DNS Cache Poisoning Attack Reloaded: Revolutions with Side Channels

K. Man, Z. Qian, Z. Wang, X. Zheng, Y. Huang, and H. Duan. 2020. DNS Cache Poisoning Attack Reloaded: Revolutions with Side Channels. *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security.* 1337–1350. (cited by 26)

Advisor: Dr. Scott CH Huang

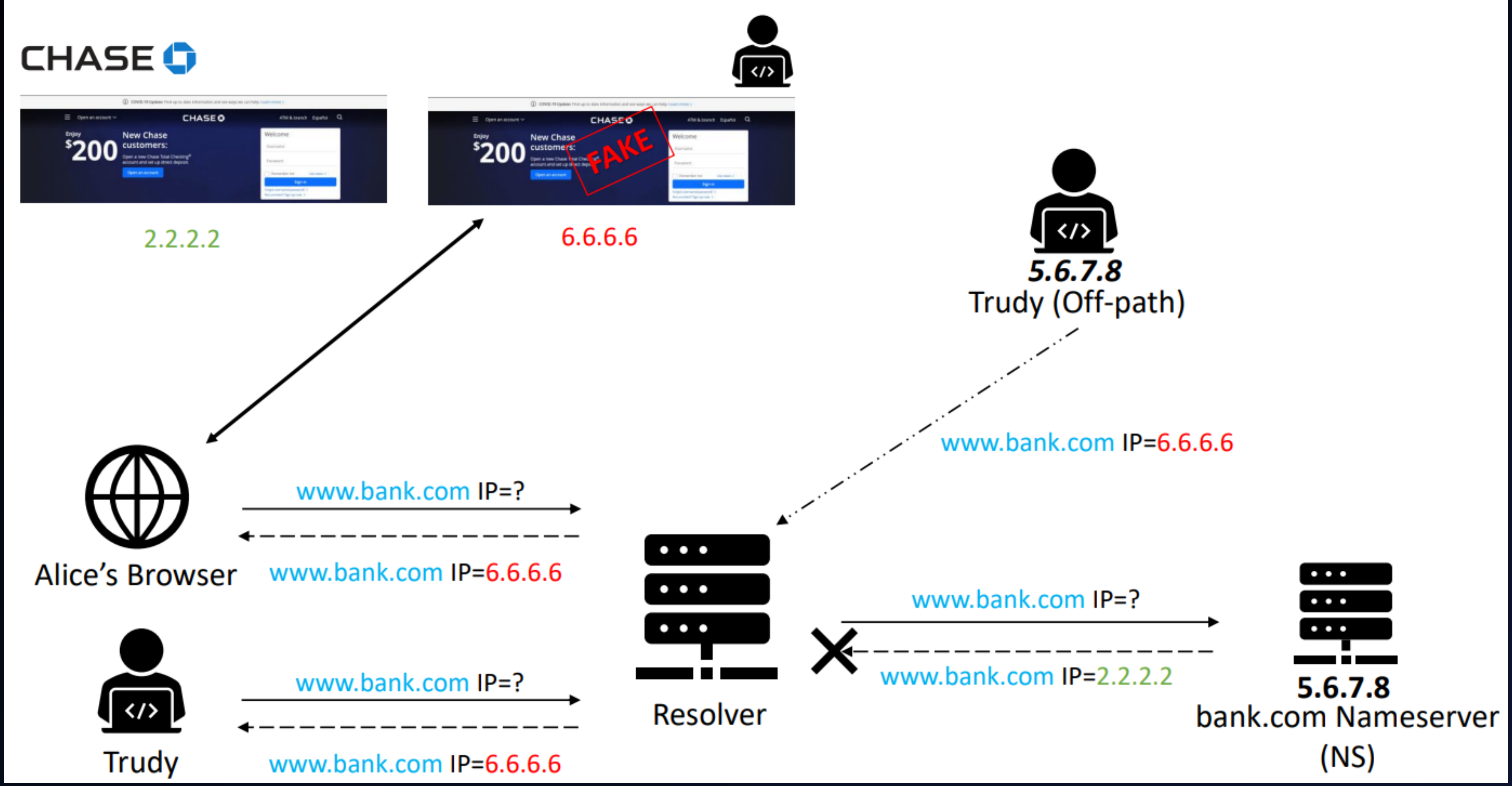
Presenter: Shao-Heng Chen

Date: May 16, 2022

Outline

