Slovak Technical University in Bratislava
Faculty of Informatics and Information Technologies

Information technology security

IPS/IDS Evasion Techniques

Author: Bc. Miroslav Todorović
Supervisor: Ing. Adam Gajdošík

November 26, 2023
2023/2024

Contents

1 Introduction

Intrusion Prevention Systems (IPS) and Intrusion Detection Systems (IDS) are vital lines of defence against hostile activity in the constantly changing field of cybersecurity. But just as security technology develop, so do enemies' strategies. The goal of this project is to analyse and comprehend the methodology that exist behind the surface of the complex world of IPS/IDS evasion tactics.

This analysis's focus is on a range of technologies that are used to secure the network and preventing the attacks. Various methods of attacking the network that are easily accessible and shared online will be examined and tested by the help of a widely know network scraping tool. In addition to gaining a theoretical understanding of various evasion techniques, the goal is to push the envelope by developing prototypes that may be used for exploitation in a safe virtual environment.

2 Analysis

A thorough examination of the issue domain is a necessary first step in the quest to comprehend and solve the complexities of IPS/IDS evasion tactics. To uncover the subtle world of cybersecurity issues presented by intrusion prevention systems (IPS) and intrusion detection systems (IDS), which acts as an entrance to the core of the matter.

The attackers constantly adapt and enhance their tactics in the dynamic world that is the modern digital environment. At this cutting edge of network security, it is critical to analyse the particular issue and uncover the nuances of evasion tactics that jeopardise the integrity of IPS/IDS systems.

2.1 Network Intrusion

Let's first examine what constitutes a network attack, or more accurately, a network intrusion, before digging into the specifics of technologies.

Computer system is compromised by an intrusion when its security is breached or when it becomes unsecure. Any unauthorised activity on a digital network is considered a network intrusion [1]. The security of networks and their data is typically compromised by network intrusions, which frequently involve the theft of priceless network resources. Threat actors are unauthorised parties who can compromise networks and endpoints. Anywhere in the world can be home to a threat actor. All they require is the ability to access the internet, a reason for doing so, and an attack technique, also known as a threat vector [1].

On Figure 1. we can see an example of a corporate network and the arrows depict from where can the attack, or a network intrusion come from. As we can see, the vulnerability is basically from every side. For that reason, certain technologies exist to prevent the attacks and guard the network.

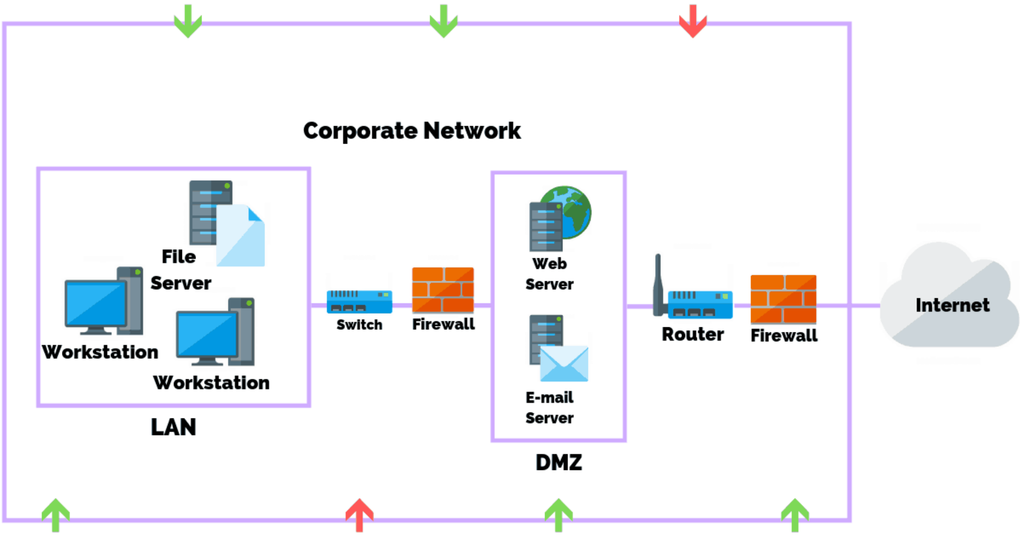

Figure 1: Corporate Network
`https://purplesec.us/intrusion-detection-vs-intrusion-prevention-systems/`

Some typical network vulnerability types that endanger an organization's cyber security include [1]:

- **Malware** - Malicious software that is usually installed on a host server or on a user's computer.
- **Social engineering attacks** - Attacks that try to trick users into providing personal data. Phishing, whaling, vishing, waterhole attacks, and tailgating are examples of these kinds of attacks.
- **Outdated or unpatched software** - Unpatched software applications put the systems they run on and possibly the entire network at risk.

Because of this, it is evident that modern intrusion attacks can be prevented only with the addition of extra security layers, as the conventional firewall and basic anti-virus software are no longer sufficient [1].

2.2 SIEM system

Security tool called security information and event management, or SIEM, assists companies in identifying and resolving possible security risks and weaknesses before they have an opportunity to impair day-to-day operations [2]. SIEM solutions assist business security teams in identifying abnormalities in user behaviour and in automating many of the laborious tasks related to threat detection and incident response through the use of artificial intelligence (AI) [2].

To detect threats and comply with data compliance regulations, all SIEM solutions, at their core, carry out some degree of data aggregation, consolidation, and sorting operations [2]. While some solutions have different capabilities, most of them have the same core features, like event correlation and analytics, compliance management and reporting, incident monitoring, security alerts, and log management [2].

