Finding Bugs: How to Write a Fuzzer

What's a Bug?

- A mistake in code
- Allows for unintended functionality

Why Bugs Matter

- Give attacker program control
- Attack Software
- Make Software

```
1 gets(buffer);
2
```

```
3 scanf("%10s", buffer);
4 printf(buffer);
```

```
6 free(ptr);
7 free(ptr);
```

What does a fuzzer do?

- Finds bugs
- Spams inputs
- Waits for crash

Code Path

- A branch of code
- Can be controlled with conditional

```
1  scanf("%d", &x);
2  if (x == 0)
3  {
4      puts("Code Path 0");
5  }
6
7  else if (x == 1)
8  {
9      puts("Code Path 1");
10 }
```

Most Important Thing

- Send valid data
- Interact like valid client

Most Important Thing

```
1   if (ntohs(header->ancount) != 0 ||
2       ntohs(header->nscount) != 0 ||
3       ntohs(header->qdcount) == 0 ||
4       OPCODE(header) != QUERY )
5     return 0;
```

Session Based

```
guy@ubuntu:~$ mysql -u root -p
```



```
mysql> select * from fuzzer.protocols;
+------+
| name |
+------+
| dns  |
+------+
1 row in set (0.00 sec)
```

Corrupters

Corrupting Data

- Buffer Overflow
  - Insert Large String
- Format String
  - Insert Format Strings like “%n”

Corrupting Data

- Signed / Unsigned Integer Bugs
  - Insert really large / small integers
- Flipping Bits
  - Helpful when dealing with single byte values

Corrupting Data

- Parsing errors
  - Insert delimiters
  - Insert multiple copies
  - Leave it out

Corrupting Data

- Corrupt just one thing
  - Most of the time
- Other 5% of the time go wild

Crash Detection and Replay

Crash Detection

- Monitor pid
  - Send null signal (signal 0)
- TCP Handshake
  - Try to establish TCP connection
- Monitor output
  - Not reliable

Seeds

- Random function uses seed
- Sequence based on seed
- Same seed same sequence

Crash Replay

- New seed for each fuzzing iteration
- Dump seeds upon crash
- Same seeds same behavior

Fuzzer Validation

Does the fuzzer work?

- Bugs Found
- Code Coverage
  - How many code paths is it missing?

Finding Bugs

- Can it find N - Days?
  - https://www.exploit-db.com/
- Can it pwn easy-ish targets?
  - https://github.com/

Finding Bugs

- How many?
- Common targets 5-10
- Less common, it depends

Code Coverage

- How much are you hitting?
  - How much are you missing?

