example of the batch JCL and TRANSMIT command is shown below.

```
//HERC01T  JOB 1,'TRANSMIT IN BATCH',CLASS=A,MSGCLASS=X
//C1       EXEC PGM=IKJEFT01
//SYSPRINT DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*
//SYSTSIN  DD *
 XMIT DA(HERC01.MACLIB(SAMPMAC)) OUT(HERC01.NETSAVE) V(PUB002)
/*
```

For the RECEIVE command in batch mode, RECEIVE will issue no prompts. This means that if the target dataset exists then RECEIVE will issue a message and simply terminate. To help avoid this, specify an explicit target dataset when using batch mode and if needed delete any prior existing dataset of that name in a previous job step. The example below shows an explicit target dataset specified on the command.

No userid prefix is available in batch mode. Therefore, always specify the fully qualified dataset name in any use of DATASET or INDATASET.

An example of the batch JCL and RECEIVE command is shown below.

```
//HERC01R  JOB 1,'RECEIVE IN BATCH',CLASS=A,MSGCLASS=X
//C1       EXEC PGM=IKJEFT01
//SYSPRINT DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*
//SYSTSIN  DD *
 RECEIVE IN(HERC01.NETDATA(SAMPMAC)) DA(HERC01.MACLIB)
/*
```

Batch TRANSMIT and RECEIVE commands always use the TSO userid of the submitter for security purposes during their operation. If your MVS 3.8 system has no security package installed, the default userid specified in the NJE38 CONFIG member is used. **Please refer to Appendix C: Security Considerations in this document for more information about the default userid.**

The TRANSMIT and RECEIVE commands have a lengthy number of restrictions and limitations; it is recommended that you review them in the Limitations and Known Issues section of this document.

Using the NJ38XMIT and NJ38RECV Batch Utilities

NOTE: all utilities and utility jobs mentioned below are located in the NJE38 SAMPLIB dataset.

If you simply want to use the batch NJ38XMIT and NJ38RECV utilities to operate on simple card image data and listings, or want to transmit more complex data such as entire partitioned datasets, you can use these two utilities for at least part of the transmit and receive process in lieu of the TSO-based TRANSMIT and RECEIVE commands. **If your MVS 3.8 system has no security product installed or enabled (such as RAKF) please also refer to Appendix C: Security Considerations.**

NJ38XMIT is responsible for reading the data you wish to have transmitted and to write it out to the NJE38 spool, and then to notify NJE38 that a new spool file is ready for transmission. NJ38XMIT must be executed from an APF-authorized library or it will be unable to notify NJE38 about the new file.

NJ38XMIT needs to know the node id and userid of where the file's destination. This information is placed in the PARM field of the EXEC JCL statement. Specify the node id followed by the userid, separated by a blank, period, or comma. The PARM field must be from 3 to 17 bytes in size and otherwise follow the rules for a JCL PARM parameter. These examples are valid PARM fields:

```
//S1     EXEC  PGM=NJ38XMIT,PARM=(MVSA,HERC01)
//S2     EXEC  PGM=NJ38XMIT,PARM='MVSB HERC02'
//S3     EXEC  PGM=NJ38XMIT,PARM='VM1.MAINT'
```

NJ38RECV is responsible for retrieving data from the spool that has arrived at the destination node. The user that is expecting the file data should prepare a job to execute NJ38RECV to retrieve the file data from the spool and place the file data into a dataset as designated by the JCL. NJ38RECV supports optional parameters in the JCL PARM parameter but does not require them; they are detailed below.

The following parameters options may be specified in the JCL PARM field for NJ38RECV:

NJ38RECV Parameter	**Use**
PURGE	Causes the spool file to be deleted after successful retrieval. This is the default.
NOPURGE	Indicates that the spool file is not to be deleted after a successful retrieval. Removal of the file will require a NJE38 C *filenum* command, or a second NJ38RECV job using the PURGE option.
FILE=*filenum*	Specifies the spool file id number of the desired file to retrieve. Use this option to receive specific files; the user executing NJ38RECV must be the file owner. The default action if FILE= is not specified is to receive the next available file owned by the user.

Example:

```
//S4       EXEC PGM=NJ38RECV,PARM='FILE=47,NOPURGE'
```

The spool file id numbers may be obtained from a NJE38 D FILES command.

Liberal use of the NOPURGE option is recommended, especially if you are unsure about the format of the data being received, or if the type of JCL to be used to get it in the right form is uncertain. In this way, the file can be re-received if the format proves to be incorrect. If it is correct, simply Cancel the spool file to remove it.

The following DD statements are required by the two utilities:

DDNAME	Use
STEPLIB	Identifies the AUTHLIB dataset containing NJE38 programs.
SYSPRINT	SYSOUT; for messages pertaining to the execution success or failure of the utility.
SYSUT1	NJ38XMIT only; identifies the *input* dataset that will be written to the NJE38 spool and queued for transmission.
SYSUT2	NJ38RECV only; identifies the *output* dataset that will contain the file data received by NJE38 over NJE.

For SYSUT1 datasets, the DCB attributes used are those of the dataset, unless overridden in JCL. For SYSUT2, DCB attributes should be specified in the JCL if creating the dataset as NEW. If SYSUT2 refers to an existing dataset, the DCB attributes will be those of the dataset if no overriding values are coded in JCL.

For both utilities, the dataset name of the NJE38 spool dataset is obtained from NJE38. However, if NJE38 is not active, then the dataset name cannot be obtained. Start NJE38 (links do not necessarily have to be started; rather NJE38 just needs to be active) before attempting to use these utilities.

Sample Jobs for Transmitting and Receiving Files

The following job streams are provided in the NJE38 SAMPLIB dataset;

Desired Function	Sample Job Member Name
Transmit basic 80-byte records (e.g., a source code member, a copybook, a macro), or 121/132/133-byte records (e.g., a listing or printout)	XMIT
Retrieve basic 80-byte records (e.g., a source code member, a copybook, a macro), or 121/132/133-byte records (e.g., a listing or printout)	RECV
Transmit a sequential dataset *	XMITSEQL
Retrieve a sequential dataset previously transmitted by XMITSEQL	RECVSEQL
Transmit a PDS * Source and LOADLIB examples provided	XMITPDS1 XMITPDS2
Retrieve a PDS previously transmitted by XMITPDS Source and LOADLIB examples provided	RECVPDS1 RECVPDS2

* The dataset format (LRECL/RECFM) must be supported by the XMIT370/RECV370 utilities.

