

[NEWS] OpenSSH Challenge–Response Buffer Overflow (Update)

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2002-07/0011.html>

From: support@securiteam.com

Date: 07/06/02

From: support@securiteam.com

To: list@securiteam.com

Date: Sat, 6 Jul 2002 08:52:16 +0200 (CEST)

The following security advisory is sent to the securiteam mailing list, and can be found at the SecuriTeam web site: <http://www.securiteam.com>

-- promotion

When was the last time you checked your server's security?

How about a monthly report?

<http://www.AutomatedScanning.com> – Know that you're safe.

OpenSSH Challenge–Response Buffer Overflow (Update)

SUMMARY

OpenSSH, a popular server utility that provides encrypted connections between hosts and is commonly used for administration and file transfer, contains a integer overflow, resulting in a heap overflow that could be exploited to execute arbitrary commands.

DETAILS

Versions Tested To Be Vulnerable:

OpenSSH versions prior to 3.4

Impact:

A local user may be able to execute arbitrary commands as the user that the OpenSSH daemon is running as prior to authentication. This is normally root.

Description:

It is the current belief of many that exploiting the recently disclosed vulnerabilities in OpenSSH's challenge–response routines is reliant upon a system's use of BSD's authentication mechanisms and therefore restricts the platforms on which this vulnerability may be exploited.

Securiteam: [NEWS] OpenSSH Challenge–Response Buffer Overflow (Update)

This is almost certainly due to various advisories posted to various formats by unnamed security companies.

Although it is widely known that all systems running versions of OpenSSH prior to 3.4 are affected by this vulnerability, many vendors have deemed their platforms invulnerable to exploitation.

In spite of this, our research has proven multiple platforms originally thought to be invulnerable to attack to be vulnerable.

As reported by GOBBLES [1], systems running vulnerable binaries, built with `--with-bsd-auth` at compile time are vulnerable to attack via an integer overflow in the `input_userauth_info_response()` function.

Conversely, under Linux and other platforms using a vulnerable version of OpenSSH compiled with `--with-pam`, the integer overflow lies in the function `input_userauth_info_response_pam()`.

In both cases, the final heap based buffer overflow is a result of the integer overflow of unsigned int `nresp`, calculated from `packet_get_int()`, the return value of `packet_get_int` being a client controlled integer.

Scope for attack:

- Because of the nature of the vulnerability, exploitation is possible before a user has authenticated with the remote host. This would potentially allow an attacker to remotely execute arbitrary commands as the UID of the daemon process, PRIOR TO AUTHENTICATION.

- To exploit the vulnerability described in the "Proof of concept" section of this advisory, the SSHd binary must have been compiled with PAM support.

Workaround:

Global InterSec recommends the following settings be disabled within SSHd's configuration. This is normally located at `/etc/ssh/sshd_config`

```
PAMAuthenticationViaKBDInt no
KbdInteractiveAuthentication no
```

However, we strongly recommend that all vulnerable binaries be upgraded as soon as possible. (See vendor solutions.)

Vendor Solutions:

Since the original disclosure by ISS [3], vendors have released their own advisories, with distribution specific fixes. A list of some of these follows.

Mandrake Secure Linux:

<http://www.mandrakesecure.net/en/advisories/2002/MDKSA-2002-040-1.php>
<http://www.mandrakesecure.net/en/advisories/2002/MDKSA-2002-040-1.php>

Securiteam: [NEWS] OpenSSH Challenge–Response Buffer Overflow (Update)

SuSE Linux:

<http://www.suse.de/de/support/security/2002_024_openssh_txt.html>
http://www.suse.de/de/support/security/2002_024_openssh_txt.html

EnGarde Secure Linux:

<http://www.linuxsecurity.com/advisories/other_advisory-2177.html>
http://www.linuxsecurity.com/advisories/other_advisory-2177.html

Conectiva Linux:

<<http://distro.conectiva.com.br/atualizacoes/?id=aa>>>
<http://distro.conectiva.com.br/atualizacoes/?id=aa>

Caldera Linux:

<<ftp://ftp.caldera.com/pub/security/OpenLinux/CSSA-2002-030.0.txt>>
<ftp://ftp.caldera.com/pub/security/OpenLinux/CSSA-2002-030.0.txt>

NetBSD:

<<ftp://ftp.netbsd.org/pub/NetBSD/security/advisories/NetBSD-SA2002-005.txt.asc>>
<ftp://ftp.netbsd.org/pub/NetBSD/security/advisories/NetBSD-SA2002-005.txt.asc>

Redhat:

<<http://rhn.redhat.com/errata/RHSA-2002-127.html>>
<http://rhn.redhat.com/errata/RHSA-2002-127.html>

Exploitation / Proof of concept:

On certain distributions of Linux, the evidence that this bug is exploitable may be more apparent than others due to the %edx register being overwritten. In either case, inspection that is slightly more careful confirms the possibility that this vulnerability could be exploited when compiled `--with-pam`.

By examining an assembler dump of `input_userauth_info_response_pam()`, we can see that the (corrupted?) %edx has been loaded from 0x8080130, where 0x8080130 is the location of the `context_pam2` structure.

```
0x80521f2 <input_userauth_info_response_pam+122>: mov
0x8080130,%edx
```

Note:

The above instruction to `mov 0x8080130 into %edx` occurs in preparation for the call to `xfree()` and after the call to `wrapped strdup()`; whilst the debugger back trace suggests that the `xfree()` [`free()`] was never called.

By allocating specific break points through out `input_userauth_info_response_pam()` and into the call to `free()`, it becomes apparent that the call to `free()` could be exploited:

```
Breakpoint 14, 0x0806c677 in xfree (ptr=0x808a380) at xmalloc.c:55
55 free(ptr);
(gdb) print 0x808a380
```

Securiteam: [NEWS] OpenSSH Challenge–Response Buffer Overflow (Update)

\$24 = 134783872
(gdb) x/10x 0x808a380
0x808a380: 0x41414141 0x41414141 0x41414141
0x41414141
0x808a390: 0x41414141 0x41414141 0x41414141
0x41414141
0x808a3a0: 0x41414141 0x41414141

From here on, exploitation becomes trivial. For more information on exploiting calls to free() see the excellent Phrack article "Once upon a free()" [2].

ADDITIONAL INFORMATION

References:

- [1] GOBBLES Security –
<<http://www.immunitysec.com/GOBBLES/exploits/sshutup-theo.tar.gz>>
<http://www.immunitysec.com/GOBBLES/exploits/sshutup-theo.tar.gz>
- [2] Phrack Magazine – Once Upon a free() –
<<http://www.phrack.com/show.php?p=57>>
<http://www.phrack.com/show.php?p=57>
- [3] ISS –
<<http://bvlive01.iss.net/issEn/delivery/xforce/alertdetail.jsp?oid=20584>>
<http://bvlive01.iss.net/issEn/delivery/xforce/alertdetail.jsp?oid=20584>

The information has been provided by <<mailto:lists@globalintersec.com>>
Global InterSec Research.

=====

This bulletin is sent to members of the SecuriTeam mailing list.
To unsubscribe from the list, send mail with an empty subject line and body to:
list-unsubscribe@securiteam.com
In order to subscribe to the mailing list, simply forward this email to: list-subscribe@securiteam.com

=====
=====

DISCLAIMER:

The information in this bulletin is provided "AS IS" without warranty of any kind.
In no event shall we be liable for any damages whatsoever including direct, indirect, incidental, consequential, loss of business profits or special damages.

-
- **Previous message:** support@securiteam.com: "[NT] Vulnerability Report for Inktomi Traffic Server"
 - **Messages sorted by:** [\[date \]](#) [\[thread \]](#) [\[subject \]](#) [\[author \]](#) [\[attachment \]](#)