IP fragmentation attack on DNS

Presented by: Ondrej Sury, CZ.NIC
Original work by: Amir Herzberg & Haya Shulman
PoC by: CZ.NIC Labs & Other parties

The new attack vector

- Cache-poisoning attack on resolvers
- Exploits IP fragmentation
  - Spoof the second fragment
- Off the path attack
- Reduces the entropy from <=32-bits (dstport + DNS-ID) to 16-bits (IP-ID)
  - All the “DNS” entropy stays in the first fragment

The attack in the box

- Set the MTU between the victim and the authoritative name server to a desired value
  - (optionally) Spoof the ICMP fragmentation needed packet
- Query the victim for a name that generates a response big enough to fragment
- Pre-load the spoofed second fragments into the victim's buffer
  - Fragments are reassembled off the order
- The victim IP stack reassembles
  - The original first fragment
  - The spoofed second fragment

PoC status

- We have two independent implementations
  - The Other party – working PoC in the test environment
  - CZ.NIC Labs – working PoC, but needs more polishing
    - The first successful attack was done yesterday and it's still little crafted

The bad

- Indeed, the attack is real
- Almost everybody is vulnerable
  - But TLDs are best targets

The good

- Some defenses are already in place
- Quite a lot of requirements to pull this off
  - Makes it difficult to write a PoC (but only for the first time)
- It's easy to defend on the authoritative side

Things under attacker's control

- Registering domain name(s)
  - Thus TLDs are more vulnerable
- Selecting/creating names of name servers for domain(s)
- Providing glue data (IPv4 and/or IPv6) for new name servers' names
- Provisioning zone contents, including possible use of zone cuts and child, grandchild, etc. zones

Things not under attacker's control

- The order of RR's returned from authoritative server
  - Can be engineered around
- TTL of the glue data served by authoritative server
  - Can be engineered around too

Tricks used – fragmentation point

- Forcing DNS response to be large enough to fragment
- Crafting the query, the NS RRSet and the Glue data, so that response split before the last RDATA in Authoritative section

Tricks used – UDP checksum

- One's complement 16-bit checksum
  - Any re-ordering of 16-bit aligned values preserves the checksum
- Choosing the NS names so they don't modify the UDP checksum when reordered
- Choosing the QNAMEs that don't change the UDP checksum and the packet length:
  - aaaaaaaazzzzzzzz.example.com.
  - aaaaaaabzzzzzzzy.example.com.
- Single second fragment will match all these first fragments
  - Only thing that varies is IP-ID

Tricks used – poisoning

- Requires a name that doesn't exist in the original zone
  - Attacker can control that – the last authoritative section RDATA is in the second fragment
- Poison the GLUE in the additional section
  - To force a query to a attacker's server
- The non-forged packets won't contain the poisoned name, so they don't interfere

The spoofed response

```
;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;aaaaaaabzzzzzzzy.aaaaaaaazzzzzzzzaaaaaaaa.dname1.chld.kasuar.cz. IN A

;; AUTHORITY SECTION:
kasuar.cz.              172800  IN      NS
eieio.iii.nic.x.cz.xx.a.xx.c.cz.a.example.c.example.forftw.ccxx.chld.kasuar.cz.
[...]
kasuar.cz.              172800  IN      NS
eieio.iii.nic.x.cz.xx.a.xx.c.cz.a.example.c.example.forftw.ggtt.chld.kasuar.cz.
kasuar.cz.              172800  IN      NS      eieio.iii.forftw.ggtt.x.cz.xx.a.xx.c.cz.a.example.c.ns.nic.cz.

;; ADDITIONAL SECTION:
c.ns.nic.cz.  172800 IN  A       192.168.6.3
eieio.iii.nic.x.cz.xx.a.xx.c.cz.a.example.c.example.forftw.ddww.chld.kasuar.cz. 172800 IN AAAA fc00::6f00:c412:c03d:c5ae:c5ae
eieio.iii.nic.x.cz.xx.a.xx.c.cz.a.example.c.example.forftw.ggtt.chld.kasuar.cz. 172800 IN A 192.168.6.3
[...]
eieio.iii.nic.x.cz.xx.a.xx.c.cz.a.example.c.example.forftw.hhss.chld.kasuar.cz. 172800 IN A 192.168.6.3
eieio.iii.nic.x.cz.xx.a.xx.c.cz.a.example.c.example.forftw.hhss.chld.kasuar.cz. 172800 IN AAAA fc00::c5ad:c5ad:c5ae:c5ae
eieio.iii.forftw.ggtt.x.cz.xx.a.xx.c.cz.a.example.c.ns.nic.cz.                  172800 IN A 192.168.6.3
eieio.iii.forftw.ggtt.x.cz.xx.a.xx.c.cz.a.example.c.ns.nic.cz.                  172800 IN AAAA fc00::c5ad:c5ad:c5ae:c5ae
eieio.iii.nic.x.cz.xx.a.xx.c.cz.a.example.c.example.forftw.iirr.chld.kasuar.cz. 172800 IN A 192.168.6.3
eieio.iii.nic.x.cz.xx.a.xx.c.cz.a.example.c.example.forftw.iirr.chld.kasuar.cz. 172800 IN AAAA fc00::c5ad:c5ad:c5ae:c5ae
[...]

;; Query time: 46 msec
;; SERVER: 192.168.5.3#53(192.168.5.3)
;; WHEN: Wed Oct 02 17:18:18 EDT 2013
;; MSG SIZE  rcvd: 2305
```

