How to Install the Apager Telnetd Server

Contents

OVERVIEW

This document describes how to hack telnetd to support the 'apager' functionality.

DESIGN

Typically, the super-server (*inetd* or *xinetd*) launches *in.telnetd* whenever it receives a TCP SYN on port 23. *in.telnetd* negotiates various telnet-protocol-specific parameters with the incoming client and then, typically, launches */bin/login*. At that point, *in.telnetd's* function is to act as a pipe, accepting characters from the client, forwarding them to /bin/login, and vice versa (accepting characters from */bin/login* and forwarding them to the telnet client). Typically, */bin/login* asks for a username & password, authenticates these against /etc/passwd and, if successful, 'execs' */bin/sh* (or */bin/bash* or whatever shell the user has specified in /etc/passwd). in.telnetd continues to act as a pipe, forwarding characters between whatever process has replaced */bin/login* (*/bin/sh*, */bin/bash*, */bin/csh*) and the telnet client.

In summary, *telnetd* is a daemon which provides 'pipe' functionality to a telnet client, forwarding packets between the client and some process running on the box hosting *telnetd*.

To provide support for 'telnet apager.fhcrc.org', we hack *in.telnetd*, such that it launches */opt/vdops/script/apager.login* instead of */bin/login*.

apager.login does *not* behave like */bin/login* does. Notably, it does not ask for a username and password. Instead, it asks for a Recipient and a Msg. Having received those, it does not launch */bin/sh* or any other shell; rather, it terminates, abruptly closing the telnet session.

/opt/vdops/script/apager.login is a Perl script which asks for Recipient and Msg and, once it has them, turns around and employs *qpage* to submit an alpha page to a paging queue. If *qpage* is running, *qpage* accepts this message, turns around, dials a local modem, and sends the page.

Notice how generic *in.telnetd* is: it doesn't care what characters are passing across the pipe it maintains, nor does it care what program sends those characters, nor what program receives them, so long as those characters arrive wrapped in the appropriate telnet-protocol-defined frames.

HACK TELNETD

Install the source RPM

Acquire the RPM from a source repository

- ftp://ftp.muug.mb.ca/mirror/centos/5.2/os/i386/CentOS/telnet-0.17-39.el5.i386.rpm
- http://www.mirrorservice.org/sites/mirror.centos.org/5/updates/SRPMS/telnet-0.17-39.el5.src.rpm

```
guru> wget
http://www.mirrorservice.org/sites/mirror.centos.org/5/updates/SRPM
S/telnet-0.17-39.el5.src.rpm
--14:30:23--
http://www.mirrorservice.org/sites/mirror.centos.org/5/updates/SRP
MS/telnet-0.17-39.el5.src.rpm
Resolving www.mirrorservice.org... 212.219.56.133, 212.219.56.134,
212.219.56.13
5, ...
Connecting to www.mirrorservice.org|212.219.56.133|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 279287 (273K) [application/x-redhat-package-manager]
Saving to: `telnet-0.17-39.el5.src.rpm'

100%[======================================>] 279,287      234K/s   in 1.2s

14:30:40 (234 KB/s) - `telnet-0.17-39.el5.src.rpm' saved [279287/279287]
guru> ls | grep telnet
telnet-0.17-39.el5.src.rpm
guru>
```

Become root

```
guru> su -
Password:
guru#
```

Create the pseudo-user 'mockbuild'

```
guru# groupadd -g 97 mockbuild
guru# useradd -u 97 -g 97 -m mockbuild
guru#
```