2.3 Intrusion Detection System (IDS)

An intrusion detection system (IDS) monitors network traffic in order to spot potentially harmful transactions and quickly alerts users when one is detected [3]. It is software that scans a system or network for illicit activity or violations of policies. Every illegal action or violation is frequently reported to an administration or centrally documented using a SIEM system [3]. IDS keeps an eye out for harmful behaviour on a network or system and guards against users, even potential insiders, gaining unauthorised access to a computer network [3]. The goal of the intrusion detector learning job is to create a predictive model, or classifier, that can discriminate between "good (normal) connections" and "bad connections," or intrusions or attacks [3].

IDS would be positioned as follows in a network scheme:

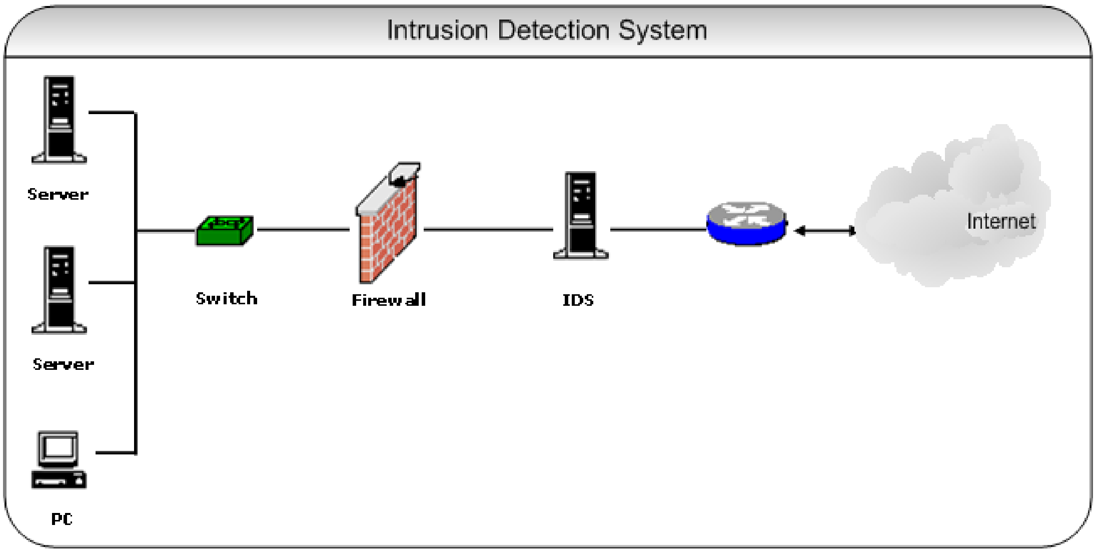

Figure 2: IDS placement
`http://imanagustrian.blogspot.com/2015/10/ids-dan-ips.html`

An IDS functions as follows [3]:

1. To identify any questionable activity, an intrusion detection system, or IDS, watches over network traffic. It examines the data passing via the network to search for trends and indications of unusual activity.
2. To find any activity that might point to an attack or intrusion, the IDS compares the network activity to a list of predefined rules and patterns.
3. The system administrator receives an alert from the IDS if it finds anything that fits one of these rules or patterns.
4. After looking into the alert, the system administrator can take action to stop any damage or additional intrusion.

2.4 IDS Types

There are five types of IDS, and the 2 most common are [4]:

- **Network Intrusion Detection System (NIDS)**
  To monitor network traffic from all connected devices, they are strategically

placed throughout the network. It observes all traffic passing through the subnet and compares that traffic to the list of known attacks [4]. The administrator can receive an alert once malicious activity is detected or an attack is detected. Installing a network intrusion detection system (NIDS) on the subnet where firewalls are situated to detect attempts to breach the firewall is an example of an NIDS [3].

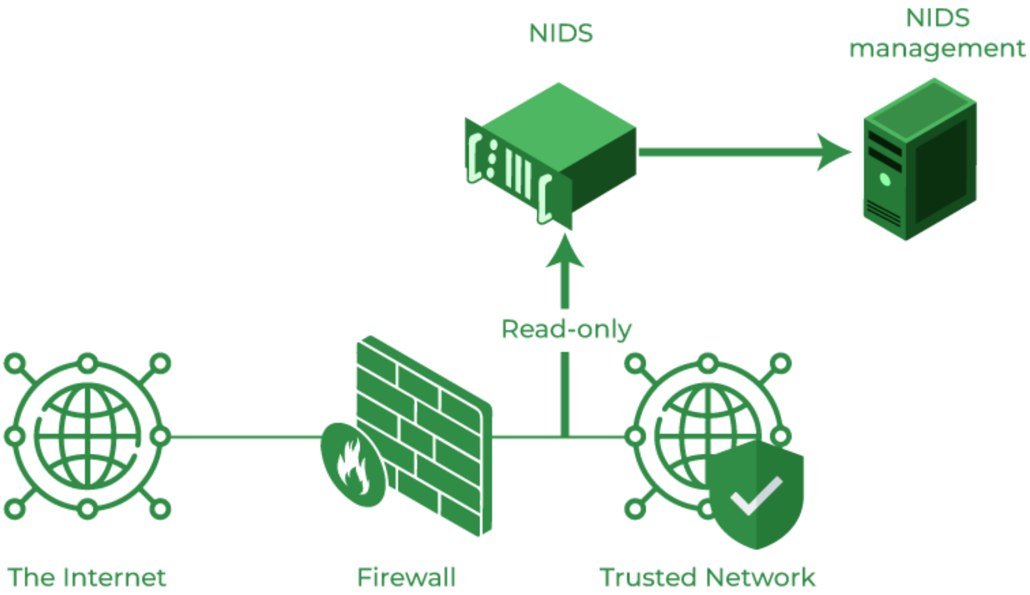

Figure 3: NIDS
`https://www.geeksforgeeks.org/intrusion-detection-system-ids/`

- **Host Intrusion Detection System (HIDS)**
  They operate on separate network hosts or devices, so only the device's incoming and outgoing packets are monitored by a HIDS, which notifies the administrator of any suspicious or malicious activity [4]. It takes a picture of the current state of the system files and contrasts it with the earlier picture, then the administrator receives a notification to look into any changes made to or deletions of the analytical system files [4]. Mission-critical machines, which are not anticipated to change their layout, are an

example of HIDS usage [3].