Once a file is written to the spool by NJ38XMIT, it is queued for transmission. If the link to the destination node, or along a path to the destination node is connected, NJE38 will begin transmission immediately if the link is available or as soon as other traffic completes. Once a file is received at the destination, the user at that end can query the spool (using the NJE38 ‘D F’ command) to see if the file has arrived and/or view the system console messages to see that it has arrived. At that time, the user can execute a NJ38RECV job to retrieve the transmitted file data and place it into a user-designated dataset. See also the section “Using the ‘D *filenum*’ Command to Assist with Dataset Retrieval”.

Transmitting Entire Datasets Using the Batch Utilities

In order to transmit sequential datasets with record sizes that do not readily fit into the 80 or 132 byte NJE standard, you must use a utility that will convert the file into a format suitable for NJE transmission.

The NJE38 TSO-based TRANSMIT and RECEIVE commands are recommended. If you prefer a batch solution, existing utilities to do this are distributed with MVS 3.8 TK4-: XMIT370 and RECV370.

XMIT370 and RECV370 do not transmit or receive anything, despite that inference from their names. Likewise, TSO TRANSMIT and RECEIVE, or VM SENDFILE or RECEIVE also do not transmit or receive anything either. The actual transmission is left up to JES2 or RSCS or NJE38. Instead, XMIT370 is used in a batch job to convert a dataset to 80-byte records and a subsequent step in the same (or different) job uses NJ38XMIT to write it to the NJE38 spool. Then NJE38 will queue the file for transmission. Upon receipt, NJ38RECV is first used to retrieve the 80-byte file records from the NJE38 spool, and a subsequent step in the same (or different) job uses RECV370 to un-convert the received data back to its original form. At the successful conclusion of this process, you have transmitted a sequential or partitioned dataset to another node!

Sample jobs XMITSEQL, RECVSEQL, and XMITPDS, RECVPDS provide working examples. Each combines a conversion step with an NJE38 utility to write to or retrieve from the spool.

Using the 'D *filenum*' Command to Assist with Dataset Retrieval

(Note: this section was written before the introduction of the TSO TRANSMIT and RECEIVE commands. It is retained here for explanatory purposes, but the RECEIVE command automates all of what is described below).

If you are unable to use the TSO RECEIVE command, one of the problems with receiving files over NJE is not necessarily knowing what kind of file or data was received, as well as what kind of dataset should be allocated in order to contain the data after retrieval. A display command is provided to allow a query by a specific spool file number in order to obtain more information.

Typically, after a file is received the user would issue a "D FILES" command in order to get a list of file(s) and obtain the spool file id number. Then the user would follow that with a "D *filenum*" command where *filenum* is the desired spool file id number. This command will return information about the file that can help you determine what kind of data it is, what format, and its size. It can also help you determine what kind of JCL to use in order to retrieve the file properly.

Below is an example of a D *filenum* command output:

```
          F NJE38,d 145
STC 3974  NJE026I  File status for node MVSA
STC 3974  File  Origin   Origin    Dest     Dest
STC 3974   ID   Node     Userid    Node     Userid    CL  Records
STC 3974  0145  MVSB     HERC01    MVSA     HERC01    A    20,777
STC 3974  Tagged name: HERC01 OUTPUT   Type: NETDATA
STC 3974  Attributes: 6160/80/FB/PO  DIRBLKS=5   File size 3514 KB
STC 3974  Origin DSN=HERC01.COBOL.SRC
```

It can be determined that from this display that the sender (HERC01 at MVSB) transmitted a dataset named HERC01.COBOL.SRC. The dataset has the DCB attributes BLKSIZE=6160, LRECL=80, RECFM=FB, DSORG=PO. The dataset requires 5 PDS directory blocks. The NJE internal data type is NETDATA, which means that the utility RECV370 must be used to decode the NETDATA after it is retrieved from the spool by utility NJ38RECV. Sample job RECVPDS2 in the NJE38.SAMPLIB dataset would be ideal for this example. This sample job retrieves the file from the NJE38 spool first, and then passes it to RECV370 for decoding the NETDATA and creating the resulting dataset.

In addition, it can be calculated from the Display command information how much space should be allocated to the dataset in your JCL. As an example, let us assume you wish to create the dataset to hold this file on a 3350 device. This device type has a maximum track size of 19069 bytes. The block size of the transmitted dataset is 6160 and three blocks of 6160 will fit on each track. The 'file size' in the display output is reported to be approximately 3514 kilobytes. To calculate the space requirement, multiply the file size shown by 1024 to obtain bytes: 3514 * 1024 = 3598336 bytes. Divide that result by the block size of 6160 to determine how many total blocks are required: 3598336 / 6160 = 584.1 blocks, so 585 blocks would be needed. As previously mentioned, three blocks can fit on a single track so to compute how many tracks are required divide the number of blocks by three: 585 / 3 = 195 tracks.

Please note – and this is very important – that the file size shown in the Display output is approximate as documented by NETDATA format documentation. Therefore, it would be wise to allocate a bit of additional space to your dataset. The additional space could be specified in the primary allocation, or in secondary allocation as you choose. Therefore, given the above information and computed example, a recommended space allocation for the new dataset could be: SPACE=(TRK,(195,10,5)). If one wanted a cylinder allocation, divide the number of tracks needed by 30 in the case of a 3350 device and round up: 195 / 30 = 6.5 cylinders and round up to the next cylinder. Thus the space allocation could be SPACE=(CYL,(7,1,5)). And finally, for those interested in less math, you could use block allocation and simply state the number of 6160-sized blocks in the SPACE parameter. We earlier calculated that 585 blocks were required, so the space allocation using this method could be SPACE=(6160,(585,20,5)). With block allocation one would not need to be as concerned about the device type and its space characteristics.