Install this RPM

```
guru# rpm -i telnet-0.17-39.el5.src.rpm
guru#
```

Build the src files

```
guru# cd /usr/src/redhat/SPECS
guru# ls
telnet.spec
guru#
[root@guru SPECS]# rpmbuild -bp --target=`uname -m` telnet.spec
Building target platforms: i686
Building for target i686
Executing(%prep): /bin/sh -e /var/tmp/rpm-tmp.77884
+ umask 022
+ cd /usr/src/redhat/BUILD
+ LANG=C
+ export LANG
+ unset DISPLAY
```

```
+ cd /usr/src/redhat/BUILD
+ rm -rf netkit-telnet-0.17
+ /bin/gzip -dc /usr/src/redhat/SOURCES/netkit-telnet-0.17.tar.gz
+ tar -xf -
+ STATUS=0
+ '[' 0 -ne 0 ']'
+ cd netkit-telnet-0.17
++ /usr/bin/id -u
+ '[' 0 = 0 ']'
+ /bin/chown -Rhf root .
++ /usr/bin/id -u
+ '[' 0 = 0 ']'
+ /bin/chgrp -Rhf root .
+ /bin/chmod -Rf a+rX,u+w,g-w,o-w .
+ mv telnet telnet-NETKIT
+ cd /usr/src/redhat/BUILD
+ cd netkit-telnet-0.17
+ /bin/gzip -dc /usr/src/redhat/SOURCES/telnet-client.tar.gz
+ tar -xf -
+ STATUS=0
+ '[' 0 -ne 0 ']'
++ /usr/bin/id -u
+ '[' 0 = 0 ']'
+ /bin/chown -Rhf root .
++ /usr/bin/id -u
+ '[' 0 = 0 ']'
+ /bin/chgrp -Rhf root .
+ /bin/chmod -Rf a+rX,u+w,g-w,o-w .
+ echo 'Patch #1 (telnet-client-cvs.patch):'
Patch #1 (telnet-client-cvs.patch):
+ patch -p0 -b --suffix .cvs -s
+ echo 'Patch #5 (telnetd-0.17.diff):'
Patch #5 (telnetd-0.17.diff):
+ patch -p0 -b --suffix .fix -s
+ echo 'Patch #6 (telnet-0.17-env.patch):'
Patch #6 (telnet-0.17-env.patch):
+ patch -p1 -b --suffix .env -s
+ echo 'Patch #10 (telnet-0.17-pek.patch):'
Patch #10 (telnet-0.17-pek.patch):
+ patch -p0 -b --suffix .pek -s
+ echo 'Patch #7 (telnet-0.17-issue.patch):'
Patch #7 (telnet-0.17-issue.patch):
+ patch -p1 -b --suffix .issue -s
+ echo 'Patch #8 (telnet-0.17-sa-01-49.patch):'
Patch #8 (telnet-0.17-sa-01-49.patch):
+ patch -p1 -b --suffix .sa-01-49 -s
+ echo 'Patch #11 (telnet-0.17-8bit.patch):'
Patch #11 (telnet-0.17-8bit.patch):
+ patch -p1 -b --suffix .8bit -s
+ echo 'Patch #12 (telnet-0.17-argv.patch):'
Patch #12 (telnet-0.17-argv.patch):
+ patch -p1 -b --suffix .argv -s
+ echo 'Patch #13 (telnet-0.17-conf.patch):'
Patch #13 (telnet-0.17-conf.patch):
```

```
+ patch -p1 -b --suffix .confverb -s
+ echo 'Patch #14 (telnet-0.17-cleanup_race.patch):'
Patch #14 (telnet-0.17-cleanup_race.patch):
+ patch -p1 -b --suffix .cleanup_race -s
+ echo 'Patch #15 (telnetd-0.17-pty_read.patch):'
Patch #15 (telnetd-0.17-pty_read.patch):
+ patch -p0 -b --suffix .pty_read -s
+ echo 'Patch #16 (telnet-0.17-CAN-2005-468_469.patch):'
Patch #16 (telnet-0.17-CAN-2005-468_469.patch):
+ patch -p1 -b --suffix .CAN-2005-468_469 -s
+ echo 'Patch #18 (telnet-gethostbyname.patch):'
Patch #18 (telnet-gethostbyname.patch):
+ patch -p1 -b --suffix .gethost -s
+ echo 'Patch #19 (netkit-telnet-0.17-ipv6.diff):'
Patch #19 (netkit-telnet-0.17-ipv6.diff):
+ patch -p1 -b --suffix .gethost -s
+ echo 'Patch #20 (telnet-0.17-errno_test.patch):'
Patch #20 (telnet-0.17-errno_test.patch):
+ patch -p1 -b --suffix .errno_test -s
+ echo 'Patch #21 (netkit-telnet-0.17-nodns.patch):'
Patch #21 (netkit-telnet-0.17-nodns.patch):
+ patch -p1 -b --suffix .dns -s
+ exit 0
guru#

guru# cd /usr/src/redhat/BUILD/netkit-telnet-0.17/telnetd
guru# ls
authenc.c          Makefile     telnetd.c
defs.h             pathnames.h          telnetd.c.cleanup_race
defs.h.fix         setproctitle.3       telnetd.c.dns
ext.h              setproctitle.c       telnetd.c.fix
ext.h.sa-01-49     setproctitle.c.argv  telnetd.c.gethost
getent.c           setproctitle.h       telnetd.c.pty_read
global.c           slc.c        telnetd.h
issue.net.5        slc.c.sa-01-49       termstat.c
issue.net.5.issue  state.c      termstat.c.sa-01-49
login.3            state.c.sa-01-49     utility.c
logout.h           sys_term.c   utility.c.issue
logwtmp.h          telnetd.8    utility.c.sa-01-49
guru#
```

