Network Working Group K. Fujiwara
Internet-Draft JPRS
Intended status: Informational February 15, 2019
Expires: August 19, 2019

Measures against cache poisoning attacks using IP fragmentation in DNS
draft-fujiwara-dnsop-fragment-attack-00

Abstract

Researchers proposed DNS cache poisoning attacks using IP fragmentation. This document shows feasible and adequate measures at full-service resolvers against these attacks. To protect resolvers from these attacks, avoid fragmentation (limit requestor's UDP payload size to 1220/1232), drop fragmented UDP DNS responses and use TCP at resolver side.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction 2
2. Current status 3
3. Possible measures 4
3.1. Use DNSSEC 4
3.2. Limit requestor's UDP payload size to 1220/1232 on IPv6 . 4
3.3. Limit requestor's UDP payload size to 512 5
3.4. Drop path MTU discovery or filter ICMP related to path MTU discovery 5
3.5. Drop all fragmented packets 5
3.6. Drop fragmented UDP DNS responses at full-service resolvers 5
3.7. Use TCP only 6
4. Proposal 6
5. Example firewall configuration 6
6. IANA Considerations 7
7. Security Considerations 7
8. Acknowledgments 7
9. Change History 7
9.1. 00 7
10. References 7
10.1. Normative References 7
10.2. Informative References 8
Appendix A. How to know path MTU value 8
Appendix B. How to generate crafted ICMP packets 8
B.1. Example of crafted ICMP Need Fragmentation and DF set packet 8
B.2. Example of crafted ICMPv6 Packet Too Big 9
Author's Address 10

1. Introduction

"Fragmentation Considered Poisonous" [Herzberg2013] proposed effective off-path DNS cache poisoning attacks using IP fragmentation. The attacks mainly depend on the use of UDP to retrieve long DNS responses, resulting in packet fragmentation. Recent full-service resolvers use good randomness for query source port numbers and ID field in DNS header to prevent cache poisoning attacks by off-path attackers. However, IP fragmentation is performed by OS kernel or routers that operators of DNS servers cannot control, and the query source number and ID field in DNS header exist only in first fragment. The attack depends on poor randomness of "Identification" field generated by IP fragmentation and some bugs in IP reassembly code. Attacker can know path MTU size

between authoritative servers and victim full-service resolvers, and responses from authoritative servers. If attackers know generation algorithm of "Identification" field, they can generate crafted second fragment packets that will be accepted by victim full-service resolvers.

[Hlavacek2013] also discussed the attacks and pointed that attackers can control path MTU values between some authoritative servers and victim full-service resolvers by sending crafted ICMP packets (Fragmentation needed and DF set, or ICMPv6 Packet Too Big). The author proposed that the defense is DNSSEC and workarounds are ignoring ICMP type=3 code=4 fragmentation needed and DF set, limit response size and set EDNS0 buffer size fit to MTU value.

And more, "Domain Validation++ For MitM-Resilient PKI" [Brandt2018] proved that off-path attackers can intervene in path MTU discovery [RFC1191] to perform intentionally fragment responses from authoritative servers. They also proved that they poisoned CAs' full-service resolvers and successfully issued some certificates.

As a result, we cannot trust all fragmented UDP packets and path MTU discovery.

By the way, TCP is considered strong against fragmentation attacks because TCP has sequence number and acknowledgement number in each sequence.

This document describes resolver side countermeasure of cache poisoning attacks using IP fragmentation.

2. Current status

DNS cache poisoning attacks using IP fragmentation are executed by combining the following two attacks. Path MTU attack targets are all authoritative DNS servers. Cache poisoning attack targets are full-service resolvers.

[Brandt2018] showed that Linux version 3.13 and older versions are vulnerable to crafted ICMP fragmentation needed and DF set packet and off-path attackers can set some of authoritative servers' path MTU value to 296.

The author tested Linux version 2.6.32, 4.18.20 and FreeBSD 12.0. Linux 2.6.32 accepts crafted "ICMP Need Fragmentation and DF set" packet and path MTU decreased to 552. Linux 2.6.32, Linux 4.18.20 and FreeBSD 12.0 accept crafted "ICMPv6 Packet Too Big" packet and path MTU decreased to 1280.

Linux version 4.18.20 may ignore crafted ICMP packet.

FreeBSD and NetBSD accept "ICMP Need Fragmentation and DF set" packet related to established TCP and ignore "ICMP Need Fragmentation and DF set" packet related to UDP.

Then, off-path attackers can decrease path MTU values from some IPv4 authoritative servers to 552 (or 296), and can decrease path MTU values from IPv6 authoritative servers to 1280 (minimal IPv6 MTU value).

Furthermore, off-path attackers can know path MTU value related to authoritative servers and they can generate crafted fragmented DNS responses to victim full-service resolvers.

Then, measures against these attacks at full-service resolvers is important.