For load library type datasets, the LRECL is not displayed as it is not used for these kinds of datasets. The Attributes display of a LOADLIB type dataset could be something like: 6144/U/PO. The value shown (6144 in this case) is always the block size when RECFM=U.

You may also receive other types of data files. A file could be a 'flat' punch file or print file. A common example of these would be if a VM user were to transmit to you a file from the CMS PUNCH command; for example:

SPOOL PUNCH TO RSCS
TAG DEV PUNCH MVSA HERC01
PUNCH COBTEST COBOL

When you receive that file, this is what might be shown by Display *filenum:*

```
F NJE38,d 136
NJE026I  File status for node MVSA
File  Origin   Origin    Dest     Dest
 ID   Node     Userid    Node     Userid    CL  Records
0136  VMR6A    CMSUSER   MVSA     HERC01    A     9,852
Tagged name: COBTEST COBOL   Type: PUNCH
Attributes: 80/F/PS
```

For these types of files, much less is known about their content and attributes because it is not in NETDATA format. It just a plain punch file. However, enough is shown to determine how to retrieve it and what kind of dataset and size it should be. From the attributes, we can see it is

LRECL=80,RECFM=F,DSORG=PS. Blocksize is mostly meaningless for VM files. Therefore, you will need to choose a blocksize when allocating your MVS-based dataset to hold this file. Let's do the calculations for a second example.

The record length of this file is 80 bytes and we know there are 9852 records. The file size is therefore approximately 788160 bytes. We want to put this on a 3330 device. A good blocksize on a 3330 that is a multiple of 80 bytes is 12960. We can fit one block of 12960 per 3330 track. Dividing that blocksize of 12960 into 788160 bytes gives us: 788160 / 12960 = 60.8 tracks, or when rounded up 61 tracks. Let's add a little extra space in the secondary allocation. Therefore, the JCL to allocate a dataset to contain this file could be:

```
//SYSUT2  DD DSN=HERC01.COBTEST.COBOL,DISP=(,CATLG),UNIT=3330,SPACE=(TRK,(69,3)),
//  DCB=(BLKSIZE=12960,LRECL=80,RECFM=FB)
```

Sample job RECV in NJE38.SAMPLIB would be ideal for this example. In the case of PRINT-type files, choose a RECFM in the JCL to be FBA or FBM based on what kind of carriage control you wish in your final dataset.

One other type of data exists that you may encounter: a file sent from a more modern VM system. If the VM user sends you the file via the PUNCH or PRINT command, then the file will be similar to the last example above and you will be able to retrieve this file. However, if the VM user sends it using the CMS SENDFILE command, there will be problems with compatibility. The RECV370 utility that is part of TK4- is not able to process modern NETDATA formats (this is detailed more in the next section). The NJE38 TSO-based RECEIVE command can successfully process this file. If you encounter such a file, it would look something like this example:

```
F NJE38,d 142
NJE026I  File status for node MVSA
File  Origin   Origin    Dest     Dest
 ID   Node     Userid    Node     Userid    CL  Records
0142  VMFAKE   TEST1     MVSA     HERC01    A       284
Tagged name: TT LISTING   Type: NETDATA
Attributes: 121/F/PS    File size 25 KB
```

The lack of a blocksize and dataset name are conspicuous. Another clue is the tagged name; it sure looks it could be a VM file. Use the TSO RECEIVE command to retrieve this file.

Important notes about using XMIT370 and RECV370

Describing how to use these utilities and documenting everything about them is beyond the scope of this document. However, it is important to understand some basic things about these utilities that will make things so much easier if you know about them, so it will be mentioned here.

These utilities are rather finicky with regard to DCB attributes. Generally, it does not appear to use the attributes from the dataset DSCB; they must be specified in JCL in order to make the utility work correctly. Specifying the DSORG parameter is extremely important. If you do not have the DCB attributes specified or if they are specified incorrectly, the utilities can fail with odd looking abend codes and no descriptive messages indicating the problem.

Like NJ38XMIT and NJ38RECV, which use SYSUT1 as input and SYSUT2 as output respectively, XMIT370 and RECV370 do the same. But there is confusion when processing PDS datasets. In this case, the other SYSUTx DDNAME is used to point to a work file. In other words, XMIT370 uses SYSUT1 as the input file, so SYSUT2 is used as a work file. Correspondingly, RECV370 uses SYSUT2 as the output file, so SYSUT1 now has to be used as the work file. The sample jobs provided have it set up correctly.

When sequential files are to be converted with XMIT370 / RECV370, the SYSUT1 dataset is used for input (for XMIT370) and for output (for RECV370). This is contrary to XMIT370/RECV370's own documentation. These inconsistencies led to having separate example jobs in the SAMPLIB dataset specific to sequential datasets.

Caution is advised, and care should be taken with all DDNAMEs and DCB parameters with XMIT370/RECV370 JCL.

The NJE38 RECEIVE command may or may not be able to receive files created by the XMIT370 utility. Inconsistencies in the NETDATA records generated by XMIT370 can cause RECEIVE to exit with an error message, or could cause an abend. Some effort has been expended to detect these and avoid an abend and try to exit gracefully; however, the inconsistencies can be unpredictable. As a general rule however, XMIT370 output files created from PDS datasets have been more reliable than those created from sequential files. Further, XMIT370 is not always placing the dataset name key into the NETDATA records, leading to a missing dataset name at receive time.

Finally, the conversion format used by XMIT370/RECV370 (called NETDATA format), is not fully compatible with the NETDATA format that is used on later versions of VM or MVS (this is documented by the XMIT370/RECV370 developer). This author is not sure if it is a bug in these utilities, or a subtle change in the NETDATA format over the years. This means that a modern TSO TRANSMIT and RECEIVE, or a modern VM SENDFILE and RECEIVE may not be able to process data from these utilities. (Note: NJE38 TSO RECEIVE can process files created by XMIT370). Nevertheless, the XMIT370/RECV370 utilities will serve well to pass datasets between MVS 3.8 systems in a batch environment.

