

Linux



Connect, Inc.
1701 Quincy Avenue, Suites 5 & 6, Naperville, IL 60540
Ph: (630) 717-7200 Fax: (630) 717-7243
www.connectrf.com

Table of Contents

Linux OpenAir Logon and CRFADMIN.....	1
Operating System Change on All OpenAir Linux Systems.....	3
Synchronizing Time Under Linux OpenAir.....	8
Changing the Date/Time and Timezone Under Linux.....	9
Restarting INETD Under Linux.....	11
Commands Similar in Function to “tcp stop and start”.....	12
Setting Inactivity Session Timer in a UNIX-like Environment.....	13
Settings for Hyperterminal or Windows Telnet on a Linux Server.....	14
Standard UNIX Error Numbers.....	15
R7 Terminal Clients and R6 OpenAir Servers.....	19
Linger Issue on IX Hosts Produces Waiting for Data/Host Process Delay ..	21
Restarting APBASE when Host Sessions Fail to End.....	22
Linux Console Error Message About Unmounting.....	26
Linux Consuming Memory.....	27
Changing Linux Login To Prompt For Terminal Type.....	28

Linux OpenAir Logon and CRFADMIN

Connect has made the following change to UNIX based hard NCUs:

All IX servers being shipped are Linux based. The only SCO based servers shipped will be those connectivities that have not completed full testing. Connect CWNEs who have been installing SCO based Zoomboxes will see the same familiar features and user interface with the IX OpenAir Linux based servers.

Operating System Differences

Logging onto to the controller under Linux:

- 1) At the login prompt, type "ops" and press Enter.

```
Debian GNU/Linux 1.2
Copyright (C) 1993-1996 Software in the Public Interest, Inc. and others
PowerNet 6.3.6 Copyright 1992-2003 Connect, Inc. All Rights Reserved.
debian login: ops
```

- 2) At the password prompt, type "ops123" and press Enter.

```
Password:
```

- 3) After logging in, at the \$ prompt, type "su" and press Enter.

```
Last login: Wed Jun 23 11:53:11 from 206.183.67.222
Linux debian 2.0.33 #6 Mon Jun 14 07:52:56 CDT 1999 i686
$ su
```

- 4) After changing to super user, the prompt will change from a "\$" to a "#". At the "#" prompt, type "crfadmin" and press Enter.

```
# crfadmin
```

- 5) This will bring up the PowerNet Main menu.

```
Linux                PowerNet IX
+ Main Menu          +
|* 1 Operations      |
| 2 Spectrum One    |
| 3 Access Point     |
| 4 VTERM            |
| 5 TN3270           |
| 6 TN5250           |
| 7 SAB Emulation    |
| 8 TCP/IP-CS        |
| 9 TCP/IP-STEP      |
| 10 Object Editors  |
| 11 Utilities       |
+-----+-----+
```

Note: Once at the Main Menu, the options will be the same as under SCO.
On all Linux based Open Air NCUs, the directory is /home/crf.
Previously under SCO, the directory was /crf.

Logging out of the controller under Linux:

- 1) At the PowerNet Main Menu, press the "Esc" key.
Press the "y" key to confirm exiting the main menu.

Exit (y/n)

- 2) At the "#" prompt, type "exit" and press Enter.

exit

- 3) At the "\$" prompt, type "exit" and press Enter.

\$ exit

Changing Terminal Types

If you would like to change the terminal type, you can do this under Linux using the environment variable TERM.

The commands are:

```
#TERM=vt100
```

```
#export TERM
```

Operating System Change on All OpenAir Linux Systems

Connect has made the following change to UNIX based hard NCUs:

All IX servers being shipped are Linux based. The only SCO based servers shipped will be those connectivities that have not completed full testing. Connect CWNEs who have been installing SCO based Zoomboxes will see the same familiar features and user interface with the IX OpenAir Linux based servers.

Operating System Differences

MSDOS commands under Linux:

```
# man mtools
```

```
MTOOLS(1)
```

```
MTOOLS(1)
```

```
NAME
```

```
  mtools - a collection of tools for manipulating MSDOS files
```

```
SYNOPSIS
```

```
  mattrib - change MSDOS file attribute flags
  mbadblocks - tests a floppy disk, and marks the bad blocks in the FAT
  mcd - change MSDOS directory
  mcopy - copy MSDOS files to/from UNIX
  mdel - delete an MSDOS file
  mdir - display an MSDOS directory
  mformat - add an MSDOS filesystem to a low-level formatted floppy disk
  mlabel - make an MSDOS volume label
  mmd - make an MSDOS subdirectory
  mmount - mount an MSDOS disk
  mrd - remove an MSDOS subdirectory
  mmove - move or rename an MSDOS file or subdirectory
  mren - rename an existing MSDOS file
  mtype - display contents of an MSDOS file
  mtest - tests and displays the configuration
```

```
DESCRIPTION
```

Mtools is a public domain collection of programs to allow UNIX systems to read, write, and manipulate files on an MSDOS filesystem (typically a floppy disk). Where reasonable, each program attempts to emulate the MSDOS equivalent command. However, unnecessary restrictions and oddities of DOS are not emulated. For instance, it is possible to move subdirectories from one subdirectory to another.