Modify Source

In this step, we change telnetd's source to instruct it to load */opt/vdops/script/apager.login* rather than */bin/login*.

Save Originals

```
guru# cp sys_term.c sys_term.c.ori
guru# cp pathnames.h pathnames.h.ori
guru#
```

Edit pathnames.h

[Obvious what we are doing here.]

Change

```
#ifndef _PATH_LOGIN
#define _PATH_LOGIN  "/bin/login"
#endif
```

to

```
#ifndef _PATH_LOGIN
#define _PATH_LOGIN  "/opt/vdops/script/apager.login"
#endif
```

Edit sys_term.c

[I forget why this was necessary -- someone at SuSE tried to explain it to me, but I didn't understand.] Go to line 630.

Change

```
  addarg (&avs, loginprg);
  addarg (&avs, "-h");
  addarg (&avs, host);
#if !defined(NO_LOGIN_P)
  addarg (&avs, "-p");
#endif
```

to

```
  /* Commented out following section 2008-04-18 --sk
  addarg (&avs, loginprg);
  addarg (&avs, "-h");
  addarg (&avs, host);
#if !defined(NO_LOGIN_P)
  addarg (&avs, "-p");
#endif
  */
```

Compile

```
guru# cd /usr/src/redhat/BUILD/netkit-telnet-0.17
guru# make
Directories: /usr/bin /usr/sbin /usr/man
Looking for a C compiler... gcc
Checking if gcc accepts gcc warnings... yes
Looking for a C++ compiler... gcc
Checking if gcc accepts gcc warnings... yes
Checking if gcc accepts -O2... yes
Checking if gcc accepts -fno-rtti... yes
Checking if gcc accepts -fno-exceptions... yes
Checking for BSD signal semantics... yes
```

```
Checking for ncurses... yes
Checking for GNU libc... yes
Checking for forkpty... -lutil
Checking for logwtmp... -lutil
Checking for snprintf declaration... ok
Checking for snprintf implementation... ok
Generating MCONFIG...
guru# make
[...]
gcc  telnetd.o state.o termstat.o slc.o sys_term.o utility.o global.o
setproctitle.o -lutil -lutil -o telnetd
make[1]: Leaving directory `/usr/src/redhat/BUILD/netkit-telnet-0.17/telnetd'
guru#
```

INSTALL

Copy the binary

```
guru# cp /usr/sbin/in.telnetd /usr/sbin/in.telnetd.ori
guru# cd /usr/src/redhat/BUILD/netkit-telnet-0.17/telnetd/telnetd
guru# cp telnetd /usr/sbin/in.telnetd
guru#
```

Load xinetd at boot

```
chkconfig --add xinetd
```

Enable telnetd

Verify that /etc/xinetd.d/telnet something like this. The key item is the "disable = no" phrase.

```
# default: on
# description: The telnet server serves telnet sessions; it uses \
#       unencrypted username/password pairs for authentication.
service telnet
{
        disable = no
        flags           = REUSE
        socket_type     = stream
        wait            = no
        user            = root
        server          = /usr/sbin/in.telnetd
        log_on_failure  += USERID
}
```

Start xinetd

```
guru# /etc/init.d/xinetd start
```

Review syslog for errors

Copy the apager.login program

Grab apager.login from somewhere. The version in the Appendix is for demonstrative purposes only; it may not be the latest.

```
guru> sudo su netops
[...]
guru>
guru> cp /home/skendric/apager.login /opt/vdops/script/apager.login
guru> chown netops:vdops /opt/vdops/script/apager.login
guru> chmod 555 /opt/vdops/script/apager.login

guru> perl -c /opt/vdops/script/apager.login
"-T" is on the #! line, it must also be used on the command line at
/opt/vdops/script/apager.login line 1.
guru>
```

