Detection and Prevention of DNS Spoofing Attacks: A Critical Review and Improvement Proposal

Students

- Ido Ben Nun 209202225
- Bar Cohen 316164938

3 INTRODUCTION

4 BACKGROUND

5 ANALYSIS

6 STRENGTHS

6 WEAKNESSES

7 IMPROVEMENT SUGGESTIONS

8 CONCLUSION

Introduction

The exponential rise in cyber threats has positioned DNS spoofing as a pivotal concern within cybersecurity, primarily due to its ability to covertly redirect users to malicious domains, thereby compromising secure HTTPS connections.
DNS spoofing, a form of man-in-the-middle attack, undermines the integrity of the internet by redirecting users to malicious sites without their knowledge.

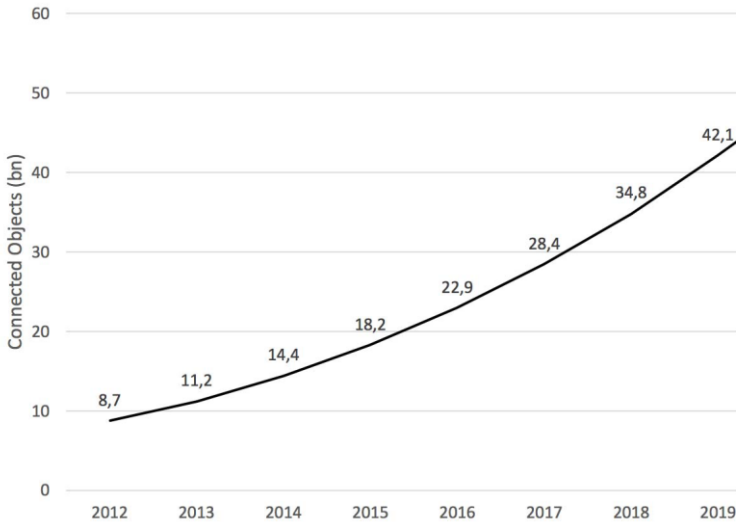

Fig. 1. Prediction growth in the number of network-connected devices up to 2020.

This project focuses on critically analyzing the paper "Detection and prevention of DNS spoofing attacks" by Artem A. Maksutov, Ilya A. Cherepanov, and Maksim S. Alekseev. The paper introduces **DNSwitch**, a tool designed to detect DNS spoofing attacks, marking a significant step forward in cybersecurity efforts to safeguard against such vulnerabilities.

The paper introduces DNS spoofing attacks as a modern threat to secure HTTPS connections, specifically targeting the SSLstrip and SSLstrip+ tools. It highlights the lack of a simple tool to detect such DNS spoofing attacks and proposes a new utility called DNSwitch to address this gap.

Background

DNS spoofing attacks manipulate the resolution of Domain Name System (DNS) queries to redirect users to fraudulent sites, posing a serious threat to online security. Traditional approaches to mitigating these attacks, such as the implementation of HTTPS and HSTS protocols, offer substantial protection but fall short in certain scenarios.

The reviewed paper builds upon the foundation of existing research, highlighting the evolution of MitM attacks facilitated by tools like SSLstrip and its more sophisticated successor, SSLstrip+. The introduction of HSTS marked a pivotal advancement, yet it too is not impervious to circumvention, as demonstrated by SSLstrip+. The authors' development of DNSwitch addresses this gap, proposing a novel method for real-time detection of DNS spoofing without the significant overhead associated with prior solutions.

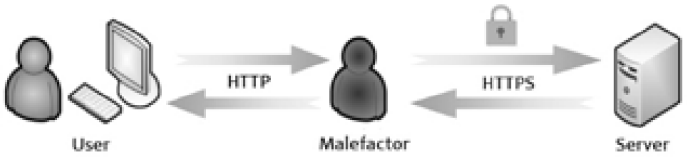

Fig. 2. MitM-attack scheme using the SSLstrip tool.

The authors provide context on the increasing importance of network security due to the growing number of internet-connected devices transmitting sensitive data. They explain how man-in-the-middle (MitM) attacks like SSLstrip work by intercepting and replacing HTTPS links with insecure HTTP links, exploiting the fact that many websites used to serve unencrypted content initially.

The HTTP Strict Transport Security (HSTS) mechanism was introduced to combat this vulnerability by forcing browsers to connect to certain websites only over HTTPS. However, the SSLstrip+ attack found a way to bypass HSTS using DNS spoofing and homograph domain name spoofing techniques.