Miscellaneous Tips

In no particular order:

- NJE38 is designed to stay active all of the time, like JES2 and VTAM and TSO. Once satisfied with its operation, consider placing it in an appropriate list to be automatically started after IPL.

- If you are sending data to a VM/370 system, you MUST use the batch job XMIT because the NETDATA format used by the NJE38 TRANSMIT command is not compatible with that version of VM. The job XMIT will send data to VM as a PRT or PUN file in the destination user's virtual reader (which type depends on the original dataset's LRECL and RECFM). However, data coming in from VM/370 can be retrieved via the TSO RECEIVE command or the batch job RECV. The NJE38 TRANSMIT and RECEIVE commands are supported with more advanced VM-based operating systems (subject to any limitations in the next section) because they are NETDATA capable.

- Protect your spool data! It is always prudent to run IDCAMS VERIFY against the spool dataset if NJE38 or any of its utilities should abend. See job VERIFY in the SAMPLIB dataset.

- NJE38 can connect to other NJE compatible software that does not necessarily operate under a typical mainframe-based operating system. Linux-based Funetnje is one example.

- Operator use of the 'MODIFY NJE38' console command can be enhanced by assigning a partially completed command image to a console PF Key. The operator would only need to press the designated PF Key and then enter the text of the NJE38 command without the MODIFY portion each time. For example, to assign PF8 to this purpose, enter this command at the system console:

  ```
  K N,PFK=(08,CMD='f nje38,_'),CON=Y
  ```

  By placing an underscore character as shown above, the cursor will be positioned for the operator to type in the command text directly after pressing the PF key.

- Use caution when specifying alternate routes or if you are using wildcard routes. Incorrect routing statements can cause looping where a file is transmitted to a remote node and that node sends it right back, and the process repeats.

Limitations and Known Issues

At present, the following are limitations in NJE38 (in no particular order). Some or all of these limitations may be lifted in future releases.

Operator Commands Limitations

- There are no commands at present to dynamically add, delete or alter links in the configuration.
- Local and remote users may issue link START and STOP commands to an NJE38 node (if authorized). However, any status messages generated by the command will still be directed to the system console of the NJE38 node that the command was directed to. For now, the issuing user could issue the D NODES command to determine the link status.
- TSO users issuing local or remote message traffic via the TSO NJE38 command please note: when using NJE38 from inside RPF (e.g., TSO commands option 6), the message text is uppercased by RPF and not by the NJE38 command. This can be demonstrated by issuing NJE38 from a TSO READY prompt instead.
- Due to size limitations in the use of the MVS SEND command, messages sent to TSO users whether they are from other local or remote users or are responses to commands, are limited to 104 characters per message line. Messages that exceed this size are truncated and there is no indication to the user that this occurred.
- NJE38 does not support multi-level CMD commands. For example, RSCS users can route commands through an intermediate node by issuing 'cmd nodea cmd nodeb Q SYS). NJE38 will not properly pass the command onwards to intermediate nodes in this fashion.

Spool Limitations

- No mechanism exists to change the order of files queued for transmission or the order they will be retrieved by the user once received. However, due to the way the spool's directory is managed, the order of the files will almost always be first-in first-out (FIFO) ordering. In addition, spool files can be retrieved out of order by using the FILE= parameter for batch utility NJ38RECV, or the *filenum* parameter on the TSO RECEIVE command.
- JCL jobstreams received via NJE are never submitted for execution; likewise, no job output is ever automatically transmitted.
- No files received via NJE are ever printed or punched directly on a JES2 device, even if that was the sender's intention (for example a VM user PRINTing a listing to come off on MVS's printer). If you want to print the output on receipt, run NJ38RECV and direct SYSUT2 DD to SYSOUT as appropriate.
- Spool classes are not used by NJE38. There is no difference to NJE38 in a file received as class A vs. any other class. Files originated at the local NJE38 node are always class A. Files received from elsewhere that have a specific spool class will retain that spool class when passed through an NJE38 node on their way somewhere else.
- The record counts shown by various NJE38 issued messages or NJE38 Display commands are approximate. They represent the number of physical records on the NJE38 spool and not necessarily the number of actual records in the actual file. However, the number shown versus the actual are usually very close and within a few percent.

TRANSMIT and RECEIVE Commands Limitations

NJE38 TRANSMIT and RECEIVE operate similarly to the corresponding commands in modern operating systems. However, there are restrictions and limitations. Many of these restrictions are also present in modern usage.

- Operate on cataloged datasets only.
- Password protected datasets are not supported.
- Only one dataset can be transmitted or received per command invocation. Multiple-file formatted NETDATA transmissions are not supported.
- Creation date, Last reference date, etc., are not used or supported (ignored).
- Generation datasets (GDG) are not supported.
- ISAM and BDAM datasets not supported.
- Datasets with keys are not supported.
- Multi-volume datasets not supported.
- Receiving incoming files originated as PDSE Program Library datasets are not supported.
- Receiving incoming files originated as PDSE Data Library datasets are supported and RECEIVE will place the data into a standard MVS 3.8 PDS. However, you must specify the RECEIVE DIR option to indicate how many directory blocks should be allocated to the standard PDS, as the incoming file does not contain this information.
- Extended format sequential datasets from modern operating systems are not supported by MVS 3.8 and therefore by NJE38 and its utilities and commands.
- VSAM datasets are not supported. However, a VSAM dataset can be REPRO-ed into a sequential file and then be transmitted or received.
- ACKnowledgements that are used by modern TSO are not supported; NJE38 TRANSMIT will not send or request an acknowledgement. NJE38 RECEIVE will tolerate (ignore) the receipt of an acknowledgement but will not send any acknowledgement of its own.
- Messages encoded within a NETDATA transmission are not supported or generated by NJE38 TRANSMIT. For incoming files, messages encoded within a NETDATA transmission from a modern TSO system are not supported and detection of message traffic using the INMTERM key within the NETDATA will abort the RECEIVE process with an error message. Do not use the MSG option with TRANSMIT from a modern operating system if your file is destined for NJE38.
- When using TRANSMIT with OUTDATASET, please note that if the dataset specified already exists, **it will be overwritten with no opportunity to confirm or select another dataset**.
- When using TRANSMIT with OUTDATASET and specifying an *existing* dataset, it could run short of space if it is too small to contain the incoming file (abend SB37-04). If this occurs, rerun TRANSMIT specifying a new OUTDATASET. New datasets created by OUTDATASET will calculate the space required based in the incoming dataset; this is not possible if the dataset already exists.
- The dataset named by INDATASET, if specified, must contain NETDATA formatted records of 80-byte fixed records only. Non-NETDATA formatted records are not retrievable with RECEIVE using INDATASET.
- If PURGE or END are specified on a RECEIVE prompt, any other parameters that may have been specified also at the prompt are ignored and the PURGE or END action takes immediate effect.
- Datasets with EXPDT are not supported; the dataset will be received but the EXPDT attribute is not set. If you wish, you may separately recopy the received dataset into a new dataset with EXPDT.
- When receiving files created by XMIT370 the origin node and userid may not be known because XMIT370 does not have this information available in order to encode it in the NETDATA.