Code Coverage

- Non-Target Specific
  - 80-95%
- Target-Specific
  - 90-100%

gcov

- Built into gcc
- Compile it with gcov
- Source code required

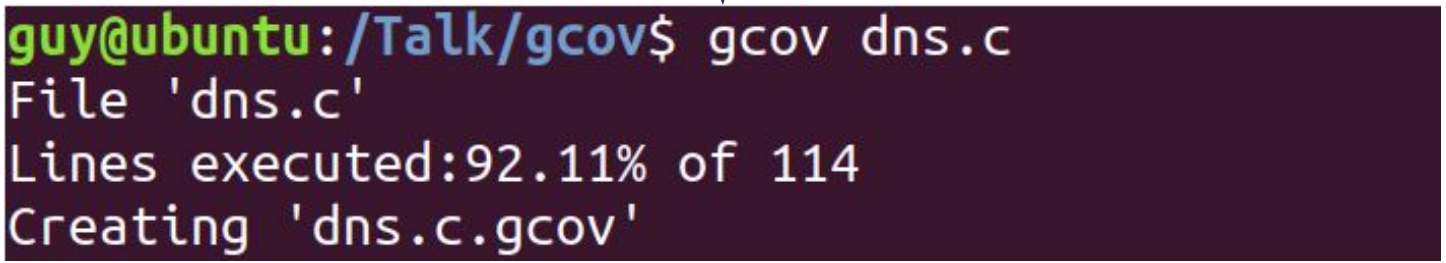
gcov

Compile

Run

gcov

Check Coverage



```
    -:  109:        int qPos, ocLen, i, newPos, limit;
    -:  110:
    -:  111:        char fun[100];
    -:  112:
    1:  113:        dnsQry->id = (rawDnsQuery[0] << 8) + rawDnsQuery[1];
    1:  114:        check = (uint16_t)rawDnsQuery[0];
    -:  115:
    1:  116:        if ((dnsQry->id >> 8) < (check & 0xff))
    -:  117:        {
#####:  118:                dnsQry->id = dnsQry->id + 0x100;
    -:  119:        }
    -:  120:
    1:  121:        dnsQry->flags = ((uint16_t)chr << 8) + rawDnsQuery[3];
    1:  122:        dnsQry->qdcount = (rawDnsQuery[4] << 8) + rawDnsQuery[5];
    1:  123:        dnsQry->ancount = (rawDnsQuery[6] << 8) + rawDnsQuery[7];
    1:  124:        dnsQry->nscount = (rawDnsQuery[8] << 8) + rawDnsQuery[9];
    1:  125:        dnsQry->arcount = (rawDnsQuery[10] << 8) + rawDnsQuery[11];
    -:  126:
    1:  127:        dnsQry->qr = ((dnsQry->flags & 0x8000) >> 15);
    1:  128:        dnsQry->opcode = ((dnsQry->flags & 0x7800) >> 11);
    1:  129:        dnsQry->rd = ((dnsQry->flags & 0x100) >> 1);
    -:  130:
    1:  131:        dnsQry->question = calloc(length, 1);
```

lighthouse

- Markus Gaasedelen
  - https://github.com/gaasedelen/lighthouse/tree/master/coverage/pin
- Uses Intel Pin
  - Framework for creation of binary analysis tools
- Plugin to Binja/IDA
- No Source Code needed

lighthouse

```
guy@ubuntu:~/lighthouse/coverage/pin$ ./pin -t obj-intel64/CodeCoverage.so -- ./dnsmasq --no-daemon
CodeCoverage tool by Agustin Gianni (agustingianni@gmail.com)
Logging code coverage information to: trace.log
Loaded image: 0x55a9528dc000:0x55a952b2babf -> dnsmasq
Loaded image: 0x7fcb6c69a000:0x7fcb6c6c0c23 -> ld-linux-x86-64.so.2
Loaded image: 0x7ffd52755000:0x7ffd5275600a -> [vdso]
Loaded image: 0x7fcb58b81000:0x7fcb58f71adf -> libc.so.6
Loaded image: 0x7fcb582a1000:0x7fcb584aa8bf -> libnss_compat.so.2
Loaded image: 0x7fcb58049000:0x7fcb58254587 -> libnss_nis.so.2
Loaded image: 0x7fcb57e2f000:0x7fcb58048a57 -> libnsl.so.1
Loaded image: 0x7fcb57c1d000:0x7fcb57e2e737 -> libnss_files.so.2
dnsmasq: started, version 2.77 cachesize 150
dnsmasq: compile time options: IPv6 GNU-getopt no-DBus no-i18n no-IDN DHCP DHCPv6 no-Lua TFTP no-con
dnsmasq: reading /etc/resolv.conf
dnsmasq: using nameserver 127.0.0.53#53
dnsmasq: read /etc/hosts - 7 addresses
Segmentation fault (core dumped)
```

File Edit View Tools Help

dnsmasq (ELF Graph)

an

- nanosleep
- fscanf
- sanitise.part.5
- cache_scan_free
- **answer_request**
- rand16
- rand32
- rand64
- canonicalise
- expand_buf
- rand_init
- canonicalise_opt
- expand_filelist
- random_sock
- cleanup_servers
- sig_handler
- clear_cache_and_relo…
- sanitise
- free_transfer
- expand
- answer_auth
- expand_workspace