MSDOS filenames are optionally composed of a drive letter followed by a colon, a subdirectory, and a filename. Filenames without a drive letter refer to UNIX files. Sub directory names can use either the '/' or '\' separator. The use of the '\' separator or wildcards will require the names to be enclosed in quotes to protect them from the shell. (Note: Wildcards in UNIX filenames should not be enclosed in quotes, because here we want the shell to expand them.)

Differences With MSDOS

The regular expression "pattern matching" routines follow the UNIX-style rules. For example, '*' matches all MSDOS files in lieu of '*.*'. The archive, hidden, read-only and system attribute bits are ignored during pattern matching.

All options use the '-' (minus) flag, not '/' as you'd expect in MSDOS.

Most mtools commands allow multiple filename parameters, which does not follow MSDOS conventions, but which is more user-friendly.

local 1

MTOOLS(1)

MTOOLS(1)

WORKING DIRECTORY

The mcd command is used to establish the device and the current working directory (relative to the MSDOS filesystem), otherwise the default is assumed to be A:/.

However, unlike MSDOS, there is only one working directory, and not one per drive.

VFAT-style long filenames

This version of mtools supports VFAT style long filenames. If a UNIX filename is too long to fit in a short DOS name, it is stored as a VFAT long name, and a companion short name is generated. This short name is what you see when you examine the disk with a pre-7.0 version of DOS.

The following table shows some examples of short names:

UNIX name	MSDOS name	Reason for the change
thisisatest	THISISAT	filename too long
alain.knaff	ALAIN.KNA	extension too long
prn.txt	XRN.TXT	PRN is a device name
.abc	X.ABC	null filename
hot+cold	HOTXCOLD	illegal character

The initial UNIX-style filename (whether long or short) is also called primary name, and the derived short name is also called secondary name.

Example:

```
mcopy /etc/motd a:Reallylongname
```

Mtools creates a VFAT entry for Reallylongname, and uses REALLYLO as a short name. Reallylongname is the primary name, and REALLYLO is the secondary name.

```
mcopy /etc/motd a:motd
```

Motd fits into the DOS filename limits. Mtools does not need to derivate another name. Motd is the primary name, and there is no secondary name.

In a nutshell: The primary name is the long name, if one exists, or the short name if there is no long name.

NAME CLASHES

When writing a file to disk, its long name (primary name) or short name may collide with an already existing file or directory. This may happen for all commands that create new directory entries (e.g. B mcopy, mmd, mren, mmove, mwrite and mread).

When a name clash happens, mtools asks what it should do.

local 2

MTOOLS(1) MTOOLS(1)

It offers several choices:

overwrite

Overwrites the existing file. It is not possible to overwrite a directory with a file.

rename

Renames the newly created file. Mtools will prompt for the new filename.

autorename

Renames the newly created file. Mtools will choose a name by itself, without prompting.

skip

Gives up on this file, and moves on to the next (if any).

To choose an option, type its first letter at the prompt. If you use a lower case letter, the option only applies for this file only. If you use an upper case letter, the option applies to all files.

You may also chose options (for all files) on the command line, when invoking mtools:

- o Overwrites primary names by default.
- O Overwrites secondary names by default.
- r Renames primary name by default.
- R Renames secondary name by default.
- a Autorenames primary name by default.
- A Autorenames secondary name by default.
- s Skip primary name by default.
- S Skip secondary name by default.
- m Ask user what to do with primary name.
- M Ask user what to do with secondary name.

By default, the user is prompted if the primary name clashes, and the secondary name is autorenamed.

If a name clash occurs in a UNIX directory, mtools only asks whether to overwrite the file, or to skip it.

Case Sensitivity of The VFAT Filesystem

The VFAT filesystem is able to remember the case of the filenames.

local 3

MTOOLS(1)

MTOOLS(1)

However, filenames that differ only in case are not allowed to coexist in the same directory. For example, if you store a file called LongFileName on a VFAT filesystem, mdir will show this file as LongFileName, and not as Longfilename. However, if you then try to add Long-Filename to the same directory, it will be refused, because case is ignored for clash checks.

The VFAT filesystem allows one to store the case of a filename in the attribute byte, if all letters of the filename are the same case and if all letters of the extension are the same case. Mtools uses this information when displaying the files and to generate the UNIX when mcopying to a UNIX directory. This may have unexpected results when applied to files written using a pre-7.0 version of DOS. These filenames map to all uppercase. This is different from the behavior of the old version of mtools, which used to generate lower case UNIX filenames.