- BATCH mode support issues no prompts; therefore, specify all necessary parameters on the RECEIVE command line including the target dataset name when running RECEIVE in batch. If a target dataset exists, RECEIVE will terminate.
- In BATCH mode no userid prefix is available; therefore, specify fully qualified dataset names in the DATASET and/or INDATASET/OUTDATASET parameters.
- Some advanced operating system features may make no sense to MVS 3.8 and in some cases may prevent the data received by NJE38 RECEIVE to not be processed properly. For example, you could transmit certain types of PDSE datasets from a modern system, but it will be a standard PDS dataset when received to MVS 3.8. Enciphered data is another example that is not supported.
- The maximum record size allowed by NJE38 TRANSMIT and RECEIVE are the same as the MVS 3.8 limit: 32760 bytes. However, VM systems allow records of 65535 bytes; any file containing record sizes above 32760 bytes will not be able to be processed by NJE38 RECEIVE. As another example, it would make no sense to transmit a PDS to a VM system.
- RECEIVE does not process VMARC-formatted data. If such data is received, it will be placed into the target dataset in its raw form like any other data. The data is still VMARC valid if later sent back to another VM system.
- RECEIVE will terminate if the block size of the incoming dataset cannot fit on the hardware track size of your DASD devices. This can occur if you have no larger track size devices (such as a 3380) online or marked public, or if you specifically named a different and smaller track size volume in the VOLSER parameter. Bring larger DASD online and specify one of those volumes and then re-receive the file. You can move and re-block the target dataset afterward, if desired.
- Receiving a 132-byte 'PRINT' type file from the NJE38 spool (that is, a plain file containing no NETDATA encoding) will always be written to the target dataset as LRECL=133,RECFM=FBA.
- RECEIVE always attempts to create a new cataloged dataset to contain the received data. If a dataset name exists, RECEIVE will prompt you for changes to the dataset name (unless you have turned off prompts with the NOPROMPT or QUIET option). If you are unable to supply a unique dataset name, RECEIVE will terminate without receiving the data. The original incoming file remains available.
- Problems receiving files created by XMIT370: RECEIVE may or may not be able to receive files created by the XMIT370 utility, which is not part of NJE38. Inconsistencies in the NETDATA records generated by XMIT370 can cause RECEIVE to exit with an error message, or could cause an abend. Some effort has been expended to detect these and avoid an abend and try to exit gracefully; however, the inconsistencies can be unpredictable. As a general rule however, XMIT370 output files created from PDS datasets have been more reliable than those created from sequential files. Further, XMIT370 is not always placing the dataset name key into the NETDATA records, leading to a missing dataset name at receive time.

Other Limitations and Known Issues

- The console operator cannot send messages to local TSO users using the Modify NJE38 command. However, the operator can use the MVS SEND operator command for this purpose.
- Some error messages resulting from an action by a local or remote user are not returned to the originating user but instead show on the MVS console. Some minor error messages may not appear at all. For example, sending a command or message to a routed remote link that is inactive will not necessarily result in a link-not-active message being returned to the sender.
- Buffer sizes above 1012 bytes for line connections to remote nodes are not supported. Nevertheless, the BUFF parameter on the NJE38 configuration file's LINK statement allows sizes above the supported 1012 value, up to 4020. Sizes above 1012 should be considered "experimental". In most cases, large sizes work fine, so try it at your own risk. However, any

problems, abends, aborted transmissions, loops, or other oddities encountered while using NJE38 with one or more links defined with a size over 1012 will not be supported.

- Once a link connection is established, the buffer size negotiated with the remote node becomes the new maximum buffer size for the link for the life of that link task.
- Because NJE38 is based on the RSCS Version 1 line driver, multi-streaming is not supported and only a single stream is supported per line. This means that if you are connecting NJE38 to more advanced RSCS or JES2 nodes, you must specify within their connection parameters to use only one stream for any lines directly connecting to an NJE38 node. For RSCS, this generally means specifying the STreams=1 and TA=0 parameters for the line. For JES2, this usually means limiting the number of job and sysout transmitters and receivers to 1. Consult the appropriate documentation for your advanced systems for the exact parameter settings. Failure to limit the streams can result in loops or stalled transmissions within NJE38.

Providing Debugging Information

Should any NJE38 utility programs NJ38XMIT, NJ38RECV, or NJEFMT abend, a dump is desired in order to best diagnose the problem. A SYSUDUMP DD statement is very adequate:

//SYSUDUMP DD SYSOUT=*

Simply add the DD statement to the failing job and recreate the failure. You can leave a SYSUDUMP DD statement in the JCL at all times if you prefer.