Xrefs MiniGraph

```
0001be16 in tcp_reques
  call    answer_reque
0001d982 in receive_qu
  call    answer_reque
```

```
void* answer_request(int64_t arg1, int64_t arg2, int64_t arg3,
    int32_t arg4, int32_t arg5, int64_t arg6, int32_t arg7,
    int32_t arg8, int32_t arg9)
```

```
movzx   eax, byte [rdx+0x3]
xor     r12d, r12d  {0x0}
shr     al, 0x4
xor     eax, 0x1
mov     ebx, eax
movzx   eax, word [rdx+0xa]
and     ebx, 0x1
mov     word [rsp+0x8 {var_120_3}], ax
ror     ax, 0x8
test    ax, ax
mov     rax, qword [rel dnsmasq_daemon]
setne   r12b  {0x1}  {0x0}
mov     rax, qword [rax+0x50]
test    rax, rax
je      0xfaa8
```

```
nop     word [rax+rax], ax
```

```
mov     dword [rax+0x20], 0x0
mov     rax, qword [rax+0x28]
test    rax, rax
jne     0xfa98
```

```
lea     rax, [rsp+0xc0 {var_68}]
mov     dword [rsp+0x24 {var_104_1}], 0x0
mov     dword [rsp+0x90 {var_98_1}], 0x0
mov     dword [rsp+0x78 {var_b0_1}], 0x1
mov     qword [rsp+0x70 {var_b8_1}], rax {var_68}
mov     rax, r15
mov     r15d, ebx
mov     ebx, r12d
mov     r12, rax
```

Coverage Overview

Coverage %	Function Name	Address	Blocks Hit	Instructions Hit	Function Size	Complexity
0.00	add_do_bit	0x3DEC0	0 / 1	0 / 10	31	1
18.87	add_edns0_conf…	0x3E000	6 / 19	40 / 212	787	9
0.00	add_extradata_…	0x7520	0 / 3	0 / 7	22	1
0.00	_extradata_… 4	0x7515	0 / 1	0 / 4	11	1
18.03	add_hosts_cname	0xA3B0	2 / 15	11 / 61	212	10
69.19	add_hosts_entry	0xB450	22 / 34	119 / 172	600	19
0.00	add_lla	0x36C00	0 / 3	0 / 34	102	1
0.00	add_local_addrs	0x303F0	0 / 20	0 / 56	173	16
0.00	add_options	0x30BC0	0 / 135	0 / 528	2035	92
0.00	add_prefixes	0x37AB0	0 / 61	0 / 265	1118	38
40.62	add_pseudohead…	0x3DA40	14 / 41	117 / 288	1116	21
29.38	dd_resource_r…	0xF220	17 / 80	109 / 371	1480	43
0.00	add_rev4	0x6710	0 / 12	0 / 88	319	6
0.00	add_rev6	0x684F	0 / 7	0 / 55	190	3
0.00	add_to_ipset	0x3B6D0	0 / 32	0 / 239	984	15
100.00	add_txt	0x12E30	3 / 3	43 / 43	142	2
43.75	add_update_ser…	0x202F0	15 / 32	77 / 176	691	16
100.00	addr6part	0x12120	3 / 3	11 / 11	37	2
0.00	address6_alloc…	0x2FC50	0 / 31	0 / 144	540	19
0.00	ddress6_avail…	0x2FE80	0 / 12	0 / 64	212	9
0.00	address6_valid	0x2FF70	0 / 8	0 / 43	112	3
0.00	address_alloca…	0x232F0	0 / 40	0 / 153	557	25
0.00	address_availa…	0x23020	0 / 15	0 / 46	139	8
0.00	alarm	0x61C0	0 / 1	0 / 1	6	1
100.00	allocate_frec	0x1B3B0	3 / 3	17 / 17	84	2
43.75	allocate_rfd	0x1C4E0	5 / 13	28 / 64	225	6
23.53	allocate_sfd	0x1FE70	7 / 28	32 / 136	469	16
0.00	answer_auth	0x39050	0 / 413	0 / 2211	9305	244
23.37	answer_request	0xF980	74 / 351	411 / 1759	7714	226
0.00	apply_delay	0x6FC8	0 / 12	0 / 59	179	5
85.29	atoi_check	0x12C30	7 / 9	29 / 34	86	4
100.00	atoi_check16	0x12CA0	3 / 3	10 / 10	26	2
100.00	bind	0x64B0	1 / 1	1 / 1	6	1
0.00	bindtodevice	0x35C80	0 / 6	0 / 34	131	3

