Analyzing Logic Vulnerabilities in DNS Response Pre-processing: From Kaminsky to TuDoor

Autret Lucas and Terrien Maxime

Index Terms—DNS Security, Logic Vulnerabilities, TuDoor Attack, Kaminsky Attack, SAD DNS, DNS Response Pre-processing

1 Introduction

The Domaine Name System (DNS) is one of the most critical infrastructure of the modern Internet because of it's fonction. Designed in the 1980s, this protocol translates human-readable domain names into IP addresses, making web navgiation easier for users. However, its age and widespread adoption have made it a prime target for attackers seeking to compromise Internet communications.

Over the past two decades, DNS has been the subject of numerous cache poisoning attacks. The Kaminsky attack in 2008 revealed fundamental weaknesses in the protocol, leading to multiple patches including source port randomization. Despite these countermeasures, SAD DNS in 2020 demonstrated that side-channel vulnerabilities in operating systems could bypass existing protections. More recently, the TuDoor attack (2024) has unveiled a new attack surface: logic vulnerabilities in DNS response pre-processing, where inconsistent handling of malformed packets across implementations creates exploitable conditions.

This paper analyzes the evolution of DNS attacks and examines in detail the TuDoor attack methodology notably on cache poisoning. We will first present the DNS architecture in section **??** to get a better understanding of how it works. Then we will get an overview of the history of DNS attacks with Kaminsky and SAD DNS. After that, we will describe the TuDoor attack in section **??**, its technical mechanisms, and comparative analysis with prior work. Finally, we will discuss the impact on DNS security and propose mitigation strategies in section **??**.

2 State of the Art and Historical Overview

2.1 How DNS Works

The Domain Name System (DNS) serves as a crucial component of the Internet infrastructure, that translates human-readable domain names into machine-readable IP addresses. As illustrated in Figure **??**, the resolution process relies on a chain of interactions between several distinct components to locate the correct resource.

The resolution process begins when a client application, such as a web browser, needs to resolve a hostname. It uses the operating system's *stub resolver* to initiate a request. To optimize performance and reduce the latency, this request is first sent to a pre-configured **DNS Forwarder**, often integrated into local network devices like home Wi-Fi routers. If the forwarder does not have the answenr in its cache, it forwards the query to a recursive resolver for further processing.

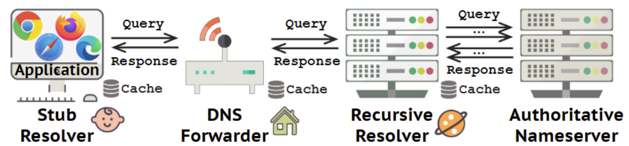

Fig. 1. General DNS resolver roles and domain name resolution process.

The **Recursive Resolver** plays an impoirtant role in the DNS resolution process. Upon receiving a query from the forwarder, it first checks its cache for a valid response. If the answer is not cached, the recursive resolver embarks on a systematic process to resolve the domain name. It begins by querying the **Root DNS Servers**, which provide referrals to the appropriate **Top-Level Domain (TLD) Servers** based on the domain's extension (e.g., .com, .org). The recursive resolver then queries the TLD servers, which in turn refer it to the **Authoritative DNS Servers** responsible for the specific domain. Finally, the authoritative server provides the requested IP address, which is relayed back through the chain to the original client.

2.2 Kaminsky attack (2008)

In 2008, security researcher Dan Kaminsky unveiled a critical vulnerability in the DNS protocol. The core issue was not just the lack of entropy in the $TxID$, but a technique that allowed an attacker to bypass the Time-To-Live (TTL) mechanism that was supposed to slow down cache poisoning attempts.

Prior to this discovery, if an attacker failed to poison a DNS cache, they would have to wait for the TTL of the cached record to expire before trying again. Kaminsky's attack exploited the fact that DNS resolvers would accept random queries for non-existent subdomains of a target domain (e.g., random123.target.com, random456.target.com). Since these subdomains do not existe in the cache, the recursive resolver is forced to query the authoritative nameserver, giving the attacker a infinite number of opportunities to flood the resolver with spoofed responses.

At the time, revolvers typically used a static source port for outgoing DNS queries, which meant that the only field an attacker needed to guess was the 16-bit $TxID$. This provided a search space of only 2^{16} (65,536) possible values. By sending a large number of spoofed DNS responses with different $TxID$ values, the attacker could eventually guess the correct one and successfully poison the cache in a matter of minutes.

Ultimately, the malicious response would be accepted by the resolver, which would then cache the incorrect mapping. This allowed the attacker to redirect users to malicious sites, intercept sensitive information, or launch further attacks. This discovery led to the implementation of **Source Port Randomization (SPR)** which increased the entropy to roughly 32 bits ($TxID$ + random 16-bit source port), making brute-forcing attacks significantly more difficult.

2.3 SAD DNS (2020)

The **SAD DNS (Side-channel AttackeD DNS) attack**, disclosed in 2020 by researchers from Tsinghua University and the University of California, Riverside, marked a critical regression in DNS security. It demonstrated a method to effectively resurrect the classic DNS cache poisoning attack by bypassing the primary mitigation implemented after the 2008 Kaminsky attack: **Source Port Randomization (SPR)**.

The success of SAD DNS relies on exploiting a subtle, yet pervasive, vulnerability in the networking stacks of modern operating systems: the predictable rate limit applied to outgoing **Internet Control Message Protocol (ICMP)** error messages, specifically the ”Port Unreachable” message. This ICMP rate limit serves as a timing side-channel that allows an off-path attacker to significantly reduce the entropy of a DNS query.

Prior to this attack, SPR had increased query entropy from 16 bits (Transaction ID, $TxID$) to 32 bits ($TxID$ plus the random 16-bit source port). The attack uses the following sequence to infer the source port:

1) **Probe Emission:** The attacker sends a large burst of spoofed UDP probe packets targeting the victim DNS recursive resolver's port range. The source IP address of these probes is spoofed to that of the target authoritative name server.
2) **ICMP Trigger:** The resolver's kernel generates an ICMP ”Port Unreachable” error message whenever a probe hits a closed port. Conversely, if the probe hits the active, open port currently used for the pending DNS query, the ICMP error is suppressed.
3) **Rate Limit Inference:** The key exploitation mechanism is the fact that the operating system applies a global rate limit to all outgoing ICMP errors. The attacker sends a final, ”unspoofed” probe to a known closed port on the resolver, observing the response time.
   - If the preceding burst of spoofed probes hit enough closed ports to deplete the global ICMP quota, the final legitimate probe will experience response delay or suppression.
   - If the burst included a hit on the active DNS source port, the corresponding ICMP error was suppressed, leaving the global quota available.
4) **Source Port Derandomization:** By analyzing the timing and successful delivery of the final probe, the attacker can systematically infer which ports in the range are currently active. This process effectively derandomizes the 16-bit source port.
5) **Cache Poisoning:** With the source port identified, the remaining entropy is reduced to the 16-bit $TxID$, enabling the attacker to easily brute-force the remaining field and inject a definitive, malicious DNS response that is accepted by the resolver.

3 TuDoor Attack

3.1 TuDoor Attack Overview

Nulla in ipsum. Praesent eros nulla, congue vitae, euismod ut, commodo a, wisi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Aenean nonummy magna non leo. Sed felis erat, ullamcorper in, dictum non, ultricies ut, lectus. Proin vel arcu a odio lobortis euismod. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Proin ut est. Aliquam odio. Pellentesque massa turpis, cursus eu, euismod nec, tempor congue, nulla. Duis viverra gravida mauris. Cras tincidunt. Curabitur eros ligula, varius ut, pulvinar in, cursus faucibus, augue.

Nulla mattis luctus nulla. Duis commodo velit at leo. Aliquam vulputate magna et leo. Nam vestibulum ullamcorper leo. Vestibulum condimentum rutrum mauris. Donec id mauris. Morbi molestie justo et pede. Vivamus eget turpis sed nisl cursus tempor. Curabitur mollis sapien condimentum nunc. In wisi nisl, malesuada at, dignissim sit amet, lobortis in, odio. Aenean consequat arcu a ante. Pellentesque porta elit sit amet orci. Etiam at turpis nec elit ultricies imperdiet. Nulla facilisi. In hac habitasse platea dictumst. Suspendisse viverra aliquam risus. Nullam pede justo, molestie nonummy, scelerisque eu, facilisis vel, arcu.

Curabitur tellus magna, porttitor a, commodo a, commodo in, tortor. Donec interdum. Praesent scelerisque. Maecenas posuere sodales odio. Vivamus metus lacus, varius quis, imperdiet quis, rhoncus a, turpis. Etiam ligula arcu, elementum a, venenatis quis, sollicitudin sed, metus. Donec nunc pede, tincidunt in, venenatis vitae, faucibus vel, nibh. Pellentesque wisi. Nullam malesuada. Morbi ut tellus ut pede tincidunt porta. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam congue neque id dolor.

3.2 Technical Details

Donec et nisl at wisi luctus bibendum. Nam interdum tellus ac libero. Sed sem justo, laoreet vitae, fringilla at, adipiscing ut, nibh. Maecenas non sem quis tortor eleifend fermentum. Etiam id tortor ac mauris porta vulputate. Integer porta neque vitae massa. Maecenas tempus libero a libero posuere dictum. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aenean quis mauris sed elit commodo placerat. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Vivamus rhoncus tincidunt libero. Etiam elementum pretium justo. Vivamus est. Morbi a tellus eget pede tristique commodo. Nulla nisl. Vestibulum sed nisl eu sapien cursus rutrum.

Nulla non mauris vitae wisi posuere convallis. Sed eu nulla nec eros scelerisque pharetra. Nullam varius. Etiam dignissim elementum metus. Vestibulum faucibus, metus sit amet mattis rhoncus, sapien dui laoreet odio, nec ultricies nibh augue a enim. Fusce in ligula. Quisque at magna et nulla commodo consequat. Proin accumsan imperdiet sem. Nunc porta. Donec feugiat mi at justo. Phasellus facilisis ipsum quis ante. In ac elit eget ipsum pharetra faucibus. Maecenas viverra nulla in massa.

Nulla ac nisl. Nullam urna nulla, ullamcorper in, interdum sit amet, gravida ut, risus. Aenean ac enim. In luctus. Phasellus eu quam vitae turpis viverra pellentesque. Duis feugiat felis ut enim. Phasellus pharetra, sem id porttitor sodales, magna nunc aliquet nibh, nec blandit nisl mauris at pede. Suspendisse risus risus, lobortis eget, semper at, imperdiet sit amet, quam. Quisque scelerisque dapibus nibh. Nam enim. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Nunc ut metus. Ut metus justo, auctor at, ultrices eu, sagittis ut, purus. Aliquam aliquam.

3.3 Mecanism of the Vulnerability

Etiam pede massa, dapibus vitae, rhoncus in, placerat posuere, odio. Vestibulum luctus commodo lacus. Morbi lacus dui, tempor sed, euismod eget, condimentum at, tortor. Phasellus aliquet odio ac lacus tempor faucibus. Praesent sed sem. Praesent iaculis. Cras rhoncus tellus sed justo ullamcorper sagittis. Donec quis orci. Sed ut tortor quis tellus euismod tincidunt. Suspendisse congue nisl eu elit. Aliquam tortor diam, tempus id, tristique eget, sodales vel, nulla. Praesent tellus mi, condimentum sed, viverra at, consectetuer quis, lectus. In auctor vehicula orci. Sed pede sapien, euismod in, suscipit in, pharetra placerat, metus. Vivamus commodo dui non odio. Donec et felis.

Etiam suscipit aliquam arcu. Aliquam sit amet est ac purus bibendum congue. Sed in eros. Morbi non orci. Pellentesque mattis lacinia elit. Fusce molestie velit in ligula. Nullam et orci vitae nibh vulputate auctor. Aliquam eget purus. Nulla auctor wisi sed ipsum. Morbi porttitor tellus ac enim. Fusce ornare. Proin ipsum enim, tincidunt in, ornare venenatis, molestie a, augue. Donec vel pede in lacus sagittis porta. Sed hendrerit ipsum quis nisl. Suspendisse quis massa ac nibh pretium cursus. Sed sodales. Nam eu neque quis pede dignissim ornare. Maecenas eu purus ac urna tincidunt congue.

Donec et nisl id sapien blandit mattis. Aenean dictum odio sit amet risus. Morbi purus. Nulla a est sit amet purus venenatis iaculis. Vivamus viverra purus vel magna. Donec in justo sed odio malesuada dapibus. Nunc ultrices aliquam nunc. Vivamus facilisis pellentesque velit. Nulla nunc velit, vulputate dapibus, vulputate id, mattis ac, justo. Nam mattis elit dapibus purus. Quisque enim risus, congue non, elementum ut, mattis quis, sem. Quisque elit.

3.4 Comparative Analysis with Previous Attacks

Maecenas non massa. Vestibulum pharetra nulla at lorem. Duis quis quam id lacus dapibus interdum. Nulla lorem. Donec ut ante quis dolor bibendum condimentum. Etiam egestas tortor vitae lacus. Praesent cursus. Mauris bibendum pede at elit. Morbi et felis a lectus interdum facilisis. Sed suscipit gravida turpis. Nulla at lectus. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Praesent nonummy luctus nibh. Proin turpis nunc, congue eu, egestas ut, fringilla at, tellus. In hac habitasse platea dictumst.

Vivamus eu tellus sed tellus consequat suscipit. Nam orci orci, malesuada id, gravida nec, ultricies vitae, erat. Donec risus turpis, luctus sit amet, interdum quis, porta sed, ipsum. Suspendisse condimentum, tortor at egestas posuere, neque metus tempor orci, et tincidunt urna nunc a purus. Sed facilisis blandit tellus. Nunc risus sem, suscipit nec, eleifend quis, cursus quis, libero. Curabitur et dolor. Sed vitae sem. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Maecenas ante. Duis ullamcorper enim. Donec tristique enim eu leo. Nullam molestie elit eu dolor. Nullam bibendum, turpis vitae tristique gravida, quam sapien tempor lectus, quis pretium tellus purus ac quam. Nulla facilisi.

4 Discussion and Conclusion

4.1 Impact on DNS Security

Duis aliquet dui in est. Donec eget est. Nunc lectus odio, varius at, fermentum in, accumsan non, enim. Aliquam erat volutpat. Proin sit amet nulla ut eros consectetuer cursus. Phasellus dapibus aliquam justo. Nunc laoreet. Donec consequat placerat magna. Duis pretium tincidunt justo. Sed sollicitudin vestibulum quam. Nam quis ligula. Vivamus at metus. Etiam imperdiet imperdiet pede. Aenean turpis. Fusce augue velit, scelerisque sollicitudin, dictum vitae, tempor et, pede. Donec wisi sapien, feugiat in, fermentum ut, sollicitudin adipiscing, metus.

4.2 Causes and Mitigations

Donec vel nibh ut felis consectetuer laoreet. Donec pede. Sed id quam id wisi laoreet suscipit. Nulla lectus dolor, aliquam ac, fringilla eget, mollis ut, orci. In pellentesque justo in ligula. Maecenas turpis. Donec eleifend leo at felis tincidunt consequat. Aenean turpis metus, malesuada sed, condimentum sit amet, auctor a, wisi. Pellentesque sapien elit, bibendum ac, posuere et, congue eu, felis. Vestibulum mattis libero quis metus scelerisque ultrices. Sed purus.

4.3 Conclusion

Donec molestie, magna ut luctus ultrices, tellus arcu nonummy velit, sit amet pulvinar elit justo et mauris. In pede. Maecenas euismod elit eu erat. Aliquam augue wisi, facilisis congue, suscipit in, adipiscing et, ante. In justo. Cras lobortis neque ac ipsum. Nunc fermentum massa at ante. Donec orci tortor, egestas sit amet, ultrices eget, venenatis eget, mi. Maecenas vehicula leo semper est. Mauris vel metus. Aliquam erat volutpat. In rhoncus sapien ac tellus. Pellentesque ligula.

References

[1] X. Li *et al.*, "TuDoor Attack: Systematically Exploring and Exploiting Logic Vulnerabilities in DNS Response Pre-processing with Malformed Packets," *2024 IEEE Symposium on Security and Privacy (SP)*, pp. 4459-4477, 2024.

[2] D. Kaminsky, "DNS Vulnerability," Black Hat USA, 2008.

[3] K. Qian *et al.*, "SAD DNS: Exploiting Weakened Trust in DNS," *ACM CCS*, 2020.

[4] H. Kopka and P. W. Daly, *A Guide to LaTeX*, 3rd ed. Harlow, England: Addison-Wesley, 1999.