OS / source	crafted ICMPv4	minimal IPv4 MTU	crafted ICMPv6	minimal IPv6 MTU
[Brandt2018]	accept	552/296	unknown	unknown
Linux 2.6.32	accept	552	accept	1280
Linux 4.18.20	ignore?		accept	1280
FreeBSD 12	ignore		accept	1280

3. Possible measures

3.1. Use DNSSEC

DNSSEC is a measure against cache poisoning attacks. However, there are many unsigned zones and full-service resolver operator need to consider these zones.

"Use DNSSEC" requires both authoritative side and resolver side support.

3.2. Limit requestor's UDP payload size to 1220/1232 on IPv6

Limiting EDNS0 requestor's UDP payload size [RFC6891] to 1220/1232 on IPv6 is countermeasure of path MTU attacks on IPv6 because minimal MTU value of IPv6 is 1280 and most of implementations ignore ICMPv6 packet too big packets whose MTU value is smaller than 1280.

3.3. Limit requestor's UDP payload size to 512

Limiting EDNS0 requestor's UDP payload size [RFC6891] to 512 may be countermeasure of path MTU attacks.

However, since most of DNSSEC responses exceed 512 octets, limiting EDNS0 requestor's UDP payload size to 512 results truncated responses and resolvers need to retry queries by TCP. It always decreases name resolution performance.

And more, [Brandt2018] showed that off-path attackers can set some of authoritative servers' path MTU cache to 296. This case, limiting EDNS0 payload size is not a countermeasure.

Section 3 of [RFC4035] defines that A security-aware name server MUST support a message size of at least 1220 octets.

3.4. Drop path MTU discovery or filter ICMP related to path MTU discovery

It is not a countermeasure of resolver side. All authoritative servers need to be changed. Changing all authoritative servers is impossible. TCP requires path MTU discovery.

3.5. Drop all fragmented packets

To avoid fragmentation attacks, "drop all fragmented packets" is one idea. However, under path MTU discovery attacks, TCP packets may be fragmented and dropped. Then, "drop all fragmented UDP packets related to DNS" is the solution.

3.6. Drop fragmented UDP DNS responses at full-service resolvers

Drop fragmented UDP DNS responses at full-service resolvers may be a countermeasure of cache poisoning attacks using IP fragmentation.

To avoid fragmentation in normal condition, use EDNS0 requestor's UDP payload size as 1220 to avoid fragmentation. 1220 is the minimal value defined by [RFC4035].

Under path MTU discovery attacks and cache poisoning attacks using IP fragmentation, UDP DNS response packets are fragmented and dropped and name resolution fails.

If resolver software retries by TCP, TCP is strong for fragmentation attacks and name resolution by TCP will success.

3.7. Use TCP only

It is believed that TCP is not vulnerable to fragmentation attacks. Unbound has "tcp-upstream" option that changes the upstream queries use TCP only for transport.

Some operators that support [RFC8078] said that they use TCP only for transport to avoid cache poisoning attacks.

The full-service resolvers of multiple CAs issuing domain validation (DV) certificates are required to withstand cache poisoning attacks, it is better to implement their full-service resolvers use TCP upstream queries only. Section 11.2 "DNS security" of [I-D.ietf-acme-acme] recommends that servers SHOULD perform DNS queries over TCP, which provides better resistance to some forgery attacks than DNS over UDP.

4. Proposal

To avoid cache poisoning attacks using IP fragmentation by full-service resolvers,

- o Full-service resolvers set EDNS0 requestor's UDP payload size to 1220. (minimal size defined by [RFC4035])

- o Full-service resolvers drop fragmented UDP responses related to DNS.

- o Full-service resolvers may retry name resolution by TCP.

5. Example firewall configuration

Linux iptables support dropping first fragment with UDP source port 53 by using m32 module. Other first fragments that is not UDP, not source port 53 are not dropped. Second and following fragments should not be dropped because they may relate to other protocols. Second fragments related to DNS will be dropped because their first fragments dropped.

```
iptables -t raw -A PREROUTING -m u32 --u32 \\
    "6&0xFFFF00FF=0x20000011&&18&0xffff=53" -j DROP

or  iptables -t raw -A PREROUTING -p udp -f -j DROP

ip6tables -A INPUT -p udp -m frag --fragfirst -m udp --sport 53 -j DROP
```

Other OSs may not handle first fragments. Then, drop all fragmented UDP packets.

On FreeBSD, 'ipfw' can drop all fragmented UDP packets (second fragments).

```
ipfw deny log udp from any to me in frag
```

6. IANA Considerations

This document has no IANA actions.

7. Security Considerations

Under path MTU discovery and fragmentation attacks, most full-service resolver software do not retry name resolution by TCP, name resolution related to attacks fails.

8. Acknowledgments

The author would like to specifically thank Mark Andrews.