Composer

A - 14.55% - trace.log

Cursor: 0xf980 ELF ▾ Graph ▾ Options ▾

Target specific fuzzer

- Does it support the entire protocol?
  - All record types
  - Dns cookies
  - Multi-question queries
  - Zone transfers (axfr)

Target specific fuzzer

- Is there more to the attack surface?
  - Structured Exception Handling (SEH)
  - Additional commands
  - Other protocols

Target specific fuzzer

- Rfc adherence
  - Does it do what it’s supposed to
  - If not, this is where the fun starts

Fuzzing Rate

- Don’t bury your target with inputs
  - If it can only handle 100 inputs a second, don’t send 1,000 inputs a second
- Will lead to issues with crash replay
  - Fuzzer thinks it crashed it on seed 40,000, when it crashed it on seed 20,000

Will my fuzzer exhaust all possible inputs?

Inputs Per Day

- 10 inputs per second
- 600 inputs per minute
- 36000 inputs per hour
- 864000 inputs per day

Combinations

- Header: ID - 0xffff (65535)
- Header: Flags - 160
- Header: Qdcount -0xffff (65535)

Combinations

- Header: Ancount - 0xffff (65535)
- Header: Nscount - 0xffff (65535)
- Header: Arcount - 0xffff (65535)

Combinations

- Question: QName - 5100 (len 20)
- Question: QType - 0xffff (65535)
- Question: QClass - 0xffff (65535)

Combinations

65535 * 160 * 65535 * 65535 * 65535 * 65535 * 65535 * 5100 * 65535 * 65535

=

42365263502777049134903*10^16

Not Going to Happen

- How long at that rate?
  - 1.34 * 10^31
- What if we sent 1,000,000 a second?
  - 1.35 * 10^25

Lowball Figure

- Didn’t include corrupters
- Didn’t include zone transfers (axfr)
- Didn’t include dns cookies
- Actual number is a lot higher

Demo Time!