XDF DISKS (LINUX ONLY)

Xdf is a high capacity format supported by OS/2. It can hold 1840k per disc. This is not very high compared to the best 2m formats, but its main advantage is that it is fast: 600 milliseconds per track. This is faster than the old 21sector format, and almost as fast as the standard 18 sector format. In order to access these disks, set the use_xdf variable for the drive. See mtools (5) for details. Fast Xdf access is only available for kernels more recent than 1.1.34.

Caution / Attention distributors: If mtools is compiled on a Linux kernel more recent than 1.3.34, it will not run on an older kernel. However, if it has been compiled on an older kernel, it still runs on a newer kernel, but Xdf access is slower. It is recommended that distribution authors only include mtools binaries compiled on kernels older than 1.3.34 until 2.0 comes out. When 2.0 is out, mtools binaries compiled on newer kernels may (and should) be distributed. Mtools binaries compiled on kernels older than 1.3.34 will not run on a 2.1 or later kernel.

EXIT CODES

All the Mtools commands return 0 on success, 1 on utter failure, or 2 on partial failure. All the Mtools commands perform a few sanity checks before going ahead to make sure that the disk is indeed an MSDOS disk (as opposed to, say an ext2 or minix disk). These checks may reject partially corrupted disks, which might otherwise be readable. To avoid these checks, set the MTOOLS_SKIP_CHECK environmental variable.

local 4

MTOOLS(1)

MTOOLS(1)

SEE ALSO

mattrib(1), mbadblocks(1), mcd(1), mdel(1), mformat(1), mmove(1), mrd(1), mren(1),
mtype(1), mcopy(1), mdir(1), mlabel(1), mmd(1), mmount(1)

BUGS

An unfortunate side effect of not guessing the proper device (when multiple disk capacities are supported) is an occasional error message from the device driver. These can be safely ignored.

The fat checking code chokes on 1.72 Mb disks mformatted with pre-2.0.7 mtools. Set the environmental variable MTOOLS_FAT_COMPATIBILITY to bypass the fat checking.

The support for non-Linux OS variants has not been tested for a long time. It may contain bugs, or not work at all.

local

Synchronizing Time Under Linux OpenAir

To synchronize time under Linux OpenAir from another TCP IP address, set up a crontab entry to execute the netdate command.

The syntax tested is "netdate tcp hostname". The hostname is the host from which you want to get the time.

Changing the Date/Time and Timezone Under Linux

If you have a keyboard and VGA monitor connected to the system, you can set the date/time with the CMOS setup utility. If you would like to do it remotely OR you don't have a keyboard/monitor installed, you can use the procedures below.

To change the date and time, the command is "date MMDDhhmm[[CC]YY][.ss]".

Where:

- MM month within year
- DD day within month
- hh hour
- mm minute
- CC first two digits of year (optional)
- YY last two digits of year (optional)
- ss second (optional)

To change the time zone, run "tzconfig".

Setting the CMOS Clock

Type "date" and enter to see the setting of the system clock. Type "clock" and enter to see the setting of the cmos clock. If the system clock is correct, use "clock -w" to set the cmos clock from the system clock. If the cmos clock is correct, use "clock -s" to set the system clock from the cmos clock. If both clocks are off, set the system time using "date MMDDhhmmCCYY.ss" (month-day-hour-minute-century-year.seconds). Use "clock -w" to set the cmos clock from the new setting of the system clock just changed.

Note: The date command will NOT update the CMOS hardware clock. When a reboot is done, Linux will read the CMOS date/time and set the Linux date/time to match.

Another Method

The latest published clockspeed package id found at:
<http://cr.yp.to/clockspeed/clockspeed-0.62.tar.gz>

Definitions

clockspeed - uses a hardware tick counter to compensate for a persistently fast or slow system clock. Given a few time measurements from a reliable source, it computes and then eliminates the clock skew. This version of clockspeed can use the Pentium RDTSC tick counter or the Solaris gethrtime() nanosecond counter.

sntpclock - checks another system's NTP clock, and prints the results in a format suitable for input to clockspeed. sntpclock is the simplest available NTP/SNTP client.

taiclock and **taiclockd** - even simpler alternatives to SNTP. They are suitable for precise time synchronization over a local area network, without the hassles and potential security problems of an NTP server.

Restarting INETD Under Linux

In some network environments, Linux will disable inetd if it gets more than 60 connections per second on port 23.

The solution is to identify the external event causing the problem. This will need to be done with a network analyzer at the site. Read the logs to see the source of the traffic.