TEST

Watch syslog for errors

```
guru> telnet guru
Trying 140.107.74.123...
Connected to guru (140.107.74.123).
Escape character is '^]'.
CentOS release 5 (Final)
Kernel 2.6.18-53.1.14.el5 on an i686
Recipient: skendric
Msg: can i send myself a page?
Connection closed by foreign host.
guru>
```

To debug, kill qpage and then reload it using /etc/init.d/qpage-debug -- this dumps debug messages to your terminal session. Don't forget to kill qpage-debug and restart the regular qpage when you are done.

```
guru> sudo /etc/init.d/qpage stop
guru> sudo /etc/init.d/qpage-debug start
[... analyze issues ...]
guru> sudo /etc/init.d/qpage-debug stop
guru> sudo /etc/init.d/qpage start
```

APPENDIX

apager.login source

```
#!/usr/bin/perl -T

# Submit a page to apager via the apager login

# V      Who       When        What
# ------------------------------------------------------------------------
# 2.5.3  skendric  2008-05-12  Syntax clean-up
```

```
# 2.5.2  skendric  2006-10-12  Fiddle with logged msg
# 2.5.1  skendric  2005-06-10  Fiddled with logging msg
# 2.5.0  skendric  2004-02-16  Added security checks
# 2.4.0  skendric  2002-05-29  Log 'duty' pages to syslog
# 2.3.0  skendric  2002-01-24  Disabled logging, $max to 500
# 2.2.0  skendric  2002-04-22  Added logging
# 2.1.0  skendric  1999-12-15  Increased $max from 240 to 400
# 2.0.0  pryan     1997-07-21  Uses qpage instead of tpage
# 1.1.0  Ron Hood  1996-03-12  Added logging
# 1.0.0  Ron Hood  1994-05-19  First version

# =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
#       Header junk
# =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=


# Invoke modules
use strict;
use warnings;
use English;
use List::MoreUtils qw(any);
use Sys::Syslog;

# Declare variables
my $gid;                                # GID to become
my $max;                                # Ceiling on message length
my $msg;                                # Mesage to send
my $qpage;                              # Location of qpage binary
my $recip;                              # String of page recipients
my $stty;                               # Location of stty binary
my $uid;                                # UID to become
my @watched;                            # List of recipients whose messages
                                        # I log

# Define variables
$OUTPUT_AUTOFLUSH = 1;
$gid              = 101;                # vdops
$max              = 500;              # Maximum message size
$qpage            = '/usr/local/bin/qpage';
$stty             = '/bin/stty';
$uid              = 9383;               # netops
@watched          = qw/duty vdops/;

# Drop privileges
($GID, $EGID) = ($gid, $gid);
($UID, $EUID) = ($uid, $uid);

# Clear PATH to satisfy taint checks
$ENV{PATH} = "";

# Set backspace character
system ("$stty erase ^H");



# =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
#      Do the work
# =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=

# Accept recipient
print 'Recipient: ';
chomp($recip = <>);
$recip = lc $recip;

# If $recip is empty, die
if (not defined $recip or $recip eq "" or $recip =~ /^\s+$/) {
  die 'You must specify a recipient';
```

```
}

# Replace spaces with commas (qpage wants commas between recipients)
$recip =~ s/\s+/,/g;

# Untaint $recip, allowing only alphanumeric characters plus "-" and
# "," in the input
if ($recip =~ /^([-,\w]+)$/) {
    $recip = $1;
}
else {
  die 'Recipient names can only contain alphanumerics and dashes';
}

# Accept message
print 'Msg: ';
chomp($msg = <>);
$msg = substr $msg, 0, $max;

# If $msg is empty, die
if (not defined $msg or $msg eq "" or $msg =~ /^\s+$/) {
  die 'You must specify a non-empty message';
}

# Untaint $msg (this is an evil way to accomplish this ... but i haven't
# thought of a restrictive regex to use here)
$msg = $1 if $msg =~ /(.+)/;

# Send message
system($qpage, "-f", "", "-p", $recip, $msg);

# Log message
log_it() if any { $_ eq $recip } @watched;


# =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
#       Log it
# =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
sub log_it {
  my $facility = "local0";
  my $ident = "apager->$recip";
  my $level = "info";

  openlog ($ident, '', $facility);
  syslog ($level, $msg);
  closelog ();
}
```