Debugging a Crash

- Target DNSMasq 2.77
  - https://security.googleblog.com/2017/10/behind-masq-yet-more-dns-and-dhcp.html
- I’m going to be using binja/gdb-gef
  - https://github.com/hugsy/gef
  - https://binary.ninja/

```
guy@ubuntu:/Talk/target/src$ gdb ./dnsmasq
GNU gdb (Ubuntu 8.1-0ubuntu3) 8.1.0.20180409-git
Copyright (C) 2018 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.  Type "show copying"
and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
GEF for linux ready, type `gef' to start, `gef config' to configure
75 commands loaded for GDB 8.1.0.20180409-git using Python engine 3.6
[*] 5 commands could not be loaded, run `gef missing` to know why.
Reading symbols from ./dnsmasq...(no debugging symbols found)...done.
gef➤  r --no-daemon
Starting program: /Talk/target/src/dnsmasq --no-daemon
dnsmasq: started, version 2.77 cachesize 150
dnsmasq: compile time options: IPv6 GNU-getopt no-DBus no-i18n no-IDN DHCP DHCPv6 no-Lua TFTP no-conntrack ipset auth no-D
NSSEC loop-detect inotify
dnsmasq: reading /etc/resolv.conf
dnsmasq: using nameserver 127.0.0.53#53
dnsmasq: read /etc/hosts - 7 addresses
```

```
guy@ubuntu:/Talk/fuzzer$ python fuzz.py -r outputFile.txt
Initializing fuzzer
read seeds are: 20
replaing seed number: 0
 ◆[ED]4◆l◆◆mp=00◆9x
Sending Corrupted Data:
replaing seed number: 1
using seed: ◆7◆◆p◆◆[00]1◆◆}f◆n?QY5◆
Sending Corrupted Data:
Length of Packet: 52
replaing seed number: 2
using seed: ◆◆◆◆Y◆◆n◆j◆◆[00]◆ kM[00]◆
Sending Corrupted Data:
Length of Packet: 60
replaing seed number: 3
using seed: W1◆◆[00]R◆◆
                  ◆◆(r~◆[00]
 ◆
Sending Corrupted Data:
Length of Packet: 32
replaing seed number: 4
using seed: ◆tn◆[00]|Ž◆◆◆◆:RZ◆◆()
Sending Corrupted Data:
Length of Packet: 7441
replaing seed number: 5
using seed: e9◆◆◆◆M◆◆&0◆◆[00]◆◆◆┤m
Sending Non-Corrupted Data:
Length of Packet: 56
replaing seed number: 6
using seed: ◆l.◆        ◆◆[00]◆◆eÈ◆◆7P'◆
Sending Corrupted Data:
Length of Packet: 33
replaing seed number: 7
```

guy@ubuntu: /Talk/target/src

File Edit View Search Terminal Help

```
$cs: 0x0033 $ss: 0x002b $ds: 0x0000 $es: 0x0000 $fs: 0x0000 $gs: 0x0000
──────────────────────────────────────────────────────────────── stack ────
0x00007fffffffdb18│+0x0000: 0x00005555555639f6  →  <answer_request+118> movzx eax, WORD PTR [rbx+0x6]    ←$rsp
0x00007fffffffdb20│+0x0008: 0x00005555557a4260  →  0x732e726173617500
0x00007fffffffdb28│+0x0010: 0x00005555557a4201  →  0x0000000000000000