For started task NJE38 failures, no SYSUDUMP DD is required. NJE38 uses an ESTAE exit and SDUMP facilities to dump the necessary storage to one of the SYS1.DUMP datasets. The SYS1.DUMP datasets should be kept cleared of any old dumps from other unrelated system failures in the past. You can clear the dump datasets with this JCL, below. Note that completion code 12 is a normal completion for the clearing operation. You should run the job to clear all of your SYS1.DUMP datasets, e.g., SYS1.DUMP00, SYS1.DUMP01, etc., so they are always available when any abend occurs.

```
//CLEAR    EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSIN    DD DUMMY
//SYSUT1   DD DUMMY
//SYSUT2   DD DSN=SYS1.DUMP00,DISP=SHR
//
```

If NJE38 itself or the TSO TRANSMIT or RECEIVE commands abend, a dump is taken to one of the DUMP datasets, messages such as these will appear on the MVS console:

```
NJE38     LINK MVSA      ABEND S0C4
PSW   078D1000 00099852  ILC 06  INTC 0004
DATA NEAR PSW  0009984A  1B77D203 F0181610 41606001 07FEC3D7
GR 0-3   C0000000  C0000027  000C205C  000C208C
GR 4-7   0009F1A0  00000002  000BD85C  00000000
GR 8-11  0009D070  000BD000  000C2000  000B2470
GR 12-15 00099000  000C2440  A0099396  00000002
```

This information should be cut and pasted into a text file and provided along with the dump. The dump dataset can be printed with job PRDMP in the SAMPLIB dataset. Upon successful printing of the dump, clear the dump dataset that was used for the next use. The text file and the Hercules printout file can then be provided for analysis.

NJE38 Version History

15 February 2021 – v2.2.1

- Minor bug fixes

17 January 2021 – v2.2.0

- Text-based configuration file
- Wildcard routing support.
- Alternate routing support.
- Add or remove links dynamically by commands.
- New command D *nodeid*
- Registered user interface.
- Bug fix: Rare ABEND S0C4 with 2 CPUs when message comes in on one link and goes out on another.
- v2.0.1, v2.1.0, v2.1.1, v2.1.2 were internal releases only to implement the above changes.

21 July 2020 – v2.0.0

- New TSO commands TRANSMIT and RECEIVE.
- D FILES command displays spool full percentage.
- Bug fix: Some spool i/o buffers not freed on spool close.

03 June 2020 – v1.3.0

- Provide for a default userid to allow NJ38XMIT and NJ38RECV to operate if there is no security product enabled on the system.
- Bug fix: ABEND U0039 when multiple links start simultaneously with empty spool.
- Bug fix: MSG NJE039I VSAM return code/error code not formatted correctly.
- Bug fix: Cannot issue some commands against a node name that begins with a numeric digit.
- Bug fix: TSO NJE38 command doesn't identify its own userid if RAKF not installed on system.

22 May 2020 – v1.2.0

- New command RESET *filenum*
- PURGE/NOPURGE and FILE=*filenum* options for utility NJ38RECV
- Minor bug fixes

14 May 2020 – v1.1.0

- New command D *filenum*
- Updated documentation to help with file retrieval using the D *filenum* command

06 May 2020 – v1.02

- TSO userids consisting of 7-characters could cause abend D23 or message IKJ572I when NJE38 attempts to send a message to that TSO user.
- Display the assembly date and time of the NJECFG configuration member at NJE38 startup.
- D AUTH command not indicating 'no users authorized' if none are defined.
- D FILES command does not display local node name if no files present in the spool.
- Configuration file assembly error if only one AUTH USER= statement specified.

18 Apr 2020 – v1.01

- Add documentation describing steps to add TCPNJE support to Hercules.
- Cosmetic bug fix to the D NODES command output.

18 Mar 2020 – v1.00

- Support for authorizing local and remote users for certain NJE38 commands
- Dynamic additions and deletions to the authorization table
- New command AUTH
- New command UNAUTH
- Delete multiple spool files via "C" Cancel command.
- Confirmation of signon sent to the remote node.
- Bug fixes.

4 Mar 2020 – v0.90 beta

- Dynamic adjustment of the routing table
- New command D ROUTES
- New command ROUTE
- Several bugs fixed, including a storage overlay within NJE38 and messaging issues.

20 Feb 2020 – v0.85 beta

- Support for TSO user NJE commands and messages
- The system operator can send/receive text messages from remote users
- The system operator can issue commands to remote nodes
- Prevent S 522 wait time limit exceeded abends when no lines active
- Experimental and optional larger transmission buffer size
- Minor bug fixes

29 Jan 2020 – v0.80 beta

- First Beta release

Appendix A: Advanced Configuration Example

An example of a more advanced NJE configuration is presented, to illustrate how the various components should be defined.   Assume a slightly more complex arrangement from that presented in the installation instructions: four systems connected together in a ring. Conceptually, it looks like this:

```
MVSA-------MVSB-------VM1-------MVSC
  |                               |
  -------------------------------------------
```

Below are the particulars for each system.

System MVSA
NJE node name: MVSA
IP address of the host computer that Hercules is running: 192.168.1.20
Port number assigned to TCPNJE: 1175
BSC line addresses: 090, 091

System MVSB
NJE node name: MVSB
IP address of the host computer that Hercules is running: 192.168.1.21
Port number assigned to TCPNJE: 1175
BSC line address: 094, 096

System VM1 (a VM/370 system using RSCS V1)
NJE node name: VM1
IP address of the host computer that Hercules is running: 192.168.1.30
Port number assigned to TCPNJE: 1175
BSC line address: 040, 041

System MVSC
NJE node name: MVSC
IP address of the host computer that Hercules is running: 192.168.1.22
Port number assigned to TCPNJE: 1175
BSC line address: 091, 690

The Hercules TCPNJE device statements and the NJE38 configuration statements are shown on the next page.

