Common Desktop Environment 2.3.1 / 1.6 libDtSvc Buffer Overflow

Authored by Marco Ivaldi

Posted Apr 17, 2020

A difficult to exploit stack-based buffer overflow in the _DtCreateDtDirs() function in the Common Desktop Environment version distributed with Oracle Solaris 10 1/13 (Update 11) and earlier may allow local users to corrupt memory and potentially execute arbitrary code in order to escalate privileges via a long X11 display name. The vulnerable function is located in the libDtSvc library and can be reached by executing the setuid program dtsession. Versions 2.3.1 and below as well as 1.6 and earlier are affected.

tags | exploit, overflow, arbitrary, local
systems | solaris
advisories | CVE-2020-2851
SHA-256 | 7f50111057b19d6619dd24b1f2d5b993965259bb33db3ffa61cb8236878b3cc3

Download | Favorite | View

Related Files

Share This

Like Tweet LinkedIn Reddit Digg StumbleUpon

Change Mirror Download

```
@Mediaservice.net Security Advisory #2020-06 (last updated on 2020-04-15)

         Title:  Stack-based buffer overflow in CDE libDtSvc
   Application: Common Desktop Environment 2.3.1 and earlier
    Common Desktop Environment 1.6 and earlier2020-06-cde-libDtSvc.txt
     Platforms:  Oracle Solaris 10 1/13 (Update 11) and earlier
    Other platforms are potentially affected (see below)
   Description:  A difficult to exploit stack-based buffer overflow in the
    libDtSvc library distributed with CDE may allow local users to
    corrupt memory and potentially execute aritrary code in order
    to escalate privileges
        Author:  Marco Ivaldi <marco.ivaldi@mediaservice.net>
 Vendor Status:  Oracle <secalert_us@oracle.com> notified on 2019-12-15
    CERT/CC notified on 2019-12-15 (tracking VU#308289)
      CVE Name:  CVE-2020-2851
   CVSS Vector: CVSS:3.0/AV:L/AC:H/PR:L/UI:N/S:C/C:H/I:H/A:H (Base Score: 7.8)
    References: https://github.com/0xdea/advisories/blob/master/2020-06-cde-libDtSvc.txt
    https://www.oracle.com/security-alerts/cpuapr2020.html
    https://sourceforge.net/p/cdesktopenv/wiki/Home/
    https://www.oracle.com/technetwork/server-storage/solaris10/
    https://www.mediaservice.net/
    https://0xdeadbeef.info/

1. Abstract.

A difficult to exploit stack-based buffer overflow in the _DtCreateDtDirs()
function in the Common Desktop Environment version distributed with Oracle
Solaris 10 1/13 (Update 11) and earlier may allow local users to corrupt memory
and potentially execute arbitrary code in order to escalate privileges via a
long X11 display name. The vulnerable function is located in the libDtSvc
library and can be reached by executing the setuid program dtsession.

Note that Oracle Solaris CDE is based on the original CDE 1.x train, which is
different from the CDE 2.x codebase that was later open sourced. In detail, the
open source CDE is not affected by this specific vulnerability, but following
our report some additional work has been done by its maintainers to properly
check bounds in the libDtSvc library. Most notably, insecure calls to strncat()
that caused buffer overflows have been fixed.

2. Example Attack Session.

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

bash-3.2$ cat /etc/release
                    Oracle Solaris 10 1/13 s10x_u11wos_24a X86
  Copyright (c) 1983, 2013, Oracle and/or its affiliates. All rights reserved.
                            Assembled 17 January 2013
bash-3.2$ uname -a
SunOS nostalgia 5.10 Generic_147148-26 i86pc i386 i86pc
bash-3.2$ id
uid=54322(raptor) gid=1(other)
bash-3.2$ grep 10.0.0.24 /etc/hosts
10.0.0.24
aaaa:aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
[activate a valid display on 10.0.0.24:0]
/usr/dt/bin/dtsession -display
aaaa:aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa:0
Segmentation Fault

3. Discussion.

The overflow occurs in the following code snippet of Oracle Solaris CDE (the
Ghidra decompiler is probably doing something wrong as some variables seem to
overlap, however its output is good enough for the purpose of this discussion):

char * _DtCreateDtDirs(int param_1)
{
  ...
  char local_f0 [104];
  char local_88 [112];
  char *heap_path2;
  char *tmp_ptr1;
  char *home;
  undefined *local_c;
  undefined local_8 [4];
  ...
  if (param_1 != 0) {
    strcpy(local_f0,*(char **)(param_1 + 0x80));
    strcpy(local_88,*(char **)(param_1 + 0x80));
  ...
}

An X11 display data structure is passed to the _DtCreateDtDirs() function as
its only parameter (param_1 in the pseudocode above). It contains the X11
display name at offset 0x80. This display name is copied into the stack buffers
local_f0 and local_88 using the insecure function strcpy() twice, therefore two
overflows occur.

Based on the inferred stack layout, the following local variables are
overflowed into before the saved return address can be reached:

heap_path2
tmp_ptr1
home
local_c
local_8

This complicates exploitation, in particular because the heap_path2 and
tmp_ptr1 pointers get in the way. A skilled attacker might be able to overwrite
all variables with safe data and leverage memory corruption to obtain arbitrary
code execution. However, there is an additional challenge: the ability to
control a hostname to be passed in the X11 display name string. In our PoC
above we have edited /etc/hosts, but this is obviously not possible for an
unprivileged local attacker. A DNS server under the control of the attacker may
be used for this purpose, but such an approach would introduce a number of
additional complications.

That said, as a rule of thumb all memory corruption issues have the potential
to become serious security vulnerabilities until otherwise proven. Therefore,
we recommend to treat this bug as a potential security vulnerability and to fix
it as such.

4. Affected Platforms.

All platforms shipping the Common Desktop Environment are potentially affected.
This includes:
```

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)

```
* Oracle Solaris 10 1/13 (Update 11) and earlier [default installation]

According to the CDE Wiki, the following platforms are officially supported:

* All Official Ubuntu variants 12.04 - 18.04
* Debian 6, 7, 8, 9
* Fedora 17 at least
* Archlinux
* Red Hat
* Slackware 14.0
* OpenBSD
* NetBSD
* FreeBSD 9.2, 10.x, 11.x
* openSUSE Tumbleweed (gcc7)
* openSUSE Leap 4.2 (gcc4)
* SUSE 12 SP3 (gcc4)
* Solaris, OpenIndiana

5. Fix.

The maintainers of the open source CDE 2.x version have issued the following
patches:
https://sourceforge.net/p/cdesktopenv/mailman/message/36900154/
https://sourceforge.net/p/cdesktopenv/code/ci/6b32246d06ab16fd7897dc344db69d0957f3ae08/

Oracle, which maintains a different CDE codebase based on the 1.x train, has
assigned the tracking# S1240932 and has released a fix for all affected and
supported versions of Solaris in the Critical Patch Update (CPU) of April 2020.

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

bash-3.2# chmod -s /usr/dt/bin/dtsession

Please note that during the audit many other potentially exploitable bugs have
surfaced in libDtSvc and in the Common Desktop Environment in general.
Therefore, removing the setuid bit from all CDE binaries is recommended,
regardless of patches released by vendors.

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