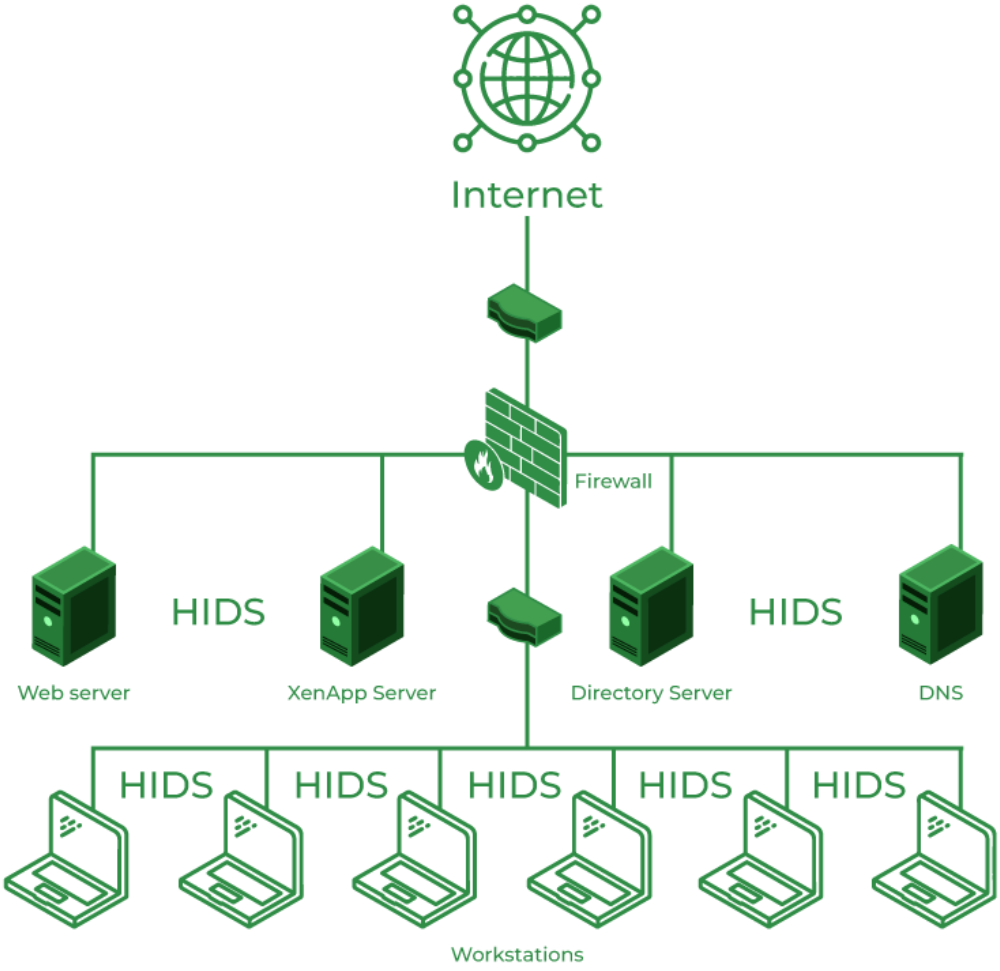

Figure 4: HIDS
`https://www.geeksforgeeks.org/intrusion-detection-system-ids/`

The following three categories apply to the remaining ones:

- **Protocol-based Intrusion Detection System (PIDS)**
  Controlling and interpreting the protocol between a user or device and the server is the responsibility of a system or agent that constantly resides at the front end of a server in a protocol-based intrusion detection sys-

tem (PIDS) [4]. By routinely observing the HTTPS protocol stream and accepting the associated HTTP protocol, it attempts to secure the web server [3]. Since HTTPS isn't encrypted, it must first reside in this interface before it can access the web presentation layer [3]. Typically, HTTPS is installed on a web server.

- **Application Protocol-based Intrusion Detection System (APIDS)**
  Generally, an application or agent that is part of a cluster of servers is called an application protocol-based intrusion detection system (APIDS) [4]. By keeping an eye on and analysing communication using application-specific protocols, it detects intrusions, for instance, when the middleware performs transactions with the web server's database, this would explicitly monitor the SQL protocol [3].
- **Hybrid Intrusion Detection System**
  Combining two or more intrusion detection system approaches results in a hybrid intrusion detection system [3]. The host agent or system data is integrated with network data in the hybrid intrusion detection system to create a comprehensive picture of the network system [3]. Compared to the other intrusion detection system, the hybrid intrusion detection system is more effective [4].