Given the above information, here are the Hercules device definitions required; these should be placed in each Hercules configuration file:

Hercules A, where system MVSA will run:

```
0090 tcpnje 2703 lnode=MVSA  rnode=MVSB  lport=1175 rport=1175 rhost=192.168.1.21
0091 tcpnje 2703 lnode=MVSA  rnode=MVSC  lport=1175 rport=1175 rhost=192.168.1.22
```

Hercules B, where system MVSB will run:

```
0094 tcpnje 2703 lnode=MVSB  rnode=MVSA  lport=1175 rport=1175 rhost=192.168.1.20
0096 tcpnje 2703 lnode=MVSB  rnode=VM1    lport=1175 rport=1175 rhost=192.168.1.30
```

Hercules 1, where system VM1 will run:

```
0040 tcpnje 2703 lnode=VM1  rnode=MVSB  lport=1175 rport=1175 rhost=192.168.1.21
0041 tcpnje 2703 lnode=VM1  rnode=MVSC  lport=1175 rport=1175 rhost=192.168.1.22
```

Hercules C, where system MVSC will run:

```
0091 tcpnje 2703 lnode=MVSC  rnode=VM1    lport=1175 rport=1175 rhost=192.168.1.30
0690 tcpnje 2703 lnode=MVSC  rnode=MVSA  lport=1175 rport=1175 rhost=192.168.1.20
```

Here are the NJE38 configuration statements, and for completeness, a part of the VM/370 RSCS configuration statements are also shown.

NJE38 Configuration for MVSA	**NJE38 Configuration for MVSB**
LOCAL MVSA	LOCAL MVSB
*	*
LINK MVSB LINE 090 AUTO YES	LINK MVSA LINE 094 AUTO YES
LINK MVSC LINE 091 AUTO YES	LINK VM1 LINE 096 AUTO YES
*	*
ROUTE VM1 TO MVSB ALT MVSC	ROUTE MVSC TO MVSA ALT VM1

RSCS Configuration for VM1	**NJE38 Configuration for MVSC**
GENLINK ID=VM1,TYPE=LOCAL	LOCAL MVSC
*	*
GENLINK ID=MVSB,LINE=040,TYPE=DMTYJB,..	LINK MVSA LINE 690 AUTO YES
GENLINK ID=MVSC,LINE=041,TYPE=DMTYJB,..	LINK VM1 LINE 091 AUTO=YES
*	*
GENROUTE MVSA,MVSB	ROUTE MVSB TO VM1 ALT MVSA
END	

The choice of direction for the routes in a ring configuration is arbitrary. Alternate routing is not presently supported by VM/370 RSCS.

Appendix B: Sample Job to Unload the Tape

If you are unable to cut and paste the sample job **job000.txt** included in the distribution .zip file used to unload the tape image in step 1 of the installation, you may also try to cut/paste the JCL below, or key it in manually if formatting is lost.

```
//HERC01A JOB CLASS=A,MSGCLASS=X
//*
//* JOB000
//* THIS JOB UNLOADS THE NJE38 DISTRIBUTION TAPE
//*
//UNLOAD   PROC VOL=PUB000,                        <== VERIFY
//            SAMPLIB='NJE38.SAMPLIB',             <== VERIFY
//            AUTHLIB='NJE38.AUTHLIB',             <== VERIFY
//            CMDLIB='SYS2.CMDLIB'                 <== VERIFY
//*
//CPY      EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=*
//SYSUT3   DD UNIT=VIO,SPACE=(CYL,5)
//SYSUT4   DD UNIT=VIO,SPACE=(CYL,5)
//IN1      DD UNIT=TAPE,DSN=N38.SAMPLIB,DISP=(OLD,KEEP),
//            VOL=(,RETAIN,SER=NJE38),LABEL=(1,SL)
//IN2      DD UNIT=TAPE,DSN=N38.AUTHLIB,DISP=(OLD,KEEP),
//            VOL=(,RETAIN,SER=NJE38),LABEL=(2,SL)
//IN3      DD UNIT=TAPE,DSN=N38.CMDLIB,DISP=(OLD,KEEP),
//            VOL=(,RETAIN,SER=NJE38),LABEL=(3,SL)
//*
//OUT1     DD DISP=(NEW,CATLG),DSN=&SAMPLIB,
//            SPACE=(CYL,(1,1,5)),UNIT=SYSDA,VOL=SER=&VOL,
//            DCB=(BLKSIZE=3120,LRECL=80,RECFM=FB)
//OUT2     DD DISP=(NEW,CATLG),DSN=&AUTHLIB,VOL=SER=&VOL,
//            SPACE=(CYL,(1,1,5)),UNIT=SYSDA,
//            DCB=(BLKSIZE=6144,RECFM=U)
//OUT3     DD DISP=SHR,DSN=&CMDLIB
//         PEND
//RUN      EXEC UNLOAD
//CPY.SYSIN DD *
 COPY INDD=IN1,OUTDD=OUT1
 COPY INDD=IN2,OUTDD=OUT2
 COPY INDD=((IN3,R)),OUTDD=OUT3
/*
```

Appendix C: Security Considerations

Security considerations could require some additional preparation before you can successfully use all parts of NJE38. This is further complicated by the differences between everyone’s individual MVS 3.8 systems: some are running TK4- which has a security package installed (RAKF), and some are running TK3 or have completely generated their own systems, which may or may not have security installed.

There are three areas of consideration for security where related to NJE38:

1. Access to the NJE38 spool dataset by TSO users.
2. Authorizing the TSO-based NJE38, TRANSMIT, and RECEIVE commands.
3. Using defaults if your system has no security package installed or operational.

Each of these areas is discussed in turn, below, and provides information about how to address these concerns.

Providing TSO User Access to the NJE38 Spool Dataset (NJE38.NETSPOOL)

If your MVS 3.8 system has no security software installed or it is otherwise disabled, no action is required and this section can be skipped.

Users with security packages operating on their MVS 3.8 system (for example RAKF on TK4-) will likely need to add a rule to allow the NJE38.NETSPOOL dataset to be accessible by users. Otherwise, you may encounter an error using TRANSMIT or RECEIVE showing a VSAM error code X’0898’. In order to avoid this problem, add a rule that provides update access to the spool dataset. Below is an example rule added for RAKF on a TK4 system.