0x00007fffffffdb30│+0x0018: 0x000000005c9e8879
0x00007fffffffdb38│+0x0020: 0x0000000000000880
0x00007fffffffdb40│+0x0028: 0x0000000055590001
0x00007fffffffdb48│+0x0030: 0x00005555557a5340  →  0x000001000001f33a
0x00007fffffffdb50│+0x0038: 0x00005555557a001c  →  0x5559398300000000
─────────────────────────────────────────────────────────── code:x86:64 ────
   0x7ffff7b72fde <__memset_avx2_unaligned_erms+142> and    rdx, 0xffffffffffffff80
   0x7ffff7b72fe2 <__memset_avx2_unaligned_erms+146> cmp    rcx, rdx
   0x7ffff7b72fe5 <__memset_avx2_unaligned_erms+149> je     0x7ffff7b72fa1 <__memset_avx2_unaligned_erms+81>
 →0x7ffff7b72fe7 <__memset_avx2_unaligned_erms+151> vmovdqa YMMWORD PTR [rcx], ymm0
   0x7ffff7b72feb <__memset_avx2_unaligned_erms+155> vmovdqa YMMWORD PTR [rcx+0x20], ymm0
   0x7ffff7b72ff0 <__memset_avx2_unaligned_erms+160> vmovdqa YMMWORD PTR [rcx+0x40], ymm0
   0x7ffff7b72ff5 <__memset_avx2_unaligned_erms+165> vmovdqa YMMWORD PTR [rcx+0x60], ymm0
   0x7ffff7b72ffa <__memset_avx2_unaligned_erms+170> add    rcx, 0x80
   0x7ffff7b73001 <__memset_avx2_unaligned_erms+177> cmp    rdx, rcx
─────────────────────────────────────────────────────────────── threads ────
[#0] Id 1, Name: "dnsmasq", stopped, reason: SIGSEGV
───────────────────────────────────────────────────────────────── trace ────
[#0] 0x7ffff7b72fe7 → __memset_avx2_unaligned_erms()
[#1] 0x5555555639f6 → answer_request()
[#2] 0x555555571987 → receive_query()
[#3] 0x55555557540b → check_dns_listeners()
[#4] 0x55555555d439 → main()
────────────────────────────────────────────────────────────────────────────
__memset_avx2_unaligned_erms () at ../sysdeps/x86_64/multiarch/memset-vec-unaligned-erms.S:200
200     ../sysdeps/x86_64/multiarch/memset-vec-unaligned-erms.S: No such file or directory.
gef➤
```

```
   0x7ffff7b72fe2 <__memset_avx2_unaligned_erms+146> cmp    rcx, rdx
   0x7ffff7b72fe5 <__memset_avx2_unaligned_erms+149> je     0x7ffff7b72fa1 <__memset_avx2_unaligned_erms+81>
 → 0x7ffff7b72fe7 <__memset_avx2_unaligned_erms+151> vmovdqa YMMWORD PTR [rcx], ymm0
   0x7ffff7b72feb <__memset_avx2_unaligned_erms+155> vmovdqa YMMWORD PTR [rcx+0x20], ymm0
   0x7ffff7b72ff0 <__memset_avx2_unaligned_erms+160> vmovdqa YMMWORD PTR [rcx+0x40], ymm0
   0x7ffff7b72ff5 <__memset_avx2_unaligned_erms+165> vmovdqa YMMWORD PTR [rcx+0x60], ymm0
   0x7ffff7b72ffa <__memset_avx2_unaligned_erms+170> add    rcx, 0x80
   0x7ffff7b73001 <__memset_avx2_unaligned_erms+177> cmp    rdx, rcx
```