9. Change History

9.1. 00

Initial version

10. References

10.1. Normative References

[RFC1191] Mogul, J. and S. Deering, "Path MTU discovery", RFC 1191, DOI 10.17487/RFC1191, November 1990, <https://www.rfc-editor.org/info/rfc1191>.

[RFC4035] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Protocol Modifications for the DNS Security Extensions", RFC 4035, DOI 10.17487/RFC4035, March 2005, <https://www.rfc-editor.org/info/rfc4035>.

[RFC6891] Damas, J., Graff, M., and P. Vixie, "Extension Mechanisms for DNS (EDNS(0))", STD 75, RFC 6891, DOI 10.17487/RFC6891, April 2013, <https://www.rfc-editor.org/info/rfc6891>.

10.2. Informative References

[Brandt2018]
Brandt, M., Dai, T., Klein, A., Shulman, H., and M. Waidner, "Domain Validation++ For MitM-Resilient PKI", Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security , 2018.

[Herzberg2013]
Herzberg, A. and H. Shulman, "Fragmentation Considered Poisonous", IEEE Conference on Communications and Network Security , 2013.

[Hlavacek2013]
Hlavacek, T., "IP fragmentation attack on DNS", RIPE 67 Meeting , 2013, <https://ripe67.ripe.net/presentations/240-ipfragattack.pdf>.

[I-D.ietf-acme-acme]
Barnes, R., Hoffman-Andrews, J., McCarney, D., and J. Kasten, "Automatic Certificate Management Environment (ACME)", draft-ietf-acme-acme-18 (work in progress), December 2018.

[RFC8078] Gudmundsson, O. and P. Wouters, "Managing DS Records from the Parent via CDS/CDNSKEY", RFC 8078, DOI 10.17487/RFC8078, March 2017, <https://www.rfc-editor.org/info/rfc8078>.

Appendix A. How to know path MTU value

- Linux: ip route get <IPv4/IPv6 address>

- FreeBSD: sysctl -o net.inet.tcp.hostcache.list

Appendix B. How to generate crafted ICMP packets

Let the crafted path MTU value be cMTU.

B.1. Example of crafted ICMP Need Fragmentation and DF set packet

```
   IP header:
               +-----------------------------------------------+
               | V/HL 0x45 /  TOS any / Total Length 20+8+20+8 |
               | Identification   any  / Flags/Offset  0       |
               | TTL  any / Protocol 1 / Header checksum: calc |
               | Source Address:  attack tool address or any   |
               | Destination:       target auth server address |
               +-----------------------------------------------+

   ICMP header:
               +-----------------------------------------------+
               | Type   3  / Code   4  / Checksum: calculate   |
               | unused      0         / Next-Hop MTU: cMTU    |
               +-----------------------------------------------+

   Internet Header + 64 bits of Original Datagram:
   IP header:  +-----------------------------------------------+
               | V/HL 0x45 /  TOS any / Total Length    1420   |
               | Identification   any / Flags/Offset 0x4000(DF)|
               | TTL  any / Protocol 17/ Header checksum: calc |
               | Source Address:    target auth server address |
               | Destination:     victim full-resolver address |
               +-----------------------------------------------+
   UDP header:
               +-----------------------------------------------+
               | Source Port    53     / Destination Port: any |
               | Length   1400         / Checksum:     any     |
               +-----------------------------------------------+
```

B.2. Example of crafted ICMPv6 Packet Too Big

```
   IPv6 header:
               +-----------------------------------------------------+
               | Version/Traffic Class/Flow Label:  0x60000000       |
               |Payload Len: cMTU-40 / NextHeader 58 / HopLimit any  |
               | Source Address:      attack tool address or any     |
               | Destination Address:   target auth server address   |
               +-----------------------------------------------------+
   ICMPv6 header:
               +-----------------------------------------------------+
               | Type   2  / Code    0  / Checksum: calculate        |
               | MTU: (64bit)                                cMTU    |
               +-----------------------------------------------------+
   Fake invoking packet
   IPv6 header:
               +-----------------------------------------------------+
               | Version/Traffic Class/Flow Label:  0x60000000       |
               |Payload Len: 1400    / NextHeader 17 / HopLimit any  |
               | Source Address:      target auth server address     |
               | Destination Address: victim full-resolver address   |
               +-----------------------------------------------------+
   UDP header:
               +-----------------------------------------------------+
               | Source Port    53       / Destination Port: any     |
               | Length   1400           / Checksum:     any         |
               +-----------------------------------------------------+
   Rest:       Fill zero to end of packet
```

Author's Address

Kazunori Fujiwara
Japan Registry Services Co., Ltd.
Chiyoda First Bldg. East 13F, 3-8-1 Nishi-Kanda
Chiyoda-ku, Tokyo 101-0065
Japan

Phone: +81 3 5215 8451
Email: fujiwara@jprs.co.jp