Report: Homework 2 - DNS Lookup

Cole McAnelly
CS 463

1. **Case 1**
    i. `random0.irl`
    ii. `random3.irl`
    iii. `random5.irl`
    iv. `random6.irl`
2. **Case 2**
3. **Case 3**
4. **Case 4**
5. **Extra Credit**

Case 1

`random0.irl`

```
Lookup  : random0.irl
Query   : random0.irl, type 1, TXID 0x0000
Server  : 128.194.135.82
*********************************
Attempt 0 with 29 bytes... response in 178ms with 82 bytes
  TXID 0x0000, flags 0x8400, questions 1, answers 2, authority 0, additional 0
  succeeded with Rcode = 0
  ------------ [questions] ----------
        random0.irl type 1 class 1
  ------------ [answers] ------------
  ++ invalid record: jump into fixed DNS header
```

This error is caused what there is a jump offset that is less than the size of the `FixedDNSHeader`, which attempts to overwrite crucial data. This can be caught with a simple bounds check before jumping.

`random3.irl`

```
Lookup  : random3.irl
Query   : random3.irl, type 1, TXID 0x0003
Server  : 128.194.135.82
*********************************
Attempt 0 with 29 bytes...
  ++ invalid reply: packet smaller than fixed DNS header
```

This error occurs when the DNS server attempts to return a packet that is smaller than the mandatory fixed header size. This can be caught immediately after the receive functionality.

`random5.irl`

```
Lookup  : random5.irl
Query   : random5.irl, type 1, TXID 0x0007
Server  : 128.194.135.82
*********************************
Attempt 0 with 29 bytes... response in 17ms with 71 bytes
```

```
  TXID 0x0007, flags 0x8400, questions 1, answers 2, authority 0, additional 0
  succeeded with Rcode = 0
  ------------ [questions] ----------
        random5.irl type 1 class 1
  ------------ [answers] ------------
        random.irl A 1.1.1.1 TTL = 30
  ++ invalid record: jump beyond packet boundary
```

This error occurs when the jump offset is outside of the packet boundary. This can be checked with a simple bounds check before jumping.

`random6.irl`

```
Lookup  : random6.irl
Query   : random6.irl, type 1, TXID 0x0008
Server  : 128.194.135.82
*******************************
Attempt 0 with 29 bytes... response in 21ms with 59 bytes
  TXID 0x0008, flags 0x8400, questions 1, answers 2, authority 0, additional 0
  succeeded with Rcode = 0
  ------------ [questions] ----------
        random6.irl type 1 class 1
  ------------ [answers] ------------
  ++ invalid record: jump loop
```

This error occurs when there is a number of jumps that cause a high number of them to occur. We can prevent this by checking that the number of jumps doesn't get higher than the maximum amount of jumps that could be stored in a packet.

Case 2 (`random1.irl`)

```
Lookup  : random1.irl
Query   : random1.irl, type 1, TXID 0x0001
Server  : 128.194.135.82
*******************************
Attempt 0 with 29 bytes... response in 42ms with 468 bytes
  TXID 0x0001, flags 0x8600, questions 1, answers 1, authority 0, additional 65535
  succeeded with Rcode = 0
  ------------ [questions] ----------
        random1.irl type 1 class 1
  ------------ [answers] ------------
        random.irl A 1.1.1.1 TTL = 30
  ------------ [additional] ---------
  ++ invalid section: not enough records
```

This errors is caused by the amount of actual records is greater than the total number of stated records. This can be checked by counting the number of responses recieved, and comparing it against the stated number of responses in the fixed DNS header.

Case 3 (`random7.irl`)

```
Lookup  : random7.irl
Query   : random7.irl, type 1, TXID 0x0009
Server  : 128.194.135.82
*******************************
Attempt 0 with 29 bytes... response in 13ms with 42 bytes
  TXID 0x0009, flags 0x8400, questions 1, answers 2, authority 0, additional 0
```

```
  succeeded with Rcode = 0
  ------------ [questions] ----------
        random7.irl type 1 class 1
  ------------ [answers] ------------
  ++ invalid record: truncated jump offset
```

This error can happen if the packet ends right in between the jump offset bytes, so the program can know that it should jump, it just doesn't have the lower 8 bits of the offset address. We can check this by doing a bounds check on the next byte before we use it for the lower half of the offset address.