Edit dataset SYS1.SECURE.CNTL(PROFILES) to add the rule:

```
DATASET IPCS.DATA.SET.DIRECTRY                      UPDATE
DATASET IPCS.PROBLEM.DIRECTRY                        UPDATE
*
DATASET NJE38.NETSPOOL                              UPDATE
*
* added  'RPF.V1R5M3.SRPFPROF'      uacc(update)
*
DATASET RPF.V1R5M3.SRPFPROF                         UPDATE
```

You may need to use a TSO userid with system programmer authority (such as HERC01 in TK4-) in order to edit this member. Add the line in bold as shown. The non-bolded lines may appear differently or may not be present on your system and are simply shown for reference.

After you add the DATASET line for the NJE38.NETSPOOL dataset, save your changes and issue a “S RAKF” command from the MVS operator console in order to have RAKF be aware of the change. Alternately, the change will take effect on the next IPL.

Authorizing the NJE38-Related TSO Commands

The TSO commands NJE38, TRANSMIT, and RECEIVE present a special case because they must run APF-authorized in order to communicate with the started-task NJE38, but TSO itself is normally an unauthorized environment. Several other commands and programs fall into this category, such as RPF, QUEUE, IEBCOPY and more.

The command name “NJE38”, “TRANSMIT”, “XMIT”, and “RECEIVE” must be placed into a table and linked into a TSO load module in order to allow these commands to run APF-authorized. Depending on several factors outlined below, you may not need to do anything else, or you may need to run a provided job stream (TSOAUTH) in order to add these commands to the authorized commands table. Determine if your situation falls into one of the cases below and decide if you need to run the job to update the table. If you need to run the job, details are provided below.

If you are using the TK4- build of MVS 3.8, and:

- you primarily use only the HERC01/HERC02 userids for TSO: no further action required. This case probably represents most users.
- you use or have users that use other TSO userids besides HERC01/HERC02: run the job.

If you are using the TK3 Turnkey MVS 3.8 system: run the job.

If you are using a MVS 3.8 system that you have generated and customized yourself: run the job.

Running the TSOAUTH Job

Within the NJE38 SAMPLIB dataset is a member named TSOAUTH. This job is used to add NJE38 to the list of commands that should be authorized under TSO. These commands have already been added to the list for you. Run the job in order to assemble the new list and link it into the system. After the job is completed, **YOU MUST RE-IPL WITH THE CLPA OPTION** in order for this change to take effect.

TK4- users could alternately run the job that is in SYS1.SYSGEN.CNTL(SG0350) which essentially performs the same function. This job was used to add the other necessary commands to the table when TK4- was built. This is probably the preferred method for TK4- use because it uses SMP in order to apply the change. However, for the purposes of this document and to stay on target where NJE38 is the subject at hand, the TSOAUTH job was provided as an alternative because it should work on all versions of MVS 3.8.

Note that the next scheduled update of the TK4- MVS system is slated to have these commands already within the authorized commands list, as well as having the NJE38 AUTHLIB dataset already added to IEAAPF00 in SYS1.PARMLIB.

Using Defaults if Your System Has no Security Package Installed or Operational.

NJE38 depends on system security information to determine the userid of the TSO user who submitted jobs to retrieve files from the spool or to place files into the spool for transmission. This helps by preventing others (in a multiple user system) from retrieving your files, for example. However, it is recognized that not all MVS 3.8 systems are running with a security package, such as RAKF, installed or enabled on their systems.

In a system without security, it cannot reliably be determined by a batch job which user (whether a TSO user or some other user via the card reader) is the submitter. Using the jobname does not necessarily guarantee a relationship with the submitting TSO userid, and the USER= parameter on the JOB statement also provides no information (when there is no security package). Therefore, some other method is needed in order to provide userid information.

If your system is running without security enabled, you should specify an additional parameter on the LOCAL statement of the NJE38 configuration member (CONFIG). This parameter defines a default userid to be used when there is no security information available. This will allow NJE38 to transmit and receive files on a default basis although no security rules can be enforced. The additional parameter is the DEFUSER parameter on the previously discussed LOCAL statement:

LOCAL Statement

The configuration member must consist of a single LOCAL entry, which must be first.

The format of the LOCAL statement is:

LOCAL *nodeid*   DEFUSER *userid*

The DEFUSER parameter allows you to specify a userid that will be the default userid used by NJE38. <u>When there is no security</u>, the following actions will be taken by NJE38 and its commands and utilities:

- files queued for transmission by batch utility NJ38XMIT or by the TRANSMIT command (batch execution only) will be assigned the default userid, so that the receiver will see this userid as the originator of the file. This userid will appear in Display commands that show spool contents. When the TRANSMIT command is invoked from a TSO terminal, the actual TSO userid is the userid assigned to the transmitted file.

- The batch NJ38RECV utility or the RECEIVE command (from a TSO terminal or batch execution) can retrieve any spool file that has a destination in the local node, regardless of the destination userid.

- Responses issued by the TSO NJE38 command will be directed to the default userid if and only if the TSO NJE38 command is issued in batch execution. Otherwise, the command responses will directed to the terminal of the issuer.

The userid chosen for the DEFUSER value should ideally be the same as your routinely used TSO userid. This will allow ‘file sent’ and ‘file received’ messages to be routed to you. <u>If you do not code DEFUSER, the value defaults to HERC01 as the default userid</u>.

Please note that the purpose of DEFUSER is for single user systems without any security enabled. If your system has multiple users and you have no security available, NJE38 cannot and will not prevent unauthorized sending and retrieval of files by others for which they are not the owner. In other words, without security then file ownership cannot be enforced.

If your system has security enabled, the DEFUSER value is ignored and has no meaning.

Appendix D: Loading the NJE38 Optional Source Materials

The source code for NJE38 is provided on the installation tape image in files 4 and 5. The source code is not required in order to use NJE38. If you wish to load the optional source materials, use JOB040 in the NJE38 SAMPLIB dataset to retrieve them from the tape. JOB040 will create two additional datasets: NJE38.ASMSRC and NJE38.MACLIB. Within NJE38.ASMSRC is a member named ASM. This is the JCL used to assemble and link all of the NJE38 components.