In the meantime, the process can be restarted by issuing the following command from the root prompt:

```
# start-stop-daemon --start --quiet --exec /usr/sbin/inetd
```

The number of connections per second may be increased in the inetd.conf file in the /etc/ directory.

Commands Similar in Function to "tcp stop and start"

The following commands perform a function similar to "tcp stop and start" under SCO by resetting the adapter.

<code>ifconfig eth0 down</code>	- turns NIC adapter OFF
<code>ifconfig eth0 up</code>	- turns NIC adapter ON
<code>/etc/init.d/network</code>	- same command as the menu option "Apply Changes" and is required to set address, etc.

Setting Inactivity Session Timer in a UNIX-like Environment

Depending on the IX variant, the inactivity session timer can be set from the locations described below:

SCO: Check the IDLETIME variable found in the /etc/default/idleout file.

```
crfv5# cat idleout
```

```
:  
#    @(#) idl.def.src 23.1 91/08/29  
#  
#    Copyright (C) 1987-1991 The Santa Cruz Operation, Inc.  
#    All Rights Reserved.  
#    The information in this file is provided for the exclusive use of  
#    the licensees of The Santa Cruz Operation, Inc.  Such users have the  
#    right to use, modify, and incorporate this code into other products  
#    for purposes authorized by the license agreement provided they include  
#    this notice and the associated copyright notice with any such product.  
#    The information in this file is provided "AS IS" without warranty.  
#    Timeout in Minutes
```

```
IDLETIME=120
```

Placing the "idleout(ADM)" command in the /etc/rc2.d/S88USRDEFINE file runs the command automatically at boot.

The inactivity session timer can also be set under the Security and Accounting package.

AIX: Same as SCO above.

Settings for Hyperterminal or Windows Telnet on a Linux Server

Configure Hyperterminal for ANSI. After logging in to the OpenAir Linux server, use the following commands:

```
TERM=pcansi  
export TERM
```

Note: These commands are case sensitive.

'Ctrl c' will now exit correctly and 'Ctrl z' will exit unconditionally creating a core file.

Standard UNIX Error Numbers

These error numbers and meanings are common across most versions of UNIX. They are returned to a running application if there is an error with a UNIX system or library call.

At times you may see these show up in PowerNet trace files. Below is an example from a trace of a TCP/IP client streaming process:

```
01/26 10:47:39.700 0ms Polling
01/26 10:47:39.870 170ms Poll returned events HOST: 00001 RF: 00000
01/26 10:47:39.870 0ms link_recv failed errno: 108
01/26 10:47:39.870 0ms Shutting down...
```

Looking up this error in the list below, we find number 108 is:

ECONNRESET (TCPERR(90)+18) Connection reset by peer

This tells us that the connection was reset by the host.

You can also find this list on most UNIX systems by looking at the file:

```
/usr/include/sys/errno.h
```

If not located there, you can use the "find" command to locate this file.

The syntax for this is:

```
find / -name errno.h -print
```

UNIX ERROR NUMBERS - FROM SCO V4.2

Copyright (C) 1988-1992 The Santa Cruz Operation, Inc.
All Rights Reserved.

The information in this file is provided for the exclusive use of the licensees of The Santa Cruz Operation, Inc. Such users have the right to use, modify, and incorporate this code into other products for purposes authorized by the license agreement provided they include this notice and the associated copyright notice with any such product. The information in this file is provided "AS IS" without warranty.

Copyright (c) 1984, 1986, 1987, 1988 AT&T
All Rights Reserved

THIS IS UNPUBLISHED PROPRIETARY SOURCE CODE OF AT&T

The copyright notice above does not evidence any actual or intended publication of such source code.

Name	Number	Description
EPERM	1	Not owner
ENOENT	2	No such file or directory
ESRCH	3	No such process
EINTR	4	Interrupted system call
EIO	5	I/O error
ENXIO	6	No such device or address
E2BIG	7	Arg list too long
ENOEXEC	8	Exec format error
EBADF	9	Bad file number
ECHILD	10	No child processes
EAGAIN	11	No more processes
ENOMEM	12	Not enough space
EACCES	13	Permission denied
EFAULT	14	Bad address
ENOTBLK	15	Block device required
EBUSY	16	Device busy
EEXIST	17	File exists
EXDEV	18	Cross-device link
ENODEV	19	No such device
ENOTDIR	20	Not a directory
EISDIR	21	Is a directory
EINVAL	22	Invalid argument
ENFILE	23	File table overflow
EMFILE	24	Too many open files
ENOTTY	25	Not a typewriter
ETXTBSY	26	Text file busy
EFBIG	27	File too large
ENOSPC	28	No space left on device
ESPIPE	29	Illegal seek
EROFS	30	Read-only file system
EMLINK	31	Too many links
EPIPE	32	Broken pipe
EDOM	33	Argument out of domain
ERANGE	34	Result too large
ENOMSG	35	No message of desired type
EIDRM	36	Identifier removed
ECHRNG	37	Channel number out of range
EL2NSYNC	38	Level 2 not synchronized
EL3HLT	39	Level 3 halted
EL3RST	40	Level 3 reset
ELNRNG	41	Link number out of range
EUNATCH	42	Protocol driver not attached
ENOCSI	43	No CSI structure available
EL2HLT	44	Level 2 halted
EDEADLK	45	Deadlock situation detected/avoided
ENOLCK	46	No record locks available