Case 4 (`random4.irl`)

```
Lookup  : random4.irl
Query   : random4.irl, type 1, TXID 0x000A
Server  : 128.194.135.82
*******************************
Attempt 0 with 29 bytes... response in 33ms with 324 bytes
  TXID 0x000A, flags 0x8400, questions 1, answers 1, authority 0, additional 11
  succeeded with Rcode = 0
  ------------ [questions] ----------
        random4.irl type 1 class 1
  ------------ [answers] ------------
        random.irl A 1.1.1.1 TTL = 30
  ------------ [additional] ---------
        Episode.IV A 2.2.2.2 TTL = 30
        A.NEW.HOPE A 2.2.2.2 TTL = 30
        It.is.a.period.of.civil.war A 2.2.2.2 TTL = 30
        Rebel.spaceships A 2.2.2.2 TTL = 30
        striking.from.a.hidden.base A 2.2.2.2 TTL = 30
        have.won.their.first.victory A 2.2.2.2 TTL = 30
        against.the.evil.Galactic.Empire A 2.2.2.2 TTL = 30
  ++ invalid record: truncated RR answer header
```

This error occurs when the Answer header has been corrupted in some way, either it was cut off by the packet ending, or other data has malformed it. We can check it be doing a bounds check on the current position *plus* the size of the Answer Header. By doing this, we can ensure that all of the data in the answer header is left unmodified.

```
Lookup  : random4.irl
Query   : random4.irl, type 1, TXID 0x0000
Server  : 128.194.135.82
*******************************
Attempt 0 with 29 bytes... response in 62ms with 54 bytes
  TXID 0x0000, flags 0x8400, questions 1, answers 1, authority 0, additional 11
  succeeded with Rcode = 0
  ------------ [questions] ----------
        random4.irl type 1 class 1
  ------------ [answers] ------------
        random.irl A 0.1.1.1 TTL = 30
  ------------ [additional] ---------
  ++ invalid section: not enough records
```

This errors is caused by the amount of actual records is greater than the total number of stated records. This can be checked by counting the number of responses recieved, and comparing it against the stated number of responses in the fixed DNS header.

```
Lookup  : random4.irl
Query   : random4.irl, type 1, TXID 0x0004
Server  : 128.194.135.82
```

```
*********************************
Attempt 0 with 29 bytes... response in 19ms with 144 bytes
  TXID 0x0004, flags 0x8400, questions 1, answers 1, authority 0, additional 11
  succeeded with Rcode = 0
  ------------ [questions] ----------
        random4.irl type 1 class 1
  ------------ [answers] ------------
        random.irl A 1.1.1.1 TTL = 30
  ------------ [additional] ---------
        Episode.IV A 2.2.2.2 TTL = 30
        A.NEW.HOPE A 2.2.2.2 TTL = 30
  ++ invalid record: truncated name
```

This error happens when the name in the packet is imcomplete, most likely because of abrupt packing ending. We can prevent this by doing bounds checks on the iterater as we go through the name, doing a bounds check each time.

Extra Credit

I queried this hostname repeatedly so that I could get a good capture of as many DNS requests as possible in Wireshark, which helped me to see the patterns that were there.

```
Lookup  : random8.irl
Query   : random8.irl, type 1, TXID 0x0001
Server  : 128.194.135.82
*********************************
Attempt 0 with 29 bytes... response in 4ms with 468 bytes
  TXID 0x0001, flags 0x8400, questions 1, answers 1, authority 0, additional 11
  succeeded with Rcode = 0
  ------------ [questions] ----------
        random8.irl type 1 class 1
  ------------ [answers] ------------
        random.irl A 1.1.1.1 TTL = 30
  ------------ [additional] ---------
        Episode.IV A 2.2.2.2 TTL = 30
        A.NEW.HOPE A 2.2.2.2 TTL = 30
        It.is.a.period.of.civil.war A 2.2.2.2 TTL = 30
        Rebel.spaceships A 2.2.2.2 TTL = 30
        striking.from.a.hidden.base A 2.2.2.2 TTL = 30
        have.won.their.first.victory A 2.2.2.2 TTL = 30
        against.the.evil.Galactic.Empire A 2.2.2.2 TTL = 30
        During.the.battle A 2.2.2.2 TTL = 30
        Rebel.spies.managed A 108.111.108.108 TTL = 1819045740
  ++ invalid record: RR value length stretches the answer beyond packet
```

