Talos Vulnerability Report

TALOS-2020-1029

# atftpd daemon Denial of Service Vulnerability

AUGUST 26, 2020

### CVE NUMBER

CVE-2020-6097

### Summary

An exploitable denial of service vulnerability exists in the atftpd daemon functionality of atftp 0.7.git20120829-3.1+b1. A specially crafted sequence of RRQ-Multicast requests trigger an assert() call resulting in denial-of-service. An attacker can send a sequence of malicious packets to trigger this vulnerability.

### Tested Versions

atftp 0.7.git20120829-3.1+b1

### Product URLs

https://github.com/seveas/atftp

### CVSSv3 Score

7.5 - CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H

### CWE

CWE-617 - Reachable Assertion

### Details

atftp is an open source TFTP server implementation. The ‘a’ stands for “advanced”, because it’s intended to be fully compliant with all related RFCs including RFC1350, RFC2090, RFC2347, RFC2348 andRFC2349.

A remote attacker may send a sequence of crafted RRQ-Multicast requests to the atftpd, triggering an assert() call in the atftpd code which results in abort of atftpd.

The vulnerablility can be traced down to the function: sockaddr_print_addr within tftp_def.c. where an unexpected sockaddr_storage *ss data with “ss_family=AF_UNSPEC” reaches the assert() call in the ‘else’ branch (line #192).

The vulnerable code snippet (tftpd_file.c)

```
183 char *
184 sockaddr_print_addr(const struct sockaddr_storage *ss, char *buf, size_t len)
185 {
186      const void *addr;
187      if (ss->ss_family == AF_INET)
188           addr = &((const struct sockaddr_in *)ss)->sin_addr;
189      else if (ss->ss_family == AF_INET6)
190           addr = &((const struct sockaddr_in6 *)ss)->sin6_addr;
191      else
192           assert(!"sockaddr_print: unsupported address family");
193      return (char *)inet_ntop(ss->ss_family, addr, buf, len);
194 }
```

An instance of “sa_family=AF_UNSPEC” is also seen in the strace output below,

```
* strace -f -p <pid>

[pid 10723] <... select resumed> )      = 0 (Timeout)
[pid 10723] stat("/etc/localtime", {st_mode=S_IFREG|0644, st_size=3545, ...}) = 0
[pid 10723] uname({sysname="Linux", nodename="kali", ...}) = 0
[pid 10723] getpid()                    = 10654
[pid 10723] write(2, "Nov 13 17:25:02 kali atftpd[1065"..., 73) = 73
[pid 10723] stat("/etc/localtime", {st_mode=S_IFREG|0644, st_size=3545, ...}) = 0
[pid 10723] uname({sysname="Linux", nodename="kali", ...}) = 0
[pid 10723] getpid()                    = 10654
[pid 10723] write(2, "Nov 13 17:25:02 kali atftpd[1065"..., 96) = 96
[pid 10723] sendto(1, "\0\6multicast\000239.239.239.0,1758,1"..., 33, 0, {sa_family=AF_UNSPEC,
sa_data="\340\222\300\250X\201\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0
\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"}, 128) =
33
[pid 10723] select(1024, [1], NULL, NULL, {tv_sec=5, tv_usec=0} <unfinished ...>
[pid 10726] <... select resumed> )      = 0 (Timeout)
[pid 10726] stat("/etc/localtime", {st_mode=S_IFREG|0644, st_size=3545, ...}) = 0
[pid 10726] uname({sysname="Linux", nodename="kali", ...}) = 0
[pid 10726] getpid()                    = 10654
[pid 10726] write(2, "Nov 13 17:25:04 kali atftpd[1065"..., 73) = 73
[pid 10726] lseek(5, 0, SEEK_SET)       = 0
[pid 10726] read(5, "Hello World! Super Cool!\n", 4096) = 25
[pid 10726] read(5, "", 4096)           = 0
[pid 10726] sendto(4, "\0\3\0\1Hello World! Super Cool!\n", 29, 0, {sa_family=AF_INET, sin_port=htons(1758),
sin_addr=inet_addr("239.239.239.1")}, 128) = 29
[pid 10726] stat("/etc/localtime", {st_mode=S_IFREG|0644, st_size=3545, ...}) = 0
[pid 10726] uname({sysname="Linux", nodename="kali", ...}) = 0
[pid 10726] getpid()                    = 10654
[pid 10726] write(2, "Nov 13 17:25:04 kali atftpd[1065"..., 82) = 82
[pid 10726] select(1024, [4], NULL, NULL, {tv_sec=5, tv_usec=0} <unfinished ...>
[pid 10723] <... select resumed> )      = 0 (Timeout)
[pid 10723] write(2, "atftpd: tftp_def.c:192: sockaddr"..., 111) = 111
[pid 10723] mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa0e4dd9000
[pid 10723] rt_sigprocmask(SIG_UNBLOCK, [ABRT], NULL, 8) = 0
[pid 10723] rt_sigprocmask(SIG_BLOCK, ~[RTMIN RT_1], [], 8) = 0
[pid 10723] getpid()                    = 10654
[pid 10723] gettid()                    = 10723
[pid 10723] tgkill(10654, 10723, SIGABRT) = 0
[pid 10723] rt_sigprocmask(SIG_SETMASK, [], NULL, 8) = 0
[pid 10723] --- SIGABRT {si_signo=SIGABRT, si_code=SI_TKILL, si_pid=10654, si_uid=65534} ---
[pid 10726] <... select resumed> <unfinished ...>) = ?
[pid 10654] <... select resumed> <unfinished ...>) = ?
[pid 10726] +++ killed by SIGABRT +++
[pid 10723] +++ killed by SIGABRT +++
+++ killed by SIGABRT +++
```

### Crash Information

Below is a backtrace when atftpd aborted under fuzzing test,

```
atftpd: tftp_def.c:192: sockaddr_print_addr: Assertion `!"sockaddr_print: unsupported address family"' failed.

Thread 78 "atftpd" received signal SIGABRT, Aborted.
[Switching to Thread 0x7ffff6d09700 (LWP 10623)]
__GI_raise (sig=sig@entry=6) at ../sysdeps/unix/sysv/linux/raise.c:50
50      ../sysdeps/unix/sysv/linux/raise.c: No such file or directory.
(gdb) bt
#0  __GI_raise (sig=sig@entry=6) at ../sysdeps/unix/sysv/linux/raise.c:50
#1  0x00007ffff7d7b535 in __GI_abort () at abort.c:79
#2  0x00007ffff7d7b40f in __assert_fail_base (fmt=0x7ffff7edd710 "%s%s%s:%u: %s%sAssertion `%s' failed.\n%n",
assertion=0x5555555636f0 "!\"sockaddr_print: unsupported address family\"", file=0x555555563608 "tftp_def.c", line=192, function=<optimized
out>) at assert.c:92
#3  0x00007ffff7d88b92 in __GI___assert_fail (assertion=assertion@entry=0x5555555636f0 "!\"sockaddr_print: unsupported address family\"",
file=file@entry=0x555555563608 "tftp_def.c", line=line@entry=192, function=function@entry=0x5555555638e0 <__PRETTY_FUNCTION__.4953>
"sockaddr_print_addr") at assert.c:101
#4  0x000055555555b2b7 in sockaddr_print_addr (ss=<optimized out>, buf=buf@entry=0x7ffff6d08a80 "", len=len@entry=46) at tftp_def.c:192
#5  0x000055555555d7f6 in tftpd_send_file (data=0x555555571930) at tftpd_file.c:784
#6  0x000055555555865d in tftpd_receive_request (arg=0x555555571930) at tftpd.c:751
#7  0x00007ffff7f92fb7 in start_thread (arg=<optimized out>) at pthread_create.c:486
#8  0x00007ffff7e502ef in clone () at ../sysdeps/unix/sysv/linux/x86_64/clone.S:95
```

### Timeline

2020-04-16 - Vendor Disclosure

2020-05-16 - 30 day follow up

2020-06-02 - 45+ day follow up

2020-06-30 - 60+ day follow up

2020-08-26 - Public Release

### CREDIT

Discovered by Peter Wang of Cisco ASIG.

VULNERABILITY REPORTS

PREVIOUS REPORT

NEXT REPORT

TALOS-2020-1030

TALOS-2020-1081