threads

```
[#0] Id 1, Name: "dnsmasq", stopped, reason: SIGSEGV
```

trace

```
[#0] 0x7ffff7b72fe7 → __memset_avx2_unaligned_erms()
[#1] 0x5555555639f6 → answer_request()
[#2] 0x555555571987 → receive_query()
[#3] 0x55555557540b → check_dns_listeners()
[#4] 0x55555555d439 → main()
```

```
__memset_avx2_unaligned_erms () at ../sysdeps/x86_64/multiarch/memset-vec-unaligned-erms.S:200
200     ../sysdeps/x86_64/multiarch/memset-vec-unaligned-erms.S: No such file or directory.
gef➤  p $rcx
$1 = 0x5555557c5000
gef➤  vmmap
Start              End                Offset             Perm Path
0x0000555555554000 0x00005555555a0000 0x0000000000000000 r-x /Talk/target/src/dnsmasq
0x00005555555579f000 0x00005555557a2000 0x000000000004b000 r-- /Talk/target/src/dnsmasq
0x00005555557a2000 0x00005555557a4000 0x000000000004e000 rw- /Talk/target/src/dnsmasq
0x00005555557a4000 0x00005555557c5000 0x0000000000000000 rw- [heap]
0x00007ffff71a2000 0x00007ffff71ad000 0x0000000000000000 r-x /lib/x86_64-linux-gnu/libnss_files-2.27.so
0x00007ffff71ad000 0x00007ffff73ac000 0x000000000000b000 --- /lib/x86_64-linux-gnu/libnss_files-2.27.so
0x00007ffff73ac000 0x00007ffff73ad000 0x000000000000a000 r-- /lib/x86_64-linux-gnu/libnss_files-2.27.so
0x00007ffff73ad000 0x00007ffff73ae000 0x000000000000b000 rw- /lib/x86_64-linux-gnu/libnss_files-2.27.so
0x00007ffff73ae000 0x00007ffff73b4000 0x0000000000000000 rw-
```

```
0x00007ffff75ca000 0x00007ffff75cb000 0x0000000000016000 r-- /lib/x86_64-linux-gnu/libnsl-2.27.so
0x00007ffff75cb000 0x00007ffff75cc000 0x0000000000017000 rw- /lib/x86_64-linux-gnu/libnsl-2.27.so
0x00007ffff75cc000 0x00007ffff75ce000 0x0000000000000000 rw-
0x00007ffff75ce000 0x00007ffff75d9000 0x0000000000000000 r-x /lib/x86_64-linux-gnu/libnss_nis-2.27.so
0x00007ffff75d9000 0x00007ffff77d8000 0x000000000000b000 --- /lib/x86_64-linux-gnu/libnss_nis-2.27.so
0x00007ffff77d8000 0x00007ffff77d9000 0x000000000000a000 r-- /lib/x86_64-linux-gnu/libnss_nis-2.27.so
0x00007ffff77d9000 0x00007ffff77da000 0x000000000000b000 rw- /lib/x86_64-linux-gnu/libnss_nis-2.27.so
0x00007ffff77da000 0x00007ffff77e2000 0x0000000000000000 r-x /lib/x86_64-linux-gnu/libnss_compat-2.27.so
0x00007ffff77e2000 0x00007ffff79e2000 0x0000000000008000 --- /lib/x86_64-linux-gnu/libnss_compat-2.27.so
0x00007ffff79e2000 0x00007ffff79e3000 0x0000000000008000 r-- /lib/x86_64-linux-gnu/libnss_compat-2.27.so
0x00007ffff79e3000 0x00007ffff79e4000 0x0000000000009000 rw- /lib/x86_64-linux-gnu/libnss_compat-2.27.so
0x00007ffff79e4000 0x00007ffff7bcb000 0x0000000000000000 r-x /lib/x86_64-linux-gnu/libc-2.27.so
0x00007ffff7bcb000 0x00007ffff7dcb000 0x00000000001e7000 --- /lib/x86_64-linux-gnu/libc-2.27.so
0x00007ffff7dcb000 0x00007ffff7dcf000 0x00000000001e7000 r-- /lib/x86_64-linux-gnu/libc-2.27.so
0x00007ffff7dcf000 0x00007ffff7dd1000 0x00000000001eb000 rw- /lib/x86_64-linux-gnu/libc-2.27.so
0x00007ffff7dd1000 0x00007ffff7dd5000 0x0000000000000000 rw-
0x00007ffff7dd5000 0x00007ffff7dfc000 0x0000000000000000 r-x /lib/x86_64-linux-gnu/ld-2.27.so
0x00007ffff7fdc000 0x00007ffff7fde000 0x0000000000000000 rw-
0x00007ffff7ff7000 0x00007ffff7ffa000 0x0000000000000000 r-- [vvar]
0x00007ffff7ffa000 0x00007ffff7ffc000 0x0000000000000000 r-x [vdso]
0x00007ffff7ffc000 0x00007ffff7ffd000 0x0000000000027000 r-- /lib/x86_64-linux-gnu/ld-2.27.so
0x00007ffff7ffd000 0x00007ffff7ffe000 0x0000000000028000 rw- /lib/x86_64-linux-gnu/ld-2.27.so
0x00007ffff7ffe000 0x00007ffff7fff000 0x0000000000000000 rw-
0x00007ffffffde000 0x00007ffffffff000 0x0000000000000000 rw- [stack]
0xffffffffff600000 0xffffffffff601000 0x0000000000000000 r-x [vsyscall]
gef➤  bt
#0  __memset_avx2_unaligned_erms () at ../sysdeps/x86_64/multiarch/memset-vec-unaligned-erms.S:200
#1  0x00005555555639f6 in answer_request ()
#2  0x0000555555571987 in receive_query ()
#3  0x000055555557540b in check_dns_listeners ()
#4  0x000055555555d439 in main ()
gef➤
```

```
guy@ubuntu:/Talk/fuzzer$ python
Python 2.7.15rc1 (default, Nov 12 2018, 14:31:15)
[GCC 7.3.0] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> hex(0x00005555555639f6 - 0x0000555555554000)
'0xf9f6'
>>>
```