Convergent Error Returns

EBADE	50	Bad exchange descriptor
EBADR	51	Bad request descriptor
EXFULL	52	Message tables full
ENOANO	53	Anode table overflow
EBADRQC	54	Bad request code
EBADSLT	55	Invalid slot
EDEADLOCK	56	File locking deadlock
EBFONT	57	Bad font file format

Stream Problems

ENOSTR	60	Not a stream device
ENODATA	61	No data available
ETIME	62	Timer expired
ENOSR	63	Out of stream resources
ENONET	64	Machine is not on the network
ENOPKG	65	Package not installed
EREMOTE	66	Object is remote
ENOLINK	67	Link has been severed
EADV	68	Advertise error
ESRMNT	69	Srmount error
ECOMM	70	Communication error on send
EPROTO	71	Protocol error
EMULTIHOP	74	Multihop attempted
ELBIN	75	Inode is remote (not really error)
EDOTDOT	76	Cross mount point (not really error)
EBADMSG	77	Not a data message
ENAMETOOLONG	78	Filename too long
EOVERFLOW	79	(Reserved for future use)
ENOTUNIQ	80	Name not unique on network
EBADFD	81	File descriptor in bad state
EREMCHG	82	Remote address changed

Shared Library Problems

ELIBACC	83	Can not access a needed shared library
ELIBBAD	84	Accessing a corrupted shared library
ELIBSCN	85	.lib section in a.out corrupted
ELIBMAX	86	Attempting to link in too many shared libs.
ELIBEXEC	87	Can not exec a shared library directly
EILSEQ	88	Illegal byte sequence
ENOSYS	89	Function not implemented

System V Streams TCP

TCPERR 90

Non-Blocking and Interrupt I/O

EWOULDBLOCK	(TCPERR+0)	Operation would block
EINPROGRESS	(TCPERR+1)	Operation now in progress
EALREADY	(TCPERR+2)	Operation already in progress

LPC/Network Software - Argument Errors

ENOTSOCK	(TCPERR+3)	Socket operation on non-socket
EDESTADDRREQ	(TCPERR+4)	Destination address required
EMSGSIZE	(TCPERR+5)	Message too long
EPROTOTYPE	(TCPERR+6)	Protocol wrong type for socket
EPROTONOSUPPORT	(TCPERR+7)	Protocol not supported
ESOCKTNOSUPPORT	(TCPERR+8)	Socket type not supported
EOPNOTSUPP	(TCPERR+9)	Operation not supported on socket
EPFNOSUPPORT	(TCPERR+10)	Protocol family not supported
EAFNOSUPPORT	(TCPERR+11)	Address family not supported by protocol family
EADDRINUSE	(TCPERR+12)	Address already in use
EADDRNOTAVAIL	(TCPERR+13)	Can't assign requested address

Operational Errors

ENETDOWN	(TCPERR+14)	Network is down
ENETUNREACH	(TCPERR+15)	Network is unreachable
ENETRESET	(TCPERR+16)	Network dropped connection on reset
ECONNABORTED	(TCPERR+17)	Software caused connection abort
ECONNRESET	(TCPERR+18)	Connection reset by peer
ENOBUFS	ENOSR	Out of stream resources
EISCONN	(TCPERR+20)	Socket is already connected
ENOTCONN	(TCPERR+21)	Socket is not connected
ESHUTDOWN	(TCPERR+22)	Can't send after socket shutdown
ETOOMANYREFS	(TCPERR+23)	Too many references: can't splice
ETIMEDOUT	(TCPERR+24)	Connection timed out
ECONNREFUSED	(TCPERR+25)	Connection refused
EHOSTDOWN	(TCPERR+26)	Host is down
EHOSTUNREACH	(TCPERR+27)	No route to host
ENOPROTOPT	(TCPERR+28)	Protocol not available

XENIX error numbers

EUCLEAN	135	Structure needs cleaning
ENOTNAM	137	Not a name file
ENAVAIL	138	Not available
EISNAM	139	Is a name file
EREMOTEIO	140	Remote I/O error
EINIT	141	(Reserved for future use)
EREMDEV	142	Error 142
ENOTEMPTY	145	Directory not empty
ELOOP	150	Too many symbolic links in path
ESTALE	151	(Reserved for future use)
EIORESID	500	(Reserved for future use)

R7 Terminal Clients and R6 OpenAir Servers

Introduction

When using R7 terminal clients in server mode and connecting to an R6 OpenAir server release, there will be some features of the R7 client that you will not be able to use with the stock R6 server code.