2.5 IDS Detection Methods

Additionally, there is a subset of IDS detection methods, of which the two most popular types are:

- **Anomaly detection**
  The process of comparing descriptions of what constitutes normal behaviour to actual occurrences in order to spot notable deviations is known as anomaly-based detection [5]. User, host, network connection, and application behaviour are all represented by profiles in an anomaly-based detection system (IDPS), so the profiles are created by keeping an eye on the traits of normal behaviour over time [5]. A trustworthy activity model is created using machine learning in anomaly-based IDS and any new information is compared to this profile model and deemed suspicious if it does

not match [3]. Almost any measurable attribute can be used to create profiles, such as the quantity of free memory, the number of emails sent from a specific host, or the number of login attempts. Profiles can be dynamic or static, and static profiles are always the same, while dynamic profiles are modified over time in response to network developments [5].

- **Signature detection**
  A signature is a pattern that is associated with already recognised threat. The process of matching signatures to observed events in order to identify potential incidents is known as signature-based detection [5]. Signatory-oriented IDS recognises attacks based on that pattern, or a signature in network traffic, which can be the quantity of bytes, 1s, or 0s [3]. Additionally, it makes the detection based on the malware's previously identified malicious instruction sequence [3].

2.6 IDS vs Firewall

Both intrusion detection systems (IDS) and firewalls are concerned with network security; however, while firewalls seek out and prevent intrusions, IDSs monitor the outside world for potential threats [3]. In order to prevent intrusion, firewalls limit access between networks; if an attack originates inside the network, it is not detected, in difference once an intrusion is suspected, an IDS notifies the user and sounds an alarm [3].

Organisations can detect and stop unwanted access to their network with the aid of an intrusion detection system (IDS), a potent tool. Through network traffic pattern analysis, intrusion detection systems (IDS) can detect any unusual activity and notify the system administrator. Any organization's security infrastructure can benefit from the insights and enhanced network performance that IDS offers.

2.7 Intrusion Prevention System (IPS)

An IPS is an extension to an IDS whereby a countermeasure component is introduced. This application for network security keeps an eye out for any malicious activity on the system or network. Intrusion prevention systems' primary duties include spotting malicious activity, gathering data about it, reporting it, and making an effort to thwart or terminate it [6].

IPS usually create reports, alert security administrators to significant observed events, and record information about observed events [6]. In addition, a lot of IPS can try to stop a threat before it even gets a chance to succeed, they employ a variety of reaction strategies, such as having the IPS halt the attack directly, altering the security setting, or changing the attack's content [6].

The way an IPS operates is by continuously monitoring network traffic and comparing it to attack patterns and signatures that are known to exist. The system prevents suspicious traffic from entering the network when it finds it. [6].

Although the IPS can be installed anywhere in the network, the Enterprise edge, Perimeter, and Enterprise data centre are the most popular places for installations [7]. An intrusion prevention system (IPS) can be set up to operate independently or it can be integrated into a next-generation firewall (NGFW)'s unified IPS function [7]. An intrusion prevention system (IPS) uses signatures, which can be exploit- or vulnerability-specific, to detect malicious traffic [7]. To find malicious activity, these usually use statistical anomaly detection or signature detection as mentioned above.

IPS initiates a protective measure known as a virtual patch once it detects malicious traffic that may be exploited by the network [7]. A virtual patch serves as a safeguard against attacks that take advantage of vulnerabilities, both known and undiscovered [7]. It functions by putting in place tiers of security rules and policies that stop and intercept exploits from using network paths to and from vulnerabilities, providing network-level vulnerability coverage as opposed to host-level vulnerability coverage [7].

In comparison to IDS, IPS would be positioned as follows in a network scheme:

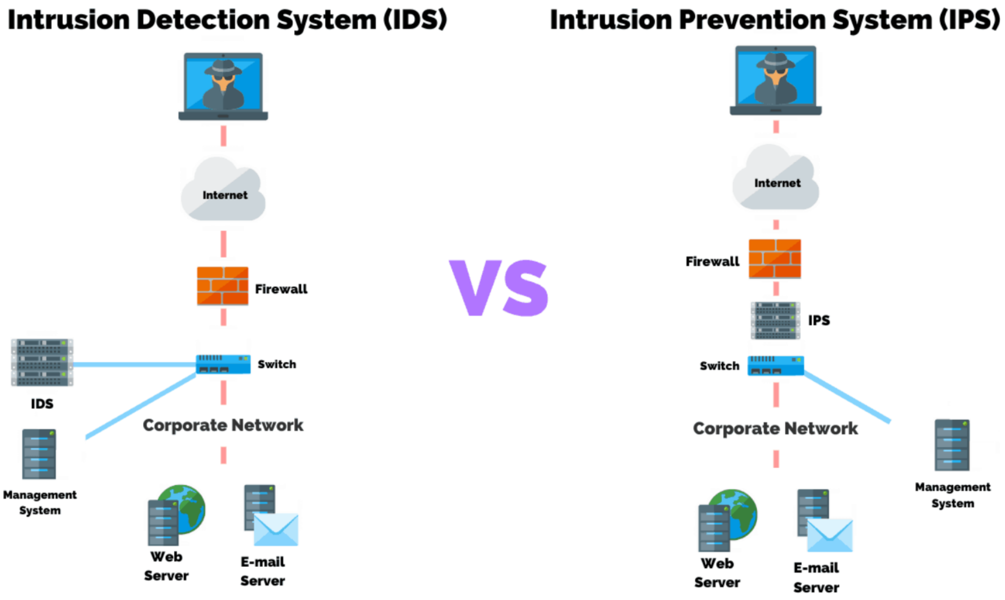

Figure 5: IDS vs IPS
`https://purplesec.us/intrusion-detection-vs-intrusion-prevention-systems/`

2.8 IPS Types

Four known categories of intrusion prevention systems exist. Every type has a distinct defence specialty of its own.

