

Re: OpenSSH 3.8 Released

Source: <http://www.derkeiler.com/Newsgroups/comp.security.ssh/2004-02/0431.html>

From: Darren Tucker (dtucker_at_dodgy.net.au)

Date: 02/29/04

Date: Sun, 29 Feb 2004 07:29:31 +0000 (UTC)

In article <c1rhdu\$t2g\$1@usenet.cso.niu.edu>, Neil W Rickert <rickert+nn@cs.niu.edu> wrote:
[snip good info on keylogin]

Thanks.

>However, on a server machine, where root can access the shadow data,
>the login works without needing a keylogin. In that case, the
>PAM routines authenticate the user without doing a keylogin. The
>keylogin should be done in the later call to pam_setcred().
>However, this can only work if there is preserved state information
>available between the various pam calls. Apparently this state
>information is not preserved unless sshd is built with
>"USE_POSIX_THREADS" defined.

Here's where it comes off the rails: for various reasons, in sshd the actual PAM authentication is done in an authentication "thread" that is normally a child process of sshd, which exits immediately after the authentication.

Now PAM modules have a few ways of passing information around: regular environment variables (eg "KRB5CCNAME", but those are visible to any other process on the system, so obviously can't be used for confidential data), PAM environment variables (pam_putenv/pam_getenv), and a private stash of per-module state (pam_set_data/pam_put_data).

Currently, sshd exports the first two but not the last one and, from the sound of it, the AFS and NIS+ modules use pam_set_data for their state.

When you compile with USE_POSIX_THREADS, the regular sshd and the child doing the authentication share the same address space so no export is needed and the state set by pam_set_data does not vanish when the authentication "thread" exits. This is then inherited when sshd forks to run the user's shell, and the calls to pam_setcred can find whatever was stashed by pam_set_data during the authentication.

Anyway, there's something of a flamewar going on over on the mailing list over PAM and threads right now, so I won't rehash the pros and

comp.security.ssh: Re: OpenSSH 3.8 Released

cons here, just redirect interested readers to:

<http://marc.theaimsgroup.com/?l=openssh-unix-dev&m=107791415130053>

(some of the posts are multipart/signed, but you should be able to get the gist...)

[snip]

>>*Does the configure option "--with-rpath" make a difference?*

>

>*Yes, that worked.*

>

>*I hadn't tried it before, since the output of "./configure --help"*

>*listed only "--without-rpath" and suggested that "--with-rpath" was*

>*the default.*

Yep, so either configure or the help is wrong. We just need to decide which and change it. (At first glance it looks like the the help is right and configure is wrong.)

--

Darren Tucker (dtucker at zip.com.au)

GPG key 8FF4FA69 / D9A3 86E9 7EEE AF4B B2D4 37C9 C982 80C7 8FF4 FA69

Good judgement comes with experience. Unfortunately, the experience usually comes from bad judgement.