The following documents show how some of these features can be used with R6 servers. Once R7 servers are available, it's recommended that a system upgrade be done rather than the procedures detailed in this document.

These patch releases are free and provided "As-Is" with no warranty. Support will be provided only on those systems covered by an active PowerNet ASSIST agreement.

Resolution

A variety of terminals, both DOS and CE based, now support variable sized fonts. The support for fonts under R6 only supports Double High and Double Wide settings. This procedure will detail how to add variable font support to R6 releases.

Warning: These patches and instructions require the user have an understanding of Linux and UNIX systems. Please do not attempt these procedures if you do not feel comfortable with this.

1. Check that the system is R6 by running the Utilities/Verify routine from the main OpenAir Menu.
2. Make sure the system is not in use and shut down the OpenAir server.
3. Save off the old programs that you will update.

as root:

```
# cd /home/crf
# mv rfvt rfvt.org
# mv 5250tn 5250.org
# mv 3270tn 3270tn.org
```

4. Copy in updated programs from this link (www.connectrf.com/Documents/patch-r6.zip) and make them executable.

```
# chmod 777 rfvt 3270tn 5250tn
```

5. Start the OpenAir server and test functionality. Please exercise the application thoroughly to make sure the updated code does not affect existing functionality.

If it does affect existing functionality, back out the patches by deleting the upgraded files and moving the original files back into use.

If there are no problems, proceed to the next step.

6. Create a configuration file to add the variable font settings. If the host list menu name is AIX-host, the file to create is titled AIX-host.cfm. Linux is case-sensitive, so make sure to match the case exactly.

Add these 2 lines and save:

```
font_size=0;  
font_bold=0;
```

Use the TC Manager to find supported values for each of these settings by selecting the target terminal and checking the values in the pull down menus.

7. Log terminal in and test. You can edit the file to change the values, but you must then log the terminal off then back on in order for the changes to take effect.

Linger Issue on IX Hosts Produces Waiting for Data/Host Process Delay

The TIME_WAIT setting if set improperly will cause a "Host Process delay" AKA "Waiting for DATA" in IX environments. The primary reason for this is that the sessions are not released by the IX host. This eventually uses up all the available user logons. Once those are exhausted, the subsequent attempts to login are blocked by the IX host resulting in the above user messages.

On a busy web server, many sockets may linger in the TIME_WAIT state. This is caused by improperly coded applications that do not properly shut down a socket. This can also be used as a type of DDoS attack.

A. AIX

No tuning recommendations.

B. Solaris

```
/usr/sbin/ndd -set /dev/tcp tcp_time_wait_interval 60000
```

This parameter affects the amount of time a TCP socket will remain in the TIME_WAIT state. The default is quite high for a busy web server, so it should be lowered to 60000 milliseconds (60 seconds). The parameter name was corrected in Solaris 7 and higher. Prior to Solaris 7, the parameter was incorrectly labeled as tcp_close_wait_interval.

C. Tru64 UNIX

No tuning recommendations.

D. HP-UX

```
ndd -set /dev/tcp tcp_time_wait_interval 60000
```

Sockets will linger in TIME_WAIT state no more than 60 seconds.

Note: Experience in Red Prairie/HPUX environments sometimes this setting is set to 0. In which case the session never times out.

E. Linux kernel 2.2

```
/sbin/sysctl -w net.ipv4.vs.timeout_timewait=60
```

Sockets will linger in TIME_WAIT state for no more than 60 seconds.

F. FreeBSD

No tuning recommendations.

Restarting APBASE when Host Sessions Fail to End

Introduction

On some host systems when backup and system maintenance is done, the telnet servers may not close the active telnet sessions. This results in the RF terminal's inability to get a new session with the host.

Resolution

Log in to the Linux system, and become root. Then set your terminal type:

```
export TERM=vt100
```

Place the following shell scripts in the /home/crf directory.

<http://www.connectrf.com/Documents/APBNC.SH>

<http://www.connectrf.com/Documents/APRST.SH>

<http://www.connectrf.com/Documents/SWRST.SH>

Then run the following commands in the /home/crf directory as su(root):

```
chown root *.sh  
chmod 755 *.sh
```

The first command changes the owner of the file. The second command makes the commands executable.