	Time	Source	Destination	Protocol	Lengtl	Info
7	2.203019	128.194.135.82	10.247.139.50	DNS	510	Standard query response 0x0001 A random8.irl A 1.1.1.1 A \002\002.\001A.NEW.HOPE/01
8	2.212116	10.247.139.50	128.194.135.82	DNS	71	Standard query 0x0002 A random8.irl
9	2.216935	128.194.135.82	10.247.139.50	DNS	510	Standard query response 0x0002 A random8.irl A 1.1.1.1 A \002\002.\001A.NEW.HOPE/01
10	2.224969	10.247.139.50	128.194.135.82	DNS	71	Standard query 0x0003 A random8.irl
11	2.228847	128.194.135.82	10.247.139.50	DNS	510	Standard query response 0x0003 A random8.irl A 1.1.1.1 A \002\002.\001A.NEW.HOPE/01
12	2.236475	10.247.139.50	128.194.135.82	DNS	71	Standard query 0x0004 A random8.irl
13	2.240967	128.194.135.82	10.247.139.50	DNS	510	Standard query response 0x0004 A random8.irl A 1.1.1.1 A \002\002.\001A.NEW.HOPE/01
14	2.246333	10.247.139.50	128.194.135.82	DNS	71	Standard query 0x0005 A random8.irl
15	2.249768	128.194.135.82	10.247.139.50	DNS	510	Standard query response 0x0005 A random8.irl A 1.1.1.1 A[Malformed Packet]
16	2.252800	10.247.139.50	128.194.135.82	DNS	71	Standard query 0x0006 A random8.irl
17	2.256696	128.194.135.82	10.247.139.50	DNS	510	Standard query response 0x0006 A random8.irl A 1.1.1.1 A \002\002.\001A.NEW.HOPE/01
18	2.262452	10.247.139.50	128.194.135.82	DNS	71	Standard query 0x0007 A random8.irl

```
0070  03 00 00 00 1e 00 04 02  02 02 02 01 41 03 4e 45   ........ ....A·NE
0080  57 04 48 4f 50 45 00 00  01 00 03 00 00 00 1e 00   W·HOPE·· ........
0090  04 02 02 02 02 02 49 74  02 69 73 01 61 06 70 65   ......It ·is·a·pe
00a0  72 69 6f 64 02 6f 66 05  63 69 76 69 6c 03 77 61   riod·of· civil·wa
00b0  72 00 00 01 00 03 00 00  00 1e 00 04 02 02 02 02   r·...... ........
00c0  05 52 65 62 65 6c 0a 73  70 61 63 65 73 68 69 70   ·Rebel·s paceship
00d0  73 00 00 01 00 03 00 00  00 1e 00 04 02 02 02 02   s·...... ........
00e0  08 73 74 72 69 6b 69 6e  67 04 66 72 6f 6d 01 61   ·strikin g·from·a
00f0  06 68 69 64 64 65 6e 04  62 61 73 65 00 00 01 00   ·hidden· base····
0100  03 00 00 00 1e 00 04 02  02 02 02 04 68 61 76 65   ........ ····have
0110  03 77 6f 6e 05 74 68 65  69 72 05 66 69 72 73 74   ·won·the ir·first
0120  07 76 69 63 74 6f 72 79  00 00 01 00 03 00 00 00   ·victory ........
0130  1e 00 04 02 02 02 02 07  61 67 61 69 6e 73 74 03   ........ against·
0140  74 68 65 04 65 76 69 6c  08 47 61 6c 61 63 74 69   the·evil ·Galacti
0150  63 06 45 6d 70 69 72 65  00 00 01 00 03 00 00 00   c·Empire ........
0160  1e 00 04 02 02 02 02 06  44 75 72 69 6e 67 03 74   ........ During·t
0170  68 65 06 62 61 74 74 6c  65 00 00 01 00 03 00 00   he·battl e·......
0180  00 1e 00 04 02 02 02 02  05 52 65 62 65 6c 05 73   ........ ·Rebel·s
0190  70 69 65 73 07 6d 61 6e  61 67 65 64 00 00 01 6c   pies·man aged···l
01a0  6f 6c 6c 6f 6c 6c 6f 6c  6c 6f 6c 6c 6f 6c 6c 6f   ollollol lollollo
01b0  6c 6c 6f 6c 6c 6f 6c 6c  6f 6c 6c 6f 6c 6c 6f 6c   llolloll ollollol
01c0  6c 6f 6c 6c 6f 6c 6c 6f  6c 6c 6f 6c 6c 6f 6c 6c   lollollo llolloll
01d0  6f 6c 6c 6f 6c 6c 6f 6c  6c 6f 6c 6c 6f 6c 6c 6f   ollollol lollollo
01e0  6c 6c 6f 6c 6c 6f 6c 6c  6f 6c 6c 6f 6c 6c 6f 6c   llolloll ollollol
01f0  6c 6f 6c 03 00 00 00 1e  00 04 02 02 02 02        lol..... ......
```

This endpoint on the server takes the default response (11 records in the `additional` name entries that are from the intro sequence to Star Wars: A New Hope), and randomly chooses a places among the records. It then enters 2 null bytes before "padding" the rest of the packet with "lol" until the length of the packet is 510 bytes in length. This would be pretty easy with a random number generator and knowing the current legnth of the packet.