- **Network-based intrusion prevention system (NIPS)**
  These devices are positioned to monitor network traffic in real time, analysing protocols, packets, and patterns to spot unusual activity like malware infections, data breaches, and illegal access attempts [8]. A NIPS finds issues and notifies administrators of possible problems [8]. By doing this, it contributes significantly to strengthening network defences and reducing the effect of cyberattacks.

- **Wireless intrusion prevention system (WIPS)**
  These days, wireless LANs are created by a combination of encryption for wireless traffic, which is typically integrated right into wireless access points and appliances, and, as WIDS developed, so-called wireless intrusion

prevention systems (WIPS) [9]. Therefore, although both WIPS and WIDS monitor the wireless LAN radio spectrum for unauthorised devices and attacks, as suggested by their names, WIPS additionally makes an effort to prevent attacks in-line, in a manner similar to that of conventional host and network-based intrusion prevention systems [9].

- **Network behavior analysis (NBA)**
  Network behaviour analysis tracks traffic patterns and highlights unusual activity to improve network security. This is different from conventional network security operations, which protect networks from harm by using conventional methods like packet checking, signature recognition, and blocking malicious websites [10]. On the other hand, network behaviour analysis uses machine learning to find patterns in data by aggregating operational network information from multiple sources. Any sudden alteration to these patterns might indicate the existence of malicious activity [10].

- **Host-based intrusion prevention system (HIPS)**
  Endpoint device protection is usually provided by host-based intrusion prevention systems [11]. The HIPS tool can do a number of things when it finds malicious activity, such as alerting the user of the computer, recording the malicious activity for further examination, reconnecting the network, deleting malicious packets, and preventing further communication from the questioned IP address [11]. Users of certain host intrusion prevention systems can forward suspicious code and malicious activity logs directly to the vendor for identification and analysis [11]. The majority of host intrusion prevention systems detect malicious activity by utilising signatures [11].

2.9 Attack Types

Different kinds of attacks must be handled by an IPS security solution, including [7]:

- **Address Resolution Protocol (ARP) Attacks**
  There are two types of ARP attacks, and you might not even be aware that a malicious developer is trying to access sensitive data by finding vulnerabilities and sneaking in [12].

IPS Technology Type	Types of Malicious Activity Detected	Scope per Sensor	Strengths
Network-Based	Network, transport, and application TCP/IP layer activity	Multiple network subnets and groups of hosts	Only IDPS which can analyze the widest range of application protocols;
Wireless	Wireless protocol activity; unauthorized wireless local area networks (WLAN) in use	Multiple WLANs and groups of wireless clients	Only IDPS able to predict wireless protocol activity
NBA	Network, transport, and application TCP/IP layer activity that causes anomalous network flows	Multiple network subnets and groups of hosts	Typically more effective than the others at identifying reconnaissance scanning and DoS attacks, and at reconstructing major malware infections
Host-Based	Host application and operating system (OS) activity; network, transport, and application TCP/IP layer activity	Individual host	Can analyze activity that was transferred in end-to-end encrypted communications

Table 1: IPS Techonlogy Strengths Comparisons
`https://www.geeksforgeeks.org/intrusion-prevention-system-ips/`

1. **ARP Spoofing** - An attacker uses counterfeit ARP packets to connect their MAC address to the IP address of a LAN-connected computer [12].
2. **ARP Poisoning** - Following a successful ARP spoof, a hacker modifies the ARP table of the organization to include fabricated MAC maps, thereby facilitating the spread of the virus [12].

Connecting a hacker's MAC to the LAN is the goal so as a result, any traffic intended for the compromised LAN will actually go through the attacker [12].

- **Buffer Overflow**
  Attackers overwrite an application's memory in order to take advantage of buffer overflow vulnerabilities [13]. This modifies the program's execution path, resulting in an outcome that corrupts files or divulges personal data [13]. To obtain access to IT systems, an attacker might, for instance, add extra code and provide the application new instructions for the goal of gaining sensitive information from that system [13].
- **Distributed Denial of Service (DDoS)**

An intentional attempt to obstruct regular traffic on a server, service, or network by flooding the target or its surrounding infrastructure with excessive amounts of Internet traffic is known as a distributed denial-of-service (DDoS) attack [14]. A DDoS attack can be compared, at a high level, to an unforeseen traffic jam that blocks the highway and keeps regular traffic from reaching its destination [14].

- **IP Fragmentation**
  When an IP packet is larger than the Maximum Transmission Unit (MTU) size allowed for a network path, IP fragmentation happens [15]. For transmission, routers need to divide the big packet into smaller pieces [15]. Cyberattacks known as IP fragmentation attacks use the way IP packets are broken up and reassembled as a means of getting around security measures and launching attacks [15]. Attackers alter offsets and sizes of fragmented packets in order to exploit security flaws or get around firewall regulations [15]. There are many IP Fragmentation attacks, such as: Teardrop Attack, Bonk Attack, Fragrouter Tool, Jolt2 Attack, Time-to-Live (TTL) Manipulation, Nestea Attack and SMS of Death [15].

- **Operating System (OS) Fingerprinting**
  The process of examining data packets that come from a network in an effort to extract intelligence for use in upcoming attacks is known as operating system fingerprinting (OS fingerprinting) [16]. It is easier for hackers to target known vulnerabilities when they can determine which operating system a network is running on [16]. OS Configuration attributes can also be gathered via fingerprinting from distant devices [16]. This kind of recon attack is typically the start of a longer-term, more comprehensive endeavour [16]. When hackers discover a vulnerability in an old, out-of-date, or unpatched operating system, those networks become prime targets [16].