The time window

- An attack window opens when TTL on an attacker's domain GLUE record expires
  - TTL can be replaced to 1 second from within the attacker's zone by a DNAME chain trick
- The time window (aka RTT between resolver and authoritative):
  - Starts with first query received by resolver
    - Resolver queries the authoritative server
  - Stops with first delegation query received
  - Every EXTRA query within the time window produce an additional query by the resolver to the authoritative server

Mixing it all together

- Pre-load the fragments into victim's cache
  - Spread them evenly into 2^16 space
- Burst enough queries into the time window with an identical second fragment in the response
  - The non-spoofed responses won't interfere with our non-existent name in spoofed RDATA
- (Under Linux) the IP fragmentation cache can hold 63 fragments per IP
  - IP-ID is not random, but sequential jump (RFC 4413)
  - Roughly 1040 queries are needed to fit into the time window

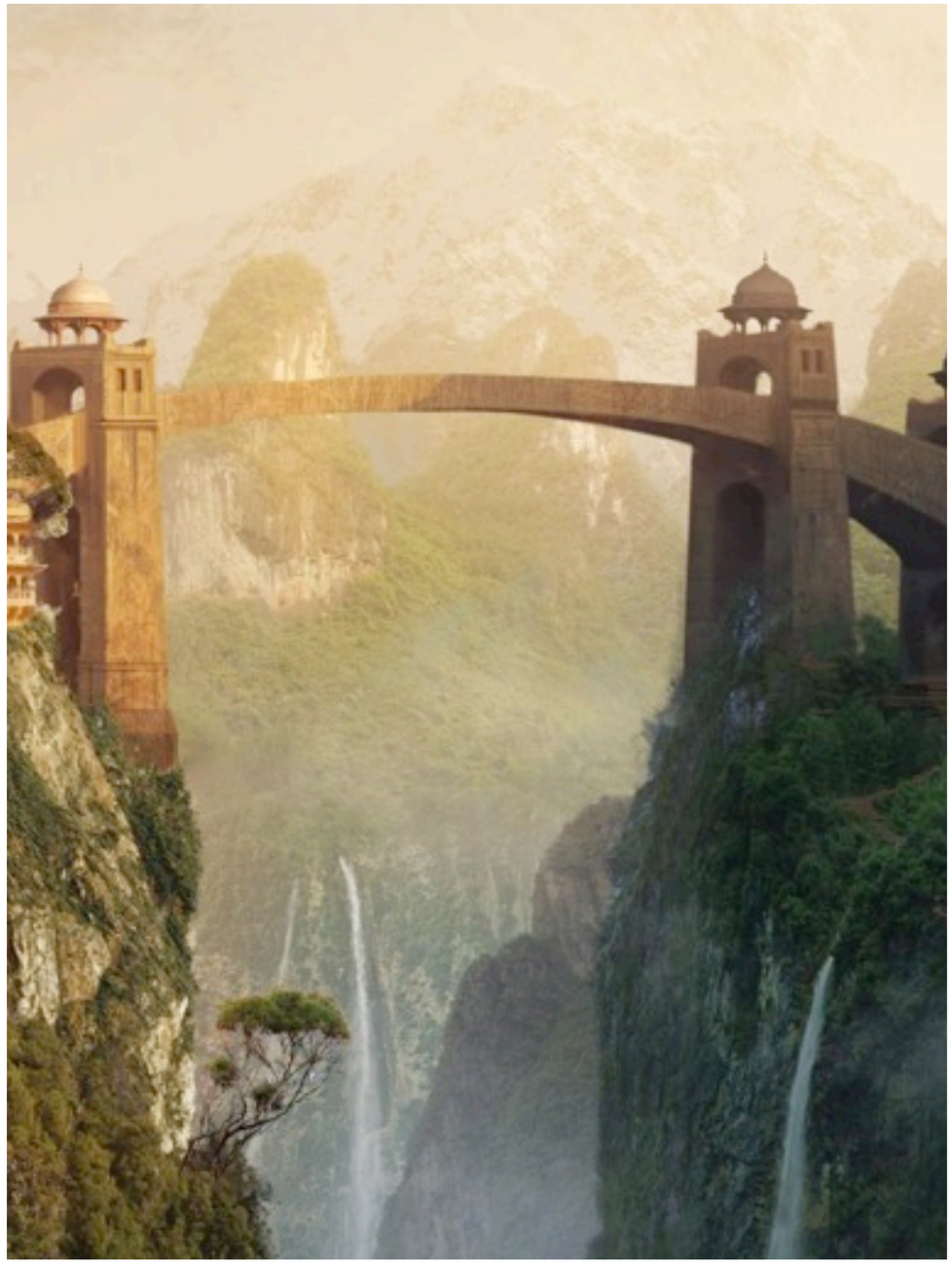
Bandwidth needed

- DNS (DNAME trick) response to trigger new query is roughly 1000 bit (160 bytes)
  - 1000*1040 bits to fit into the time window
  - RTT 1ms = 1Gbps
  - RTT 10ms = 100Mbps

Quest for Shangri-La

- Deploy BCP38 everywhere

- Enable DNSSEC everywhere



Mitigation

- Ignore “ICMP fragmentation needed” at authoritative server
- Set EDNS0 buffer size under MTU value (preferably on both sides)
- The answer will truncated before it can fragment and TCP will be used

Thank you

mail: ondrej.sury@nic.cz