At this point, the shells are in place, and you will need to add a crontab entry as root. Add the line that is in **red** below. To make this run, remove the "#" character and exit. You will be editing with vi sequences.

```
crontab -e
```

```
# DO NOT EDIT THIS FILE. Edit the master and reinstall.  
# (/tmp/crontab.XXXa24226 installed on Tue Aug 24 10:59:26 2004)  
# (Cron version -- $Id: crontab.c,v 2.13 1994/01/17 03:20:37 vixie Exp $)  
# DO NOT EDIT THIS FILE. Edit the master and reinstall.  
# (/tmp/cron.root installed on Mon Jun 7 08:19:51 1999)  
# (Cron version -- $Id: crontab.c,v 2.13 1994/01/17 03:20:37 vixie Exp $)  
#0 2 * * * /home/crf/APBNC.sh > /dev/null 2>&1  
0 * * * * /home/crf/CRON_H.sh  
15 23 * * * /home/crf/CRON_D.sh  
30 23 * * 0 /home/crf/CRON_W.sh  
45 23 1 * 0 /home/crf/CRON_M.sh
```

The line above will shut down APBASE at 2:00 am, and then start it again every day, every month, and every week day.

Overview of the Cron Daemon

The cron daemon is the mechanism by which all timed events are initiated. It is executed upon system initialization and remains active while the system is operating in multi-user mode. Cron wakes up every minute and examines all the stored configuration files, called crontabs, to check each for commands that may be scheduled to be executed at the current time. Some systems have limits to the number of tasks that can be scheduled during the one minute time period. Most notably, because of the low number, is SGI's IRIX 5.3, which has a limit of 25 jobs.

Besides starting commands each minute, some cron daemons also check to see if its spool directory's last modified time has been updated. If it has, cron will check the modification time on all crontabs and reread the ones that have been modified. Other cron daemons examine new crontab files when first initialized and when the commands *crontab* or *at* are executed. This reduces the overhead of checking for new or changed files at regularly scheduled intervals.

Cron searches the crontab spool directory for crontab files. These files are named after user accounts. For instance, if the system administrator logged into the root account creates a crontab file, it will be named root and will be placed in the crontab spool directory. If Bill Wilson, whose username is bill, creates a crontab file, it is named bill in the crontab spool directory. When executing commands, any output is mailed to the owner of the crontab (or to the user named in the MAILTO environment variable in the crontab, if such exists).

The configuration files used to control the operation of cron are called crontab files or cron tables. These files contain information about the time, date, and command to execute. Different versions of UNIX store cron and support files in different locations and may use a slightly different format.

Linux

<code>/var/spool/cron</code>	main cron directory
<code>/var/spool/cron/cron.allow</code>	grant access to the cron facility
<code>/var/spool/cron/cron.deny</code>	revoke access to the cron facility

The Cron Table, Crontab, File Format

Cron table files, or crontabs, are text files, which direct the cron daemon's activity. Each line or entry has six fields, which are separated by space characters. The first five fields instruct the cron daemon as to when to execute the command, which is contained in the sixth field.

<u>Field</u>	<u>Value</u>
minute	00 to 59
hour	00 to 23 (military time)
day	1 to 31
month	1 to 12
weekday	0 to 6 (0=Sunday)

The first five fields can also use any one of the following formats.

An asterisk character (*) to match all values.

A single integer that matches that exact value.

A comma separated list of integers, like 1,3,5 to match one of the listed values.

A range of integers separated by a dash, like 4-6, to match the values within the range.

An asterisk slash integer (*/5) every 5 seconds, minutes, days, months or years.

Here are sample entries along with a short explanation of when the operation will be performed.

```
0 * * * * echo "WAKE UP" 2>&1 /dev/console
```

This entry sends the string WAKE UP to the device /dev/console at the start of every hour on every day of every month.

```
0 0 * * * calendar -
```

This entry runs the command *calendar*, which reminds users of holidays and other events at the start of the hour at the start of the day on every day of the month.

```
10,20,30,40,50 * * * * /adm/checkdaemon 2>&1 | /bin/mail -s "CRON:Check" root
```

This entry runs the command *checkdaemon* and mails the output of the command to root. The command is run 10, 20, 30, 40, and 50 minutes after the hour on every day of every month.

Which of these is the correct definition for this entry?

```
15 5 * * 1 find /var/preserve/. -mtime +7 -exec rm -f {} \;
```

The *Crontab* Command

The *crontab* files are not generated by editing the *crontab* file in the *crontab* spool directory. Instead, the *crontab* command is used to edit, list, create, or remove a *crontab* file for a user. The *crontab* command can be used by all the users on a system to create personal *crontab* as well as by the root account. Users are not allowed to view, edit, or create *crontab* files for other users.