Analysis

The methodology employed by Maksutov, Cherepanov, and Alekseev revolves around the DNSwitch utility, which operates by comparing DNS responses from a trusted server with those from the system's default DNS server. This dual-check system allows for the immediate detection of discrepancies indicative of spoofing activities.

The utility's performance in test environments showcases its efficacy in identifying altered DNS responses, thus providing an effective defense mechanism against DNS spoofing. However, while the DNSwitch utility showcases promising results in detecting spoofed DNS responses, a closer inspection reveals areas for enhancement.

The original paper's methodology, primarily the dual-check system, is innovative but lacks a comprehensive analysis on scalability and countermeasure resilience. This section would benefit from a comparative analysis with current DNS security practices, evaluating DNSwitch's effectiveness across diverse network configurations and potential adversarial tactics.

The paper thoroughly analyzes the principles behind SSLstrip and SSLstrip+, including their attack vectors and the techniques they employ to bypass security mechanisms like HSTS. The authors explain how DNS spoofing is a crucial component of the SSLstrip+ attack, allowing the attacker to poison the victim's DNS cache with spoofed domain-IP mappings.

Existing methods to protect against DNS spoofing, such as IPSec tunnels and DNSCrypt, are discussed, but the authors highlight their drawbacks, such as performance overhead and the need for client-side configurations.
The authors then introduce their proposed solution, the DNSwitch utility, which aims to balance performance and security.

DNSwitch operates in two modes:

1. **Single-check mode**: Sends parallel DNS queries over insecure and secure channels, comparing the results to detect any spoofing.
2. **Monitoring mode**: Continuously validates each DNS response against a secure channel response, notifying the user or automatically switching to the secure channel if spoofing is detected.

The paper describes the implementation details of DNSwitch, including the need for a server component to establish the secure channel with a trusted DNS resolver.

Strengths

- Clear and comprehensive explanation of DNS spoofing attacks and their impact on secure connections.
- Thorough analysis of existing solutions and their limitations.
- Novel approach with DNSwitch, balancing performance and security by selectively using a secure channel for validation.
- Practical implementation details and testing results demonstrating the utility's effectiveness in detecting DNS spoofing.

Weaknesses

- Limited discussion on the potential performance impact of DNSwitch, particularly in the monitoring mode where every DNS response is validated.
- No quantitative analysis or benchmarking of DNSwitch's performance compared to existing solutions or without any protection.
- Limited discussion on the scalability and deployment challenges of DNSwitch, especially in large networks or scenarios with multiple DNS resolvers.

Improvement Suggestions

To enhance DNSwitch's utility, a multi-faceted approach is suggested:

Firstly, detailed performance benchmarking against established solutions would provide essential data to stakeholders. Secondly, integrating machine learning for adaptive threat detection could address evolving DNS spoofing tactics. Lastly, exploring DNSwitch's integration into broader cybersecurity frameworks could facilitate widespread adoption, offering layered protection against DNS-based threats.

1. Conduct comprehensive performance benchmarking and analysis of DNSwitch, comparing its overhead with existing solutions like DNSCrypt and unprotected DNS queries. This would help quantify the trade-off between security and performance and guide deployment decisions.
2. Explore more efficient techniques for DNS response validation, such as caching or sampling mechanisms, to reduce the load on the secure channel, especially in monitoring mode.
3. Investigate mechanisms to enable automatic deployment and configuration of DNSwitch across multiple DNS resolvers in large networks, ensuring consistent protection without manual intervention.
4. Extend the scope of DNSwitch to detect and mitigate other types of DNS-based attacks, such as DNS cache poisoning or amplification attacks, increasing its overall utility as a comprehensive DNS security solution.

Conclusion

The paper presents a novel approach to detecting and preventing DNS spoofing attacks with the DNSwitch utility. While it has limitations, such as the lack of performance analysis and limited scalability discussion, the proposed solution offers a promising compromise between security and performance.

By implementing the suggested improvements, such as performance benchmarking, efficient validation techniques, and enhanced scalability mechanisms, DNSwitch could become a valuable addition to the cybersecurity toolkit, complementing existing solutions and providing proactive protection against DNS-based threats.

References

1. Detection and prevention of DNS spoofing attacks.
2. DNS hijacking.