- **Ping of Death**
  An attacker can cause a system to crash by sending oversized or malformed packets via the ping command [7].

- **Port Scanning**
  Hackers frequently utilise a port scan to find weak spots or open doors

in a network, and the attack includes locating open ports and determine whether they are receiving or sending data by using a port scan attack [17]. It can also show whether an organisation uses firewalls or other active security devices, then hackers can ascertain whether a port is in use and whether there are any potential vulnerabilities that could be exploited by looking at the response they receive when they send a message to that port [17].

- **Server Message Block (SMB) Probes**
  An attacker can use an SMB relay attack to obtain a user's NTLM hash and send it to a different networked machine. using SMB authentication while posing as the user to obtain file or shell access [18].

- **Smurf**
  A distributed denial-of-service (DDoS) attack known as a "Smurf attack" involves the attacker trying to overload a server with Internet Control Message Protocol (ICMP) packets [19]. The targeted device's spoof IP address is used to send requests to one or more computer networks, which in turn respond to the targeted server [19]. This process amplifies the initial attack traffic and may even overwhelm the target, making it unreachable [19].

- **Secure Sockets Layer (SSL) Evasion**
  This attack uses Transport Layer Security (TLS) and SSL encryption to conceal malicious content so that it can evade detection and get past network security [7].

- **SYN Flood**
  Another type of DDoS attack that tries to render a server inoperable for legitimate traffic is a SYN flood, also known as a half-open attack. The attack consumes all of the server's resources [20]. An attacker can overload a targeted server machine's ports by sending initial connection request (SYN) packets frequently [20]. This makes the targeted device respond to legitimate traffic slowly or not at all [20].

3 Evasion Techniques

At this point we know what kid of technologies exist, how they work and what kind of attacks exist. Now let's analyze types of Evasion Techniques used for IDS/IPS evasion. There is a sea of are numerous techniques intruders may use to avoid detection of those systems, however, only certain ones pose problems for IDS/IPS because they are designed to get around current detection methods.

3.1 Flooding

In order to properly capture packets, analyse traffic, and report malicious attacks, intrusion detection systems (IDSs) rely on resources like memory and processor power [21]. An attacker can make an intrusion detection system (IDS) run out of resources by flooding a network with noise traffic while it examines harmless traffic [21]. The attacker can target the system with little to no assistance from the IDS while it is preoccupied and diverted by the volume of noise traffic, so the goal of this attack is to overwhelm the detector and cause the control mechanism to fail [21]. Once a detector malfunctions, all traffic will be permitted [21]. Spoofing the authentic User Datagram Protocol (UDP) and Internet Control Message Protocol (ICMP) is a common method of triggering flooding [21]. Subsequently, the traffic flooding serves as a cover for the irregular actions of the offender. Because of this, the IDS would struggle mightily to identify malicious packets amidst a dizzying amount of traffic [21].

3.2 Fragmentation

An intruder can stay hidden thanks to fragmentation since there won't be an attack signature to recognise [21]. At the IP layer, the recipient node reconstructs fragmented packets later. The application layer receives them after that [21]. By adding new data to the fragmented packets that make up the attack, fragmentation attacks create malicious packets [21]. By breaking up their attacks into ever-tinier pieces, hackers can circumvent intrusion detection systems (IDS) and launch a surprise attack on a target host upon reassembling the packets [21].

3.3 Encryption

Network-based intrusion detection is predicated on the examination of data that is recorded during a transfer from one point on the network to another [21]. An encrypted session established between a hacker and its target host via Secure Shell (SSH), Secure Socket Layer (SSL), or a virtual private network (VPN) tunnel will prevent the intrusion detection system (IDS) from analysing the packets and permit the malicious traffic to flow through [21]. It goes without saying that this method necessitates the attacker to create a safe encrypted connection with the intended host.

3.4 Obfuscation

An increasingly common evasive tactic is obfuscation, which entails masking an attack with special characters. Control characters like the space, tab, backspace, and delete can be used with it [21]. In order to avoid the IDS, the method may also represent characters in hexadecimal format. Another efficient technique to avoid IDSs is to use Unicode representation, in which every character has a distinct value independent of the platform, application, or language [21]. An attacker could, for instance, use the Unicode character c1 to simulate a slash for a Web page request in order to get around an IDS [21]. By making a message hard to understand, obfuscation can be used to hide an attack and prevent detection. By hiding it and jeopardising readability, the goal is to decrease detectability to the point where it can be used in static analysis or reverse engineering processes [21]. For example malware can be made to evade intrusion detection systems by obfuscating it.

4 Proposing a solution to the problem

This section outlines a thorough testing methodology and solution strategy, taking a proactive approach to addressing the challenges presented by IPS/IDS evasion techniques.

In order to analyse and comprehend evasion techniques fundamentally, this project makes use of the highly adaptable Nmap utility—a industry standard for network exploration and security auditing. Nmap is the main tool of our testing approach because of its wide range of applications in network environment probing. Through its use, we hope to replicate real-world situations and examine how well IPS/IDS systems work against a range of evasion strategies.

Python script will be used and compiled to improve and expedite the testing procedure. This custom script will enable the implementation of multiple evasion strategies and offer an intuitive user interface, making it accessible to a wider range of cybersecurity professionals. The goal is to close the knowledge gap between advanced evasion techniques and useful, implementable insights.