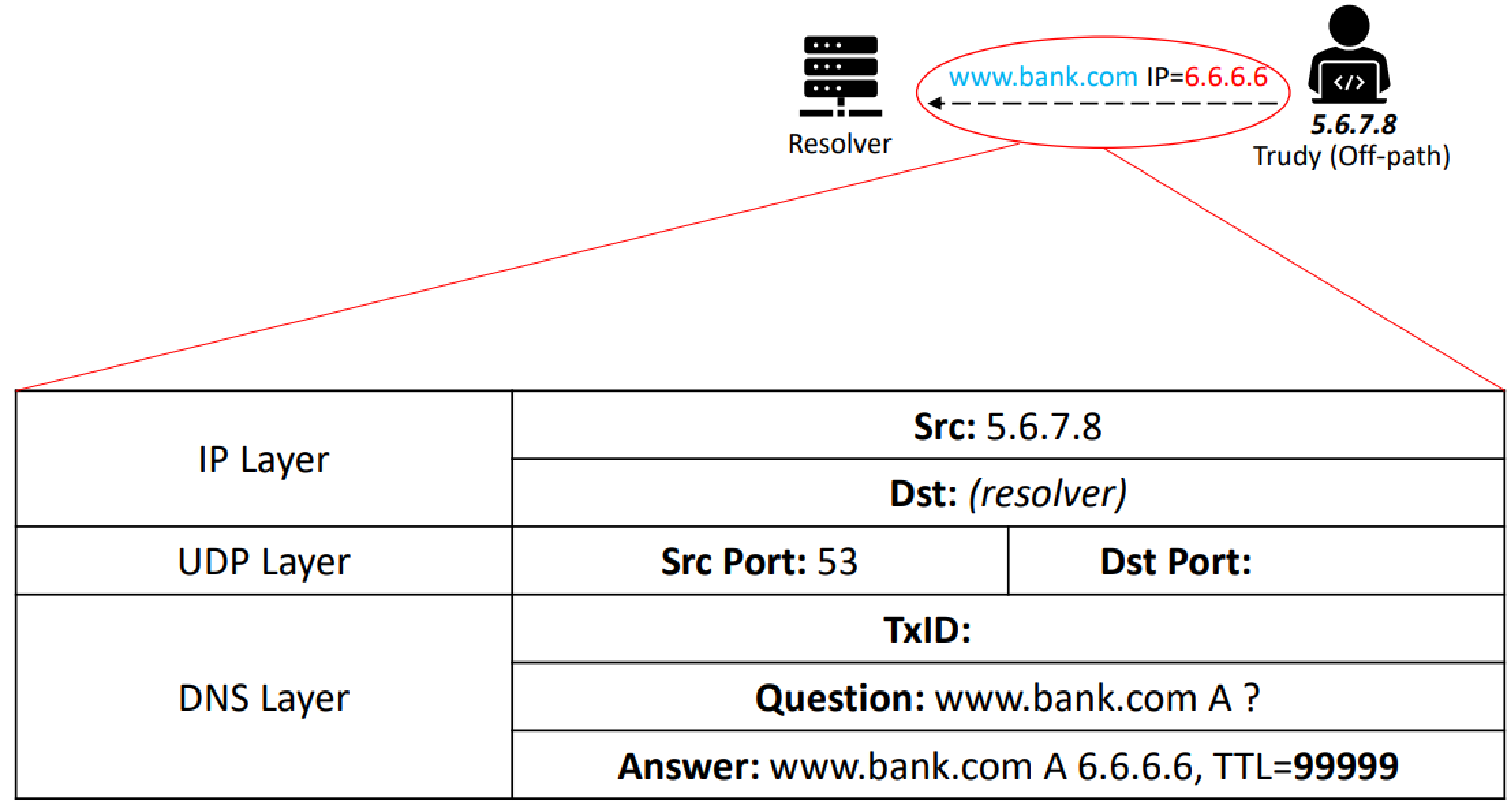
1. What is DNS Cache Poisoning Attack?
2. How to Infer the Ephemeral Port?
3. How to Extend the Attack Window?
4. Real-world Attacking Results
5. How to Defense?

What is DNS Cache Poisoning Attack?



Sniffing Traffic is the process of capturing and viewing traffic as it is passed along the network.

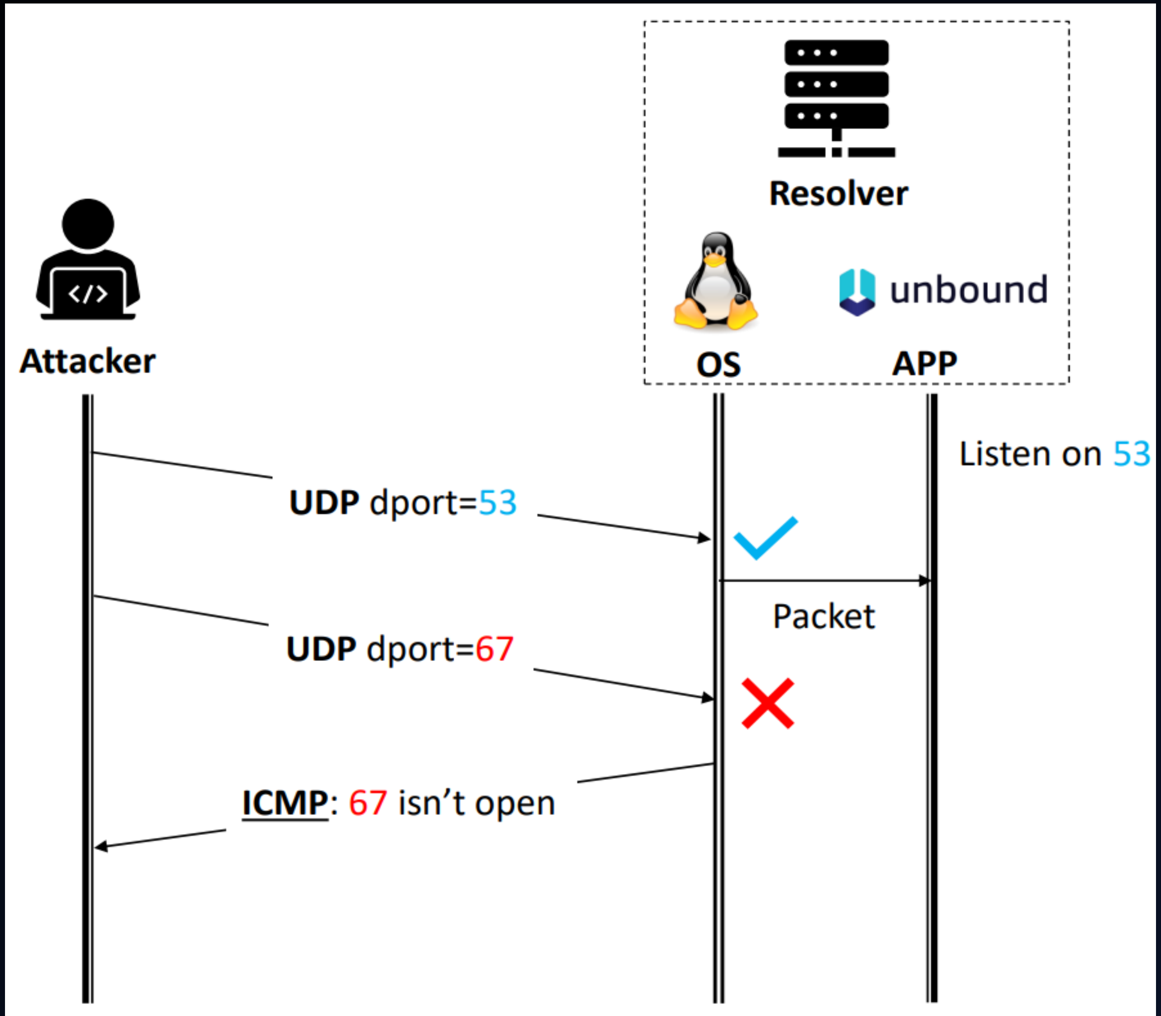
How to craft a validated DNS as an injection packet?