Functions

Cross Referen…

dnsmasq (ELF Graph)

```
void* answer_request(int64_t arg1, int64_t arg2, int64_t arg3, int32_t arg4, int32_t arg5, int64_t arg6, int32_t arg7, int32_t arg8,
    int32_t arg9)
```

```
answer_request:
push    r15 {__saved_r15}
push    r14 {__saved_r14}
push    r13 {__saved_r13}
push    r12 {__saved_r12}
push    rbp {var_28}
push    rbx {__saved_rbx}
mov     rbx, rdx
sub     rsp, 0xf8
mov     rax, qword [fs:0x28]
mov     qword [rsp+0xe8 {var_40}], rax
xor     eax, eax  {0x0}
mov     qword [rsp+0x40 {var_e8}], rsi
mov     rax, qword [rel dnsmasq_daemon]
sub     rsi, rdi
mov     qword [rsp+0x50 {var_d8}], rdx
mov     dword [rsp+0x3c {var_ec}], ecx
mov     rdx, rsi
mov     rcx, rbx
sub     rdx, rbx
mov     qword [rsp+0x28 {var_100}], rdi
mov     rbx, rdi
xor     esi, esi  {0x0}
add     rdi, rcx
mov     dword [rsp+0x5c {var_cc}], r8d
mov     qword [rsp+0x10 {var_118}], r9
mov     r15, qword [rax+0x360]
mov     dword [rsp+0xb4 {var_74}], 0x0
call    memset
movzx   eax, word [rbx+0x6]
mov     word [rsp+0x8 {var_120}], ax
ror     ax, 0x8
test    ax, ax
jne     0xfd20
```

```
mov     rdx, qword [rsp+0x28 {var_100}]
movzx   eax, word [rdx+0x8]
mov     word [rsp+0x8 {var_120_1}], ax
```

Feature Map

Mini Graph

```
1217  int nameoffset;
1218  unsigned short flag;
1219  int q, ans, anscount = 0, addncount = 0;
1220  int dryrun = 0;
1221  struct crec *crecp;
1222  int nxdomain = 0, auth = 1, trunc = 0, sec_data = 1;
1223  struct mx_srv_record *rec;
1224  size_t len;
1225
1226  /* Clear buffer beyond request to avoid risk of
1227     information disclosure. */
1228  memset(((char *)header) + qlen, 0,
1229     (limit - ((char *)header)) - qlen);
1230
1231  if (ntohs(header->ancount) != 0 ||
1232      ntohs(header->nscount) != 0 ||
1233      ntohs(header->qdcount) == 0 ||
1234      OPCODE(header) != QUERY )
1235    return 0;
1236
1237  /* Don't return AD set if checking disabled. */
1238  if (header->hb4 & HB4_CD)
1239    sec_data = 0;
1240
```

Questions?

Try it Out

- Try fuzzing DNSMasq 2.77
- CTF challenge specifically made to go with this talk
- Write your own fuzzer

Special Thanks

- Scott Devault \ Jonathan Grimm
  - Got me interested in infosec
- Hack@ucf
  - Helped me take my first couple of steps in infosec
- Raytheon SI (Center of Innovation)
  - Teaching me most of the stuff I said here

Projects

- DNS Fuzzer
  - https://github.com/guyinatuxedo/dns-fuzzer
- Other projects
  - Ctf writeups
  - Emulator / debugger
  - https://github.com/guyinatuxedo

Who is This Guy?

- College Student
- Worked at Raytheon SI, and a couple other places
- Used to do CCDC, now just CTFs
- Spends most of my time staring at assembly
- Twitter: @aguyinatux
- Just a guy, who’s not in a tuxedo