I'll use **"http://scanme.nmap.org"** which is a collaborative and real-time environment, for my testing. People use this web service, which was kindly provided by the Nmap itself for the virtual testing ground to evaluate the resilience of IPS/IDS systems. In result conducting ethical and controlled testing on this platform, preventing any impact on real-world systems and obtaining important insights into the effectiveness of evasion techniques.

The goals of this proposition are twofold: firstly, to comprehensively analyze the IPS/IDS evasion landscape using Nmap tool, and secondly, the development of a Python script for easier use of Nmap techniques and streamlined testing, by choosing "http://scanme.nmap.org" as our target.

5 Architecture

For testing and development I used Kali Linux as a secure environment, which is an incredible tool used for penetration testing. When conducting network penetration testing, Nmap comes in very handy, it helps identify security flaws in the system in addition to providing network information, also Nmap can be used on a variety of common operating systems, such as Windows, Linux, macOS, and BSD, and is not dependent on any particular platform [22]. It has a command-line interface (CLI) and a graphical user interface (GUI), and it is simple to use, which is the reason I decided to use it in this project.

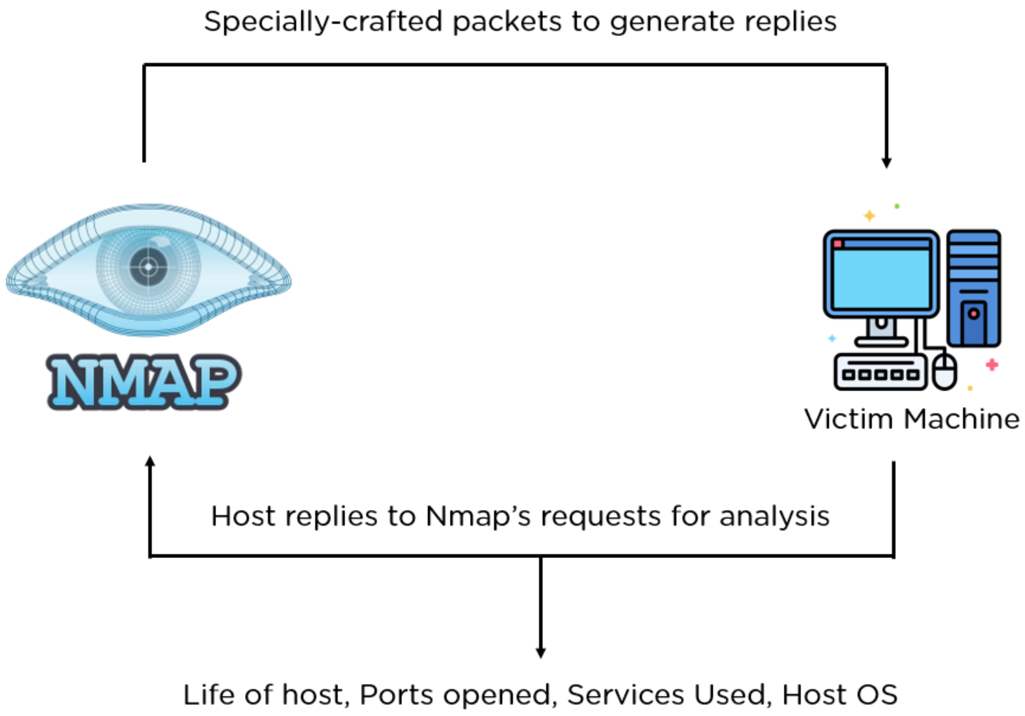
Specially-crafted packets to generate replies
NMAP
Victim Machine
Host replies to Nmap's requests for analysis
Life of host, Ports opened, Services Used, Host OS

Figure 6: How does Nmap work?
`https://www.simplilearn.com/tutorials/cyber-security-tutorial/what-is-nmap`

5.1 Evasion Techniques using Nmap

As an example, I'll work with the host target, which is available from nmap and is located at **"http://scanme.nmap.org"** and I'll try a few different combinations of the widely used techniques to get around firewalls and IPS/IDS. Therefore, in order to perform an evasion, we must first determine which ports are open and identify the live host.

After entering the command we want and the type of scan, Nmap will attempt to establish a connection with that target to spoof the port, allowing us to ascertain whether the host is utilising a particular protocol and we can potentially gain information about the device used for defence. But this is also how firewalls log traffic, so when we start probing, the firewall records it and locks the IP address. If we want to avoid that and remain anonymous, we should not do that, we must be careful to do it periodically and not be noisy. After conducting the scan, it's important to remember that if we conducted any noisy scan on a specific target, the IP address of the target will be locked in the firewall, intrusion detection system, or other system, and they will be aware that we have scanned their network.

That's why before any scan or an attempt to attack we must have a specific strategy. After finding a port we can try to perform various evasion techniques that nmap offers.

6 Comparison of attacks and methods

In this section I described which are the most common options that can be used in Nmap which can result in completely evading the Firewall and IDS. I implemented all of these options in a script which will be used for testing the evasion methods.