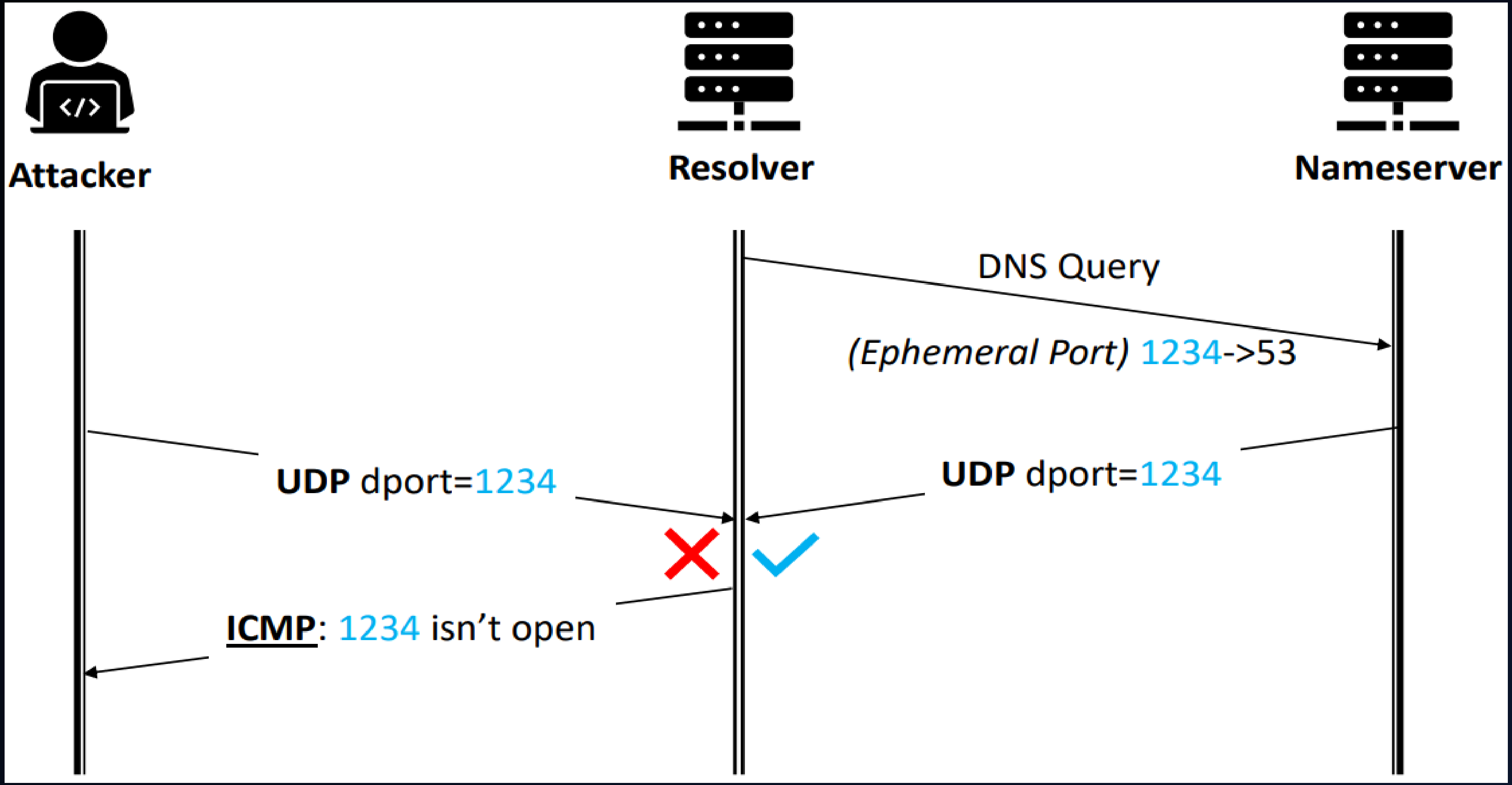
IP Layer	**Src:** 5.6.7.8	
	Dst: *(resolver)*	
UDP Layer	**Src Port:** 53	**Dst Port:**
DNS Layer	**TxID:**	
	Question: www.bank.com A ?	
	Answer: www.bank.com A 6.6.6.6, TTL=**99999**	

Ephemeral Port is a communications endpoint (port) of a transport layer protocol of the Internet protocol suite that is used for only a short period of time for the duration of a communication session.

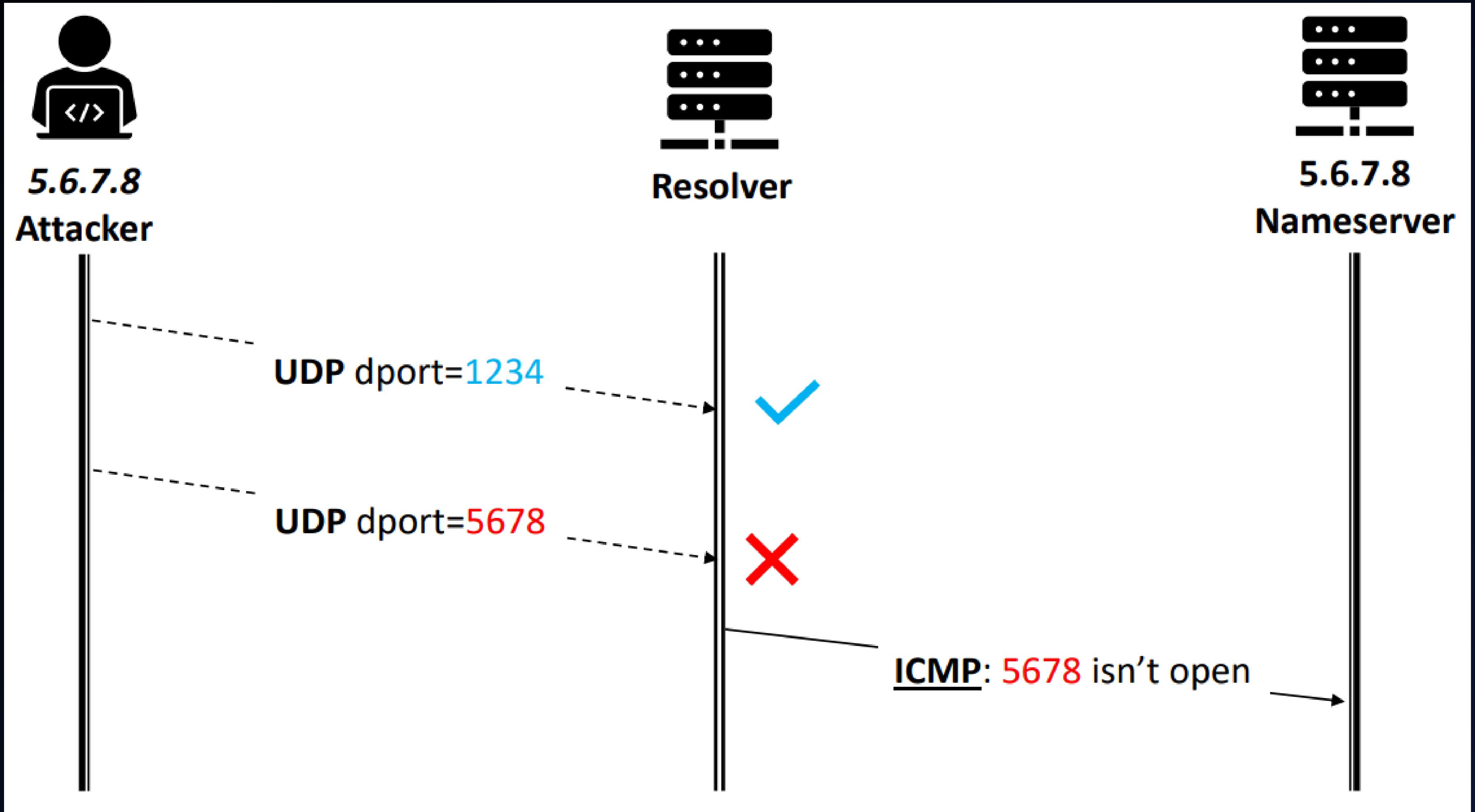
Basic Port Inference



Port Inference of Ephemeral Port

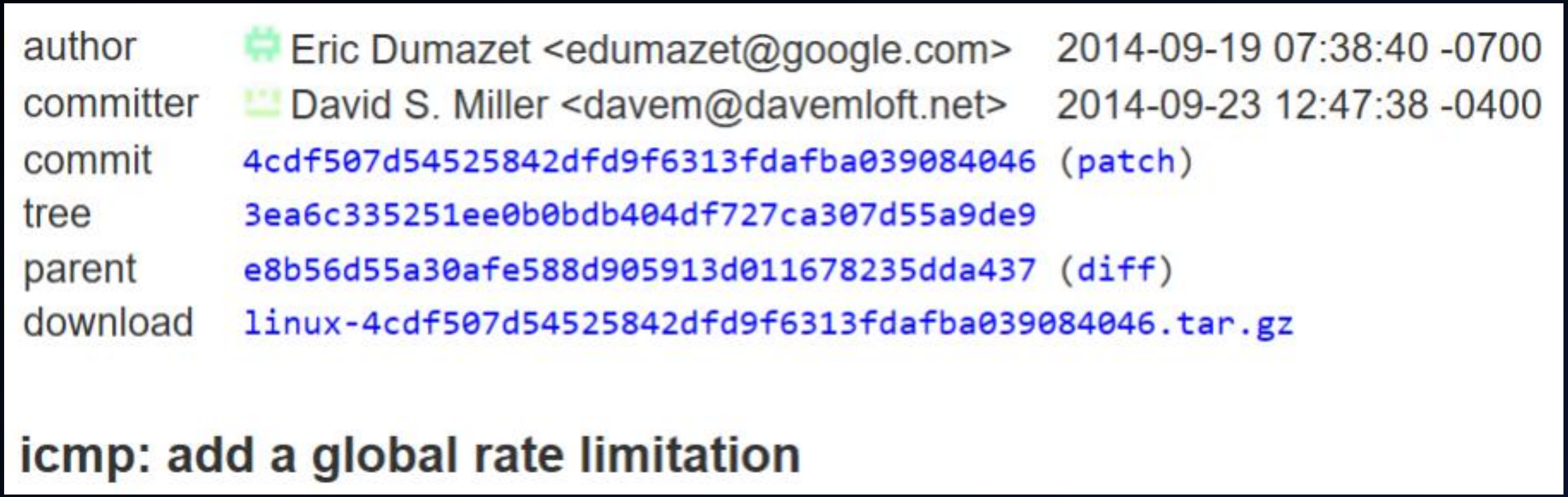


Port Inference with IP Spoofing



Port Inference with Side Channels

– ICMP Global Rate Limit: (1) Limit sending rate, (2) Shared by all IPs

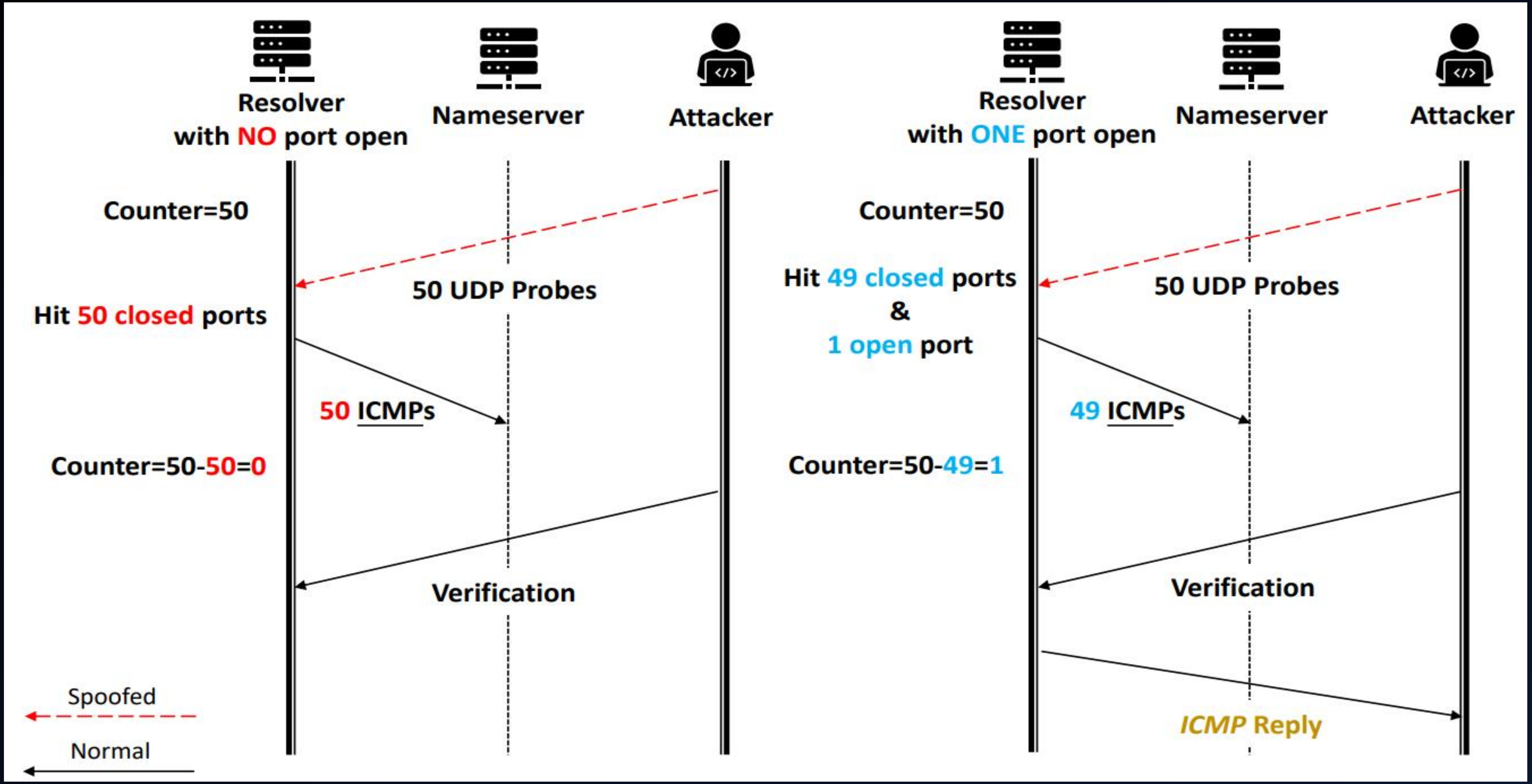


author	Eric Dumazet <edumazet@google.com>	2014-09-19 07:38:40 -0700
committer	David S. Miller <davem@davemloft.net>	2014-09-23 12:47:38 -0400
commit	4cdf507d54525842dfd9f6313fdafba039084046 (patch)	
tree	3ea6c335251ee0b0bdb404df727ca307d55a9de9	
parent	e8b56d55a30afe588d905913d011678235dda437 (diff)	
download	linux-4cdf507d54525842dfd9f6313fdafba039084046.tar.gz	

icmp: add a global rate limitation

[1] Yue Cao, Zhiyun Qian, Zhongjie Wang, Tuan Dao, Srikanth V. Krishnamurthy, and Lisa M. Marvel. 2016. Off-path TCP exploits: global rate limit considered dangerous. In Proceedings of *the 25th USENIX Conference on Security Symposium (SEC'16)*. USENIX Association, USA, 209–225.

How this Special Port Inference Works

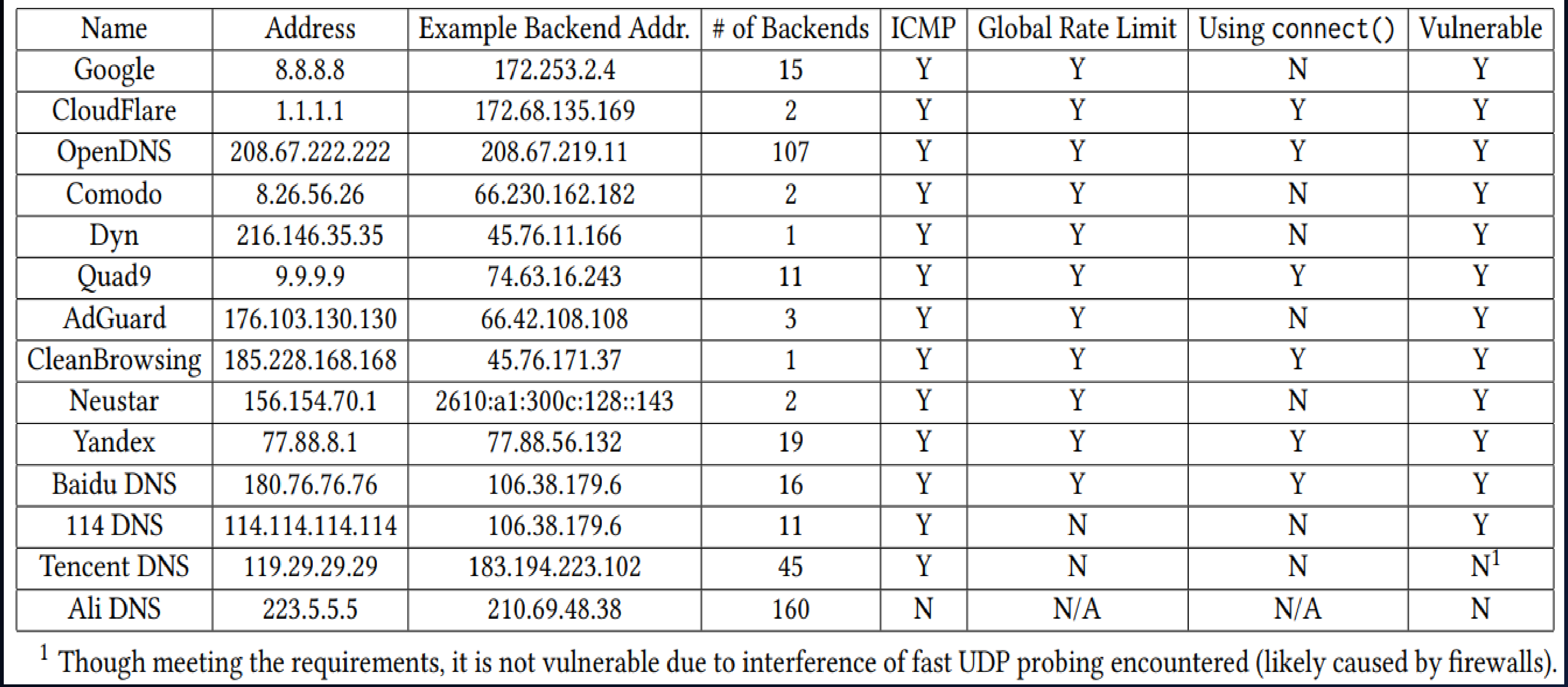


Jitter is the variation in time delay between signals transmitted and received over a network connection.

Port Inference Measurements

– Open Resolvers: 34% Vulnerable

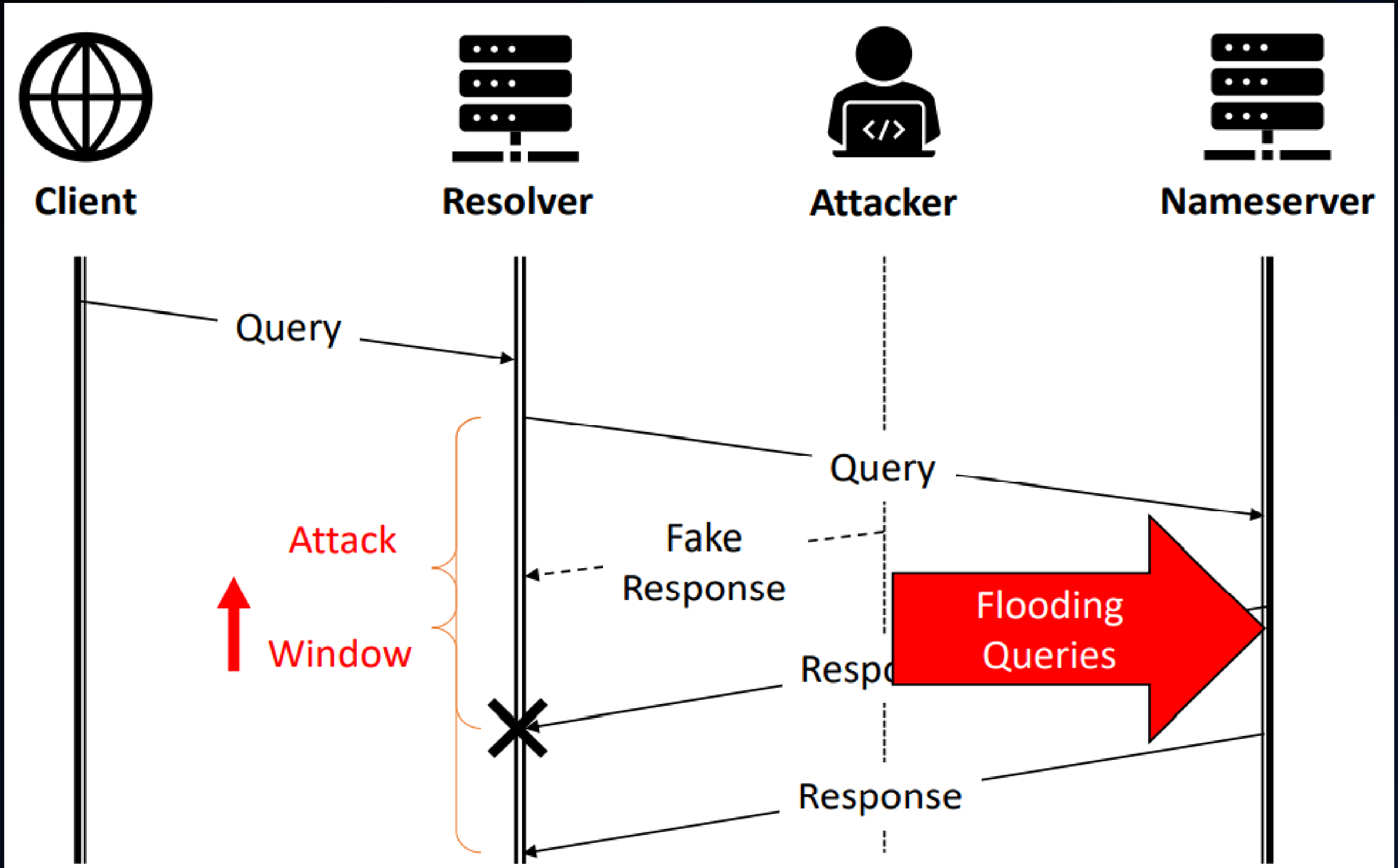
– Popular Public Resolvers: 12 / 14 Vulnerable



Name	Address	Example Backend Addr.	# of Backends	ICMP	Global Rate Limit	Using connect()	Vulnerable
Google	8.8.8.8	172.253.2.4	15	Y	Y	N	Y
CloudFlare	1.1.1.1	172.68.135.169	2	Y	Y	Y	Y
OpenDNS	208.67.222.222	208.67.219.11	107	Y	Y	Y	Y
Comodo	8.26.56.26	66.230.162.182	2	Y	Y	N	Y
Dyn	216.146.35.35	45.76.11.166	1	Y	Y	N	Y
Quad9	9.9.9.9	74.63.16.243	11	Y	Y	Y	Y
AdGuard	176.103.130.130	66.42.108.108	3	Y	Y	N	Y
CleanBrowsing	185.228.168.168	45.76.171.37	1	Y	Y	Y	Y
Neustar	156.154.70.1	2610:a1:300c:128::143	2	Y	Y	N	Y
Yandex	77.88.8.1	77.88.56.132	19	Y	Y	Y	Y
Baidu DNS	180.76.76.76	106.38.179.6	16	Y	Y	Y	Y
114 DNS	114.114.114.114	106.38.179.6	11	Y	N	N	Y
Tencent DNS	119.29.29.29	183.194.223.102	45	Y	N	N	N[1]
Ali DNS	223.5.5.5	210.69.48.38	160	N	N/A	N/A	N

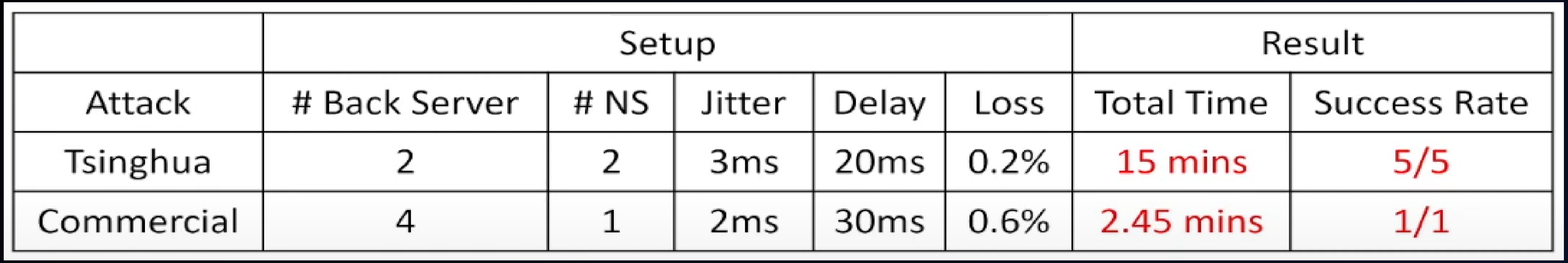
[1] Though meeting the requirements, it is not vulnerable due to interference of fast UDP probing encountered (likely caused by firewalls).

How to Extend Attack Window

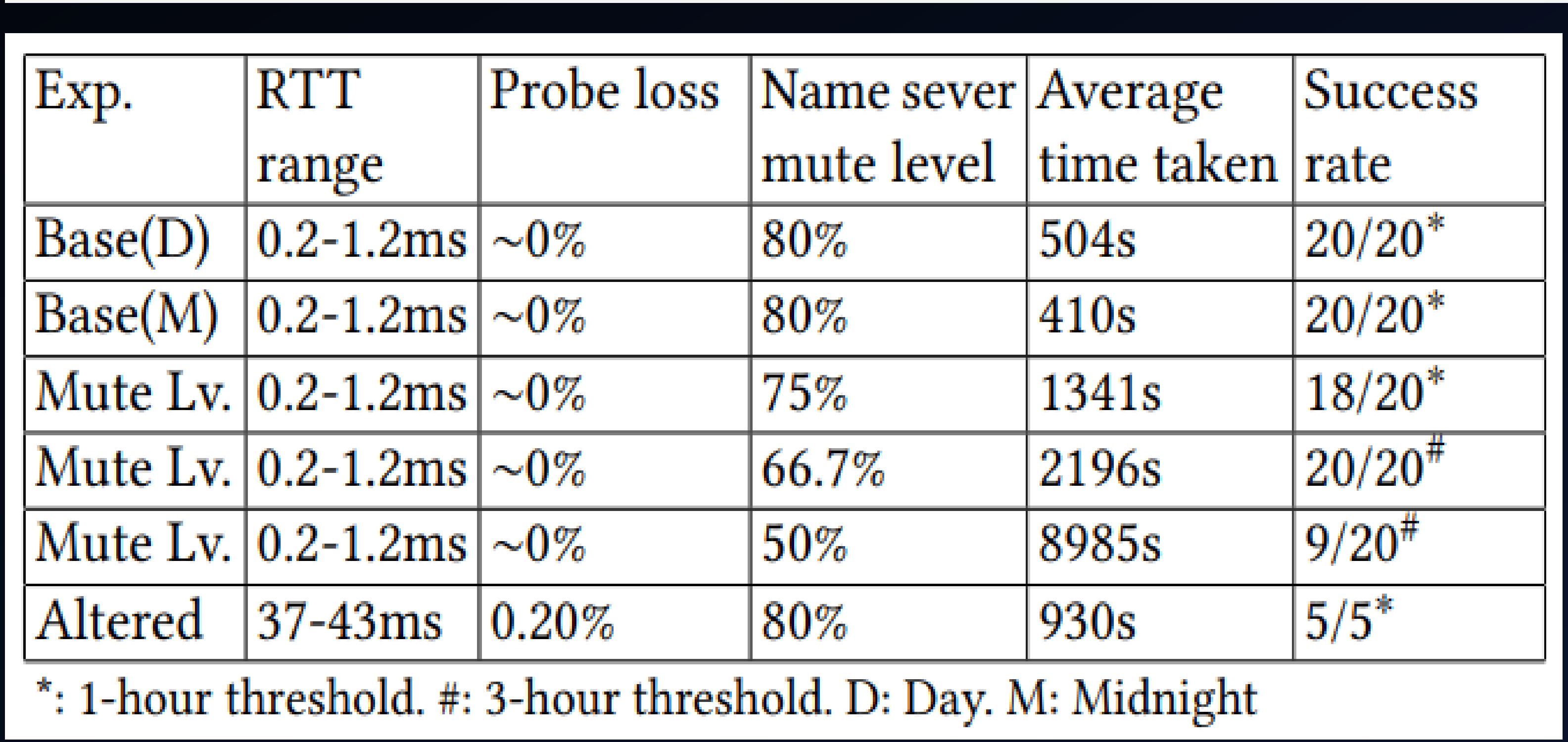


Berkeley Internet Name Domain, BIND is currently the most common DNS software on the internet.

Resolver Attack Results



	Setup					Result	
Attack	# Back Server	# NS	Jitter	Delay	Loss	Total Time	Success Rate
Tsinghua	2	2	3ms	20ms	0.2%	15 mins	5/5
Commercial	4	1	2ms	30ms	0.6%	2.45 mins	1/1



Exp.	RTT range	Probe loss	Name sever mute level	Average time taken	Success rate
Base(D)	0.2-1.2ms	~0%	80%	504s	20/20*
Base(M)	0.2-1.2ms	~0%	80%	410s	20/20*
Mute Lv.	0.2-1.2ms	~0%	75%	1341s	18/20*
Mute Lv.	0.2-1.2ms	~0%	66.7%	2196s	20/20#
Mute Lv.	0.2-1.2ms	~0%	50%	8985s	9/20#
Altered	37-43ms	0.20%	80%	930s	5/5*

*: 1-hour threshold. #: 3-hour threshold. D: Day. M: Midnight

How to Defense

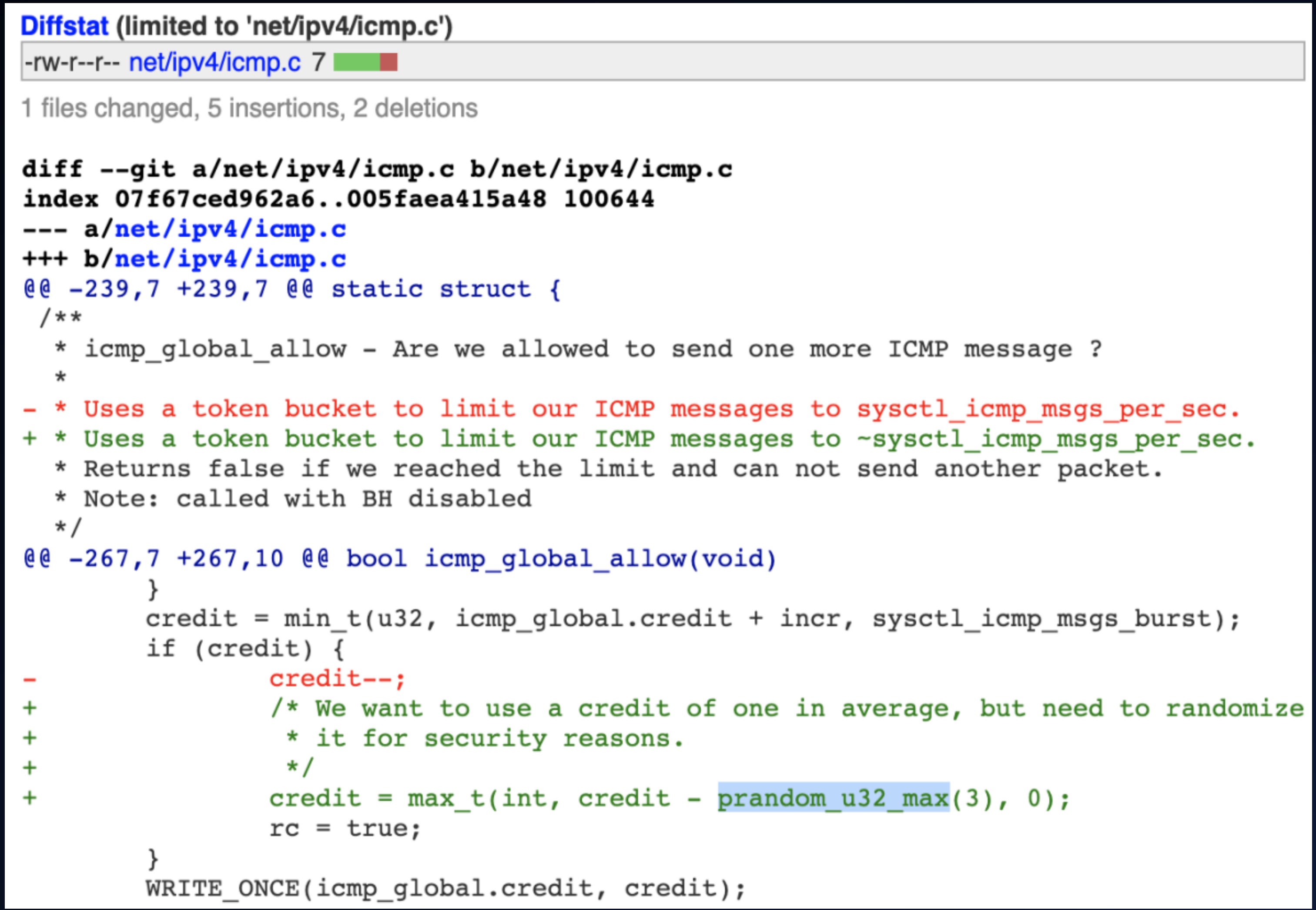
– DNSSEC

– 0x20 Encoding

– DNS cookie

– Disable ICMP port

– Randomize ICMP global rate limit



Diffstat (limited to 'net/ipv4/icmp.c')

-rw-r--r-- net/ipv4/icmp.c 7

1 files changed, 5 insertions, 2 deletions

```diff
diff --git a/net/ipv4/icmp.c b/net/ipv4/icmp.c
index 07f67ced962a6..005faea415a48 100644
--- a/net/ipv4/icmp.c
+++ b/net/ipv4/icmp.c
@@ -239,7 +239,7 @@ static struct {
 /**
  * icmp_global_allow - Are we allowed to send one more ICMP message ?
  *
- * Uses a token bucket to limit our ICMP messages to sysctl_icmp_msgs_per_sec.
+ * Uses a token bucket to limit our ICMP messages to ~sysctl_icmp_msgs_per_sec.
  * Returns false if we reached the limit and can not send another packet.
  * Note: called with BH disabled
  */
@@ -267,7 +267,10 @@ bool icmp_global_allow(void)
        }
        credit = min_t(u32, icmp_global.credit + incr, sysctl_icmp_msgs_burst);
        if (credit) {
-               credit--;
+               /* We want to use a credit of one in average, but need to randomize
+                * it for security reasons.
+                */
+               credit = max_t(int, credit - prandom_u32_max(3), 0);
                rc = true;
        }
        WRITE_ONCE(icmp_global.credit, credit);
```

Conclusion

– Side channel based on UDP port scan

– Make DNS cache poisoning attack possible again

– Effective real-world attack results