Additionally, the use of *cron* can be denied to users. This is done to prevent system unfriendly, or security compromising tasks. When the *crontab* command is invoked, it examines the files *cron.deny* and *cron.allow* in the system's *cron* directory to grant or revoke the modification of the *crontab* spool file. If a username appears in the file *cron.allow*, the *crontab* command may be executed. If that file does not exist and the user's name does not appear in the *cron.deny* file, then *cron* can be used. If only an empty *cron.deny* exists, all users can use *cron*. If neither of these files exist, then only the root user can use *cron*.

The *crontab* command without options reads from standard input, so when executed, it takes the information entered at the keyboard as input. This makes it easy to remove the existing *crontab* without trying. If the *crontab* is run without options, it should be exited with a "Control C" so that the existing *crontab* is unmodified. Entering a "Control D" will cause the current user's *crontab* to be replaced with no information, thereby erasing the existing *crontab*.

The edit option *crontab -e* for the *crontab* command copies or creates the current user's *crontab* file. After editing is complete, the file is installed as the user's *crontab* file in the *crontab* spool directory. The default editor used by this command is *ed*. To specify an alternative, set the environment variable *EDITOR*. Not all systems' *crontab* have an edit option. If not, a file containing the *crontab* information can be created and read by the *crontab* command.

The list option, *crontab -l*, displays the contents of the current user's *crontab* file.

The remove option, *crontab -r*, empties the contents of the current user's *crontab* file.

The *crontab* command will accept an account name as the first argument if current user has superuser privileges.

Here is a sample session that adds a crontab entry for the current user, lists the crontab entry, and then removes it.

```
#crontab -e          (Create the crontab entry. Within an editor, enter:)
```

```
1 * * * * /usr/local/bin/runreport
```

```
# crontab -l        (List the user's crontab file.)
```

```
1 * * * * /usr/local/bin/runreport
```

```
# crontab -r        (Remove the user's crontab file.)
```

Using the *crontab* command without options to create the crontab file can be done by creating and editing a file, in this example, *allcron*.

```
#vi allcron  
#crontab allcron
```

Linux Console Error Message About Unmounting

Introduction

The following is information on a Linux console error message about unmounting. Every week on some system, a background task is run to mirror the first hard drive to a second hard drive. In doing so, it sets off a counter, which is defaulted to 25, that produces a console warning message about unmounting and running e2fsck.

Resolution

To correct this issue, run the following commands as root:

```
tune2fs -c 0 /dev/hda2
tune2fs -c 0 /dev/hda3
mount /dev/hdb2 /mnt
tune2fs -c 0 /dev/hdb2
umount /mnt
mount /dev/hdb3 /mnt
tune2fs -c 0 /dev/hdb3
umount /mnt
```

Then from the console when the system is idle as root run:

```
init 1
e2fsck -y /dev/hda2
e2fsck -y /dev/hda3
init 0
```

Links to command descriptions:

<http://www.die.net/doc/linux/man/man8/tune2fs.8.html>

<http://www.die.net/doc/linux/man/man8/e2fsck.8.html>

Linux Consuming Memory

Introduction

If Linux is left idle, it consumes RAM. This is normal and should be expected.

Resolution

Linux Support @ CNET: "There is no memory leak - the kernel functions perfectly normally.

There is no sense for the kernel to keep some RAM totally unused. So, if there is no other sensible use, the system keeps disk buffer cache in RAM for as much as there is space. The cache speeds up disk accesses if the access is for sectors already in RAM. The kernel kicks extra sectors out of the cache as soon as there is some better use for the RAM space"

As this indicates, Linux grabs free memory over time for caching, but gives it back when needed. As an example, our primary Linux server has been up for 38 days, and shows 2020K free. By telneting in a number of times and running an application (top), I can make the free space go down to 500K. Beyond that, it does not go down any further and actually goes up some, as the system gives back some of the memory it borrowed. As I then close my telnet sessions, the free memory goes back up beyond where it was, so I am at 5424K free now.

This is not something most other UNIX systems do, so it is not surprising that it was reported as problem. However, it is normal operation for Linux, and can be ignored.

Changing Linux Login To Prompt For Terminal Type

When logging into Linux over a telnet connection, you must login as ops first and then su over to root - especially if you are using a vt100 telnet session. If you modify ops's .bash_profile by adding the eval statement shown below, the system prompts for your term type. - From Kevin Lollock

```
PS_PERSONALITY=POSIX;export PS_PERSONALITY
PATH="/usr/local/bin:/usr/bin:/bin:/sbin."; export PATH
eval `tset -m :?${TERM:-ansi} -r -s -Q`
ulimit -c unlimited
umask 022
```

About This Document

This document is based on the following Technical Documents in our Notes Database that have been made obsolete: A1014, A1017, A1102, T1110, T1112, T1115, T1148, T1150, T1153, T1169, T1206, T1207, T1230, T1232, and T1240.

Please let us know about any errors in this document at:
<http://207.241.78.223/isoxpert/calltrak.nsf/WebTracking?OpenForm>.