NAME

tnos -- Tampa Network Operating System

SYNOPSIS

tnos [*options...*]

DESCRIPTION

This manual page reflect TNOS release 2.03 and greater.

TNOS is an application which makes your computer a complete TCP/IP-capable machine, complete with support for Amateur Packet radio, via the AX25 Link Level Protocol. *TNOS* support both MS-DOS (or Windows, Win-95, OS-2), and Unix. The MS-DOS version is limited to the DOS 640K conventional memory boundaries, which limits the number of servers and features that can be compiled into the executable.

TNOS really shines, though, on a Unix platform, like Linux! ALL of *TNOS*'s features can be compiled in, and the *TNOS* box can act as a completely separate virtual machine from the normal Linux environment, allowing for a secure Amateur firewall.

Most TCP/IP servers are contained within *TNOS*, including FTP, telnet, HTTP, NNTP, and many more. There are also some special servers that are particular to Amateur Radio, including the Ham version of IRC, the Conference Bridge.

Also contained in *TNOS* is a full-featured Packet Bulletin Board System *(PBBS)*, fully compatible with non-NOS PBBSs. It supports compressed and non-compressed PBBS forwarding, file upload/download, and lots more. In addition, the PBBS can be accessed via the Internet (assuming your station has Internet connectivity), from anyone's WWW Browser!

HISTORY - FROM THE AUTHOR

There was no original intention to start a "*TNOS*" project. I stumbled onto GRINOS and found it to be FUN and CHALLENGING. This caused me to look around and I found JNOS, which was more complete and stable. After initial testing, I ws ready to replace the AA4RE BBS that I was running at our club station. I asked the operator of the BBS that was feeding me traffic to hold off for a few days while I made the change.

All went well, everything was in place, users were happy. So I told the BBS SYSOP to start feeding me traffic. BOOM! CRASH! POW! POP! There were 30 second (and greater) delays changing into certai message areas that held more than 500 messages. Messages selected for forwarding never left the system, since the other BBS was timing out waiting for my system to FIND what I wanted to send.

And trying to remotely SYSOP left me frustrated, since very few commands could be executed without going into remote sysop mode, and back again.

And so the sleepless nights started! I became labeled by those helping with the project as the "programmer that never sleeps"! Mad dashes through the code were made to make temporary adjustments to bring the performance up to an acceptable level.

After getting several in the Tampa area to serve as beta-testers (in addition to the original club BBS), I then set out to clean up the rubble and make sense of it.

Then (while on a roll) I decided to enhance many of the existing features, including the Conference Bridge. Additional servers were added. A few GUI "frills" were added. A few adjustments were made to make it work better with ROSE, since that is what our local network uses.

Thus begat TNOS! Several were critical that I did not let the world get their hands on it earlier, but since this was never MEANT to be anything other than a personal project, I did not see that I had the time to put into supporting a project in turmoil. I waited until it was stable (relatively) and not undergoing daily changes. I'm sorry if that didn't please everyone, but (to paraphrase the song), "It's my code and I'll release when I want to".

COPYRIGHT INFO

OPTIONS

-?

Displays the version information on this *TNOS* executable.

-A devices

Defines the maximum number of SLIP/AX25-capable serial devices allowed. This number must be the same or smaller than the number of physical ASY serial ports defined. If this setting is higher than the number of serial devices (set with the '-P' option), then both are set to this value. The default number of devices is 5.

-B processes

Defines the maximum number of BBS processes allowed. The default number of processes is 40.

-D definition

Defines a new environment variable for use within *TNOS*. The *definition* may be in one of two formats, *label=value*, or *label*. If the latter form is used, the value of *label* will be set to the name of the label. This command is only available to executables that have had the *SCRIPTING* flag defined in their compile.

-E

Instructs *TNOS* to import all environment variables from the OS. This command is only available to executables that have had the *SCRIPTING* flag defined in their compile.

-P serialdevices

Defines the maximum number of ASY serial devices allowed. If the setting for the '-A' option is greater, than both are set to that value. The default number of devices is 5. This command is available on release 2.21 and greater.

-S manager

Sets default session manager. (Unix versions only)

-T manager

Sets session manager to be used for the trace session. (Unix versions only)

-U timeout

Sets the timeout value for the auto-update feature, available to those with the TNOS Browser feature compiled in. The timeout waited to make the connection and check for updates defaults to 30 seconds. Give a new value here for 'timeout' to change this. Those sites NOT connected to the Internet may want to use a '-U 0' here, to disable the 30 second check on booting. This command is available on release 2.21 and greater.

-V

Give the same output as with the *-v* options, but also prompts you at each command, as to whether you wish to execute it or not. Also good for diagnosing a troubled autoexec.nos file.

-a interfaces

Defines the maximum number of AXIP interfaces allowed. The default number of AXIP interfaces is 16. This command is only available to executables that have had the *AXIP* flag defined in their compile.

-c
Sets a TNOS 68000 system to use Curses for it's screen display management. There **are** no systems of this type available outside of the author's bench. This command is only available to executables that have had the *TNOS_68K* flag defined in their compile.

-d rootdir
Sets the *TNOS* root directory to *rootdir*. By default, the rootdir is '.' for Unix and '/' for MS-DOS.

-e
Use EGA mode for the screen output, giving either a 43 or a 50 line display (depending on the display adapter). (MS-DOS version only)

-f configfile
Instructs *TNOS* to use the file *configfile* as a file to define alternate definitions for the internal *TNOS* filename strings.

-g colorname
Sets the foreground color for *TNOS*.

-h heapsize
Sets the initial heapsize to a value other than the default of 100K. This value is the size in bytes, which defaults to 102400. (MS-DOS version only)

-k colorname
Sets the background color for *TNOS*.

-l lockdirpath
Sets the directory used for the LOCKDIR path. (Unix version only) This command is available on release 2.22 and greater.

-m
Displays the TNOS features MAP information on this *TNOS* executable. This command is available on release 2.20 and greater.

-n
Disables tracing to a separate session. This will make all trace output go to the *TNOS* console, instead of to the session accessible via the *F9* function key. This command is only available to executables that have had the *TRACE* flag defined in their compile.

-o sessions
Sets the maximum number of interactive console sessions. This cannot be set to less than 5, The default for this is set in the mkconfig.tcl development program, and is usually set to 20.

-r circuits
Defines the maximum number of Netrom open circuits allowed. The default number of circuits is 20. This command is only available to executables that have had the *NETROM* flag defined in their compile.

-s sockets
Sets the maximum number of sockets available within *TNOS*. The default number of sockets is set up in the mkconfig.tcl development program, and is usually set to 40.

-t
Disables internal timers - for debugging purposes only. (Unix versions only)

-v
Startup in verbose mode. This causes all commands in the autoexec.nos file to be displayed before they are executed, to help determine where you might be having errors in processing that file.

AUTHOR

The author, mastermind, tempermental ruler, and primary fan of *TNOS* is Brian A. Lantz <brian@lantz.com>.

PLEASE DO NOT send *TNOS* support email to that address. *TNOS* support is handled ONLY on the TNOS-TOPICS and TNOS-SUPPORT mailing lists.

SEE ALSO

The *TNOS* Frequently Asked Questions *(FAQ)* document. Do a **finger tnos@lantz.com** for info on how to obtain the *FAQ* document.

The *TNOS-TOPICS* mailing list is a great user-supported way of getting the help you may need from time to time. The same **finger** command listed above will give you info on this, also.

Voluntarily registering your *TNOS* will get you access to support from the author of *TNOS*, Brian A. Lantz, through a special author-supported mailing list (TNOS-SUPPORT) available for only **registered users** of *TNOS*.

You can find *TNOS CENTRAL*, with the definitive information about *TNOS* at http://www.lantz.com.

The alpha code development machine (that the author uses to do the actual *TNOS* development on) can be found by telneting to ko4ks.ampr.org or on the WWW at http://ko4ks.ampr.org.

BUGS

Surely you joke ;-)

COPYRIGHT

TNOS DOCUMENTATION PROJECT

This MAN page is part of the TNOS Documentation Project and is maintained by the author of TNOS, Brian A. Lantz/KO4KS, and is Copyright 1996 by Brian A. Lantz on behalf of the TNOS Documentation Project.

For further information on the TNOS Documentation Project, see the TDP Web Pages. <http://www.lantz.com/TDP/>