6.1 Source IP address spoofing

An attacker can pretend to be the source IP address (from the victim's subnet) in order to pass through IDS/firewall and appear to be a legitimate user [23].

```
Usage: nmap -S <spoofed ip> <other options> <target>
```

6.2 Source port spoofing

When scanning a target, an attacker can pretend to be the source port number in order to get around firewall rules that only permit requests from specific ports, like Port 53.

```
Usage: nmap --source-port <port no> <other options>
```

6.3 Decoys

Decoys can be used to confuse network scanners and make it harder for them to detect the real target [23]. Using decoys in Nmap scans allows the scan to appear as if it is coming from a different IP address, providing an advantage in both local area networks and across the internet [23]. Using decoy IP addresses can help evade firewalls and IDS so they can hide the true IP address of the attacker.

```
Usage: nmap -D <IP> <target>
```

6.4 Fragmentation

Packet fragmentation can be used as a firewall evasion technique by breaking down packets into smaller pieces, making it difficult for firewalls or filters to understand their contents [23]. Fragmentation can be used as a technique to evade detection by intelligent systems, as packets can be reassembled at the target [23]. Although most systems can detect fragmentation, it's worth mentioning it because if it's used in combination with some other techniques, it can be very powerful.

```
Usage: nmap -f <IP> <target>
```

6.5 MTU Manipulation

Similar to a fragmentation option but we can input the size of the MTU packets that are going be sent [23].

```
Usage: nmap --mtu 16 <IP> <target>
```

6.6 Timing Scanning

Now this option adds a lot to the evasion methods, since nmap scans can be very noisy and looking for ports, IP addresses can get the target locked from you by the Firewall or IDS/IPS [23].This option enables a timed attack with options such as:

- **T0** - Paranoid (Waits 5 minutes between sending each probes, not detected by IDS/IPS)
- **T1** - Sneaky (waits 15 seconds)
- **T2** - Polite
- **T3** - Normal
- **T4** - Aggressive
- **T5** - Insane (easily detectable)

```
Usage: nmap -T<0-5> <other options>
```

6.7 Summary

From all the above options, I will make a script which can utilize them with different settings. As a default, SYN scan will be utizilized on a TCP protocol. It is a scan where we send packets with SYN flags and then we use it to analyze the ACK response. This enables us to snoof available ports which are potential targets to evade or attack.

7 Testing method

Performing evasion and attacks on a network can have consequences as for example getting blacklisted from a certain service or a website. For that reason an attacker must plan carefully his steps, that's why he needs all the information about which ports are available, what IP address is whitelisted or safe to use to avoid detection and such. That's why for the testing method I used Wireshark, which was more than sufficient to give me information about all scans and attacks. I would perform a scan or implement some sort of evasion tactic while capturing the data on Wireshark. Which I would later use to analyze the actual response from the script and compare the captured packets on Wireshark.

All testing results would be located on the github repository for this project.

8 Description of the functions and use

My prototype uses **subprocess** module to interact with the Nmap tool for network exploration and security auditing.

- **nmap_scan function**
  This function takes two parameters:
  - **target** - The target IP address or domain to be scanned.
  - **options** - A list of Nmap command-line options.

  This function has a very straightforward functionality:
  1. Construct the Nmap command by combining the base command "nmap" the target, and the provided options.
  2. Attempt to run the Nmap command using **subprocess.run**.
  3. If the command runs successfully, print the standard output of the command.
  4. If an error occurs during the command execution (if subprocess.CalledProcessError is raised), print an error message along with the standard error output.
- **Main script**
  On the other side, Main includes more steps, which is an interaction with a user:
  1. Define default Nmap options for a TCP SYN scan ("-sS") and service version detection ("-sV").
  2. Prompt the user for the target IP address or domain.
  3. Ask the user whether to scan all ports or only the most common 100 ports. Update the options accordingly.
  4. Prompt the user to select the scan speed from a set of predefined options. Update the options with the chosen speed.
  5. Ask the user for additional scan options:
     - Packet Fragmentation

   - Decoy Scan
   - Spoofed source IP address
   - Spoofed source port
   - MTU manipulation

6. Process the user's choice and update the options accordingly.
7. Run the Nmap scan using the nmap_scan function with the target IP/domain and the constructed options.

Note: The script performs input validation and exits if an invalid option is chosen during the user prompts.

8.1 Use of prototype

Overall, the script provides a simple interface for users to customize and execute Nmap scans with various options.

Example Usage:

- The user enters the target IP or domain.
- Chooses to scan the 100 most common ports.
- Selects a scan speed option (e.g., Polite).
- Chooses additional scan options (e.g., Decoy Scan with a specified decoy IP address).
- The script constructs the Nmap command with the chosen options and runs the scan.
- The script prints the standard output of the Nmap command or displays an error message if the scan encounters an issue.

Below is an actual output from the prototype which also shows results from a decoy scan.

```
>> Start of output
```

```
Enter the target IP address or a domain: scanme.nmap.org
Scan all ports or only the most common 100?
1. All ports
2. 100 most common ports
Enter the option number (e.g. 1 or 2): 2

Select scan speed:
0. Paranoid (Every 5 minutes each probe - Not detected by IDS/IPS)
1. Slow (15 seconds between probes)
2. Polite (Scan at the default rate)
3. Normal (Scan slower than normal)
4. Agressive (Scan faster than normal)
5. Insane (Scan as fast as possible - Easily detected by IDS/IPS)
Enter the speed option (e.g., 0, 1, 2, etc.): 3

Select option:
1. Packet Fragmentation
2. Decoy Scan
3. Spoofed source IP address
4. Spoofed source port
5. MTU manipulation
Enter the option number (e.g., 1, 2, etc.): 2
Enter decoy IP address (comma-separated if multiple): 10.0.0.2
Performing scan...
[’-sS’, ’-sV’, ’-F’, ’-T3’, ’-D’, ’10.0.0.2’]
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-26 20:11 Central Europe Standar
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.29s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
Not shown: 96 closed tcp ports (reset)
PORT    STATE    SERVICE VERSION
22/tcp  open     ssh     OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protoco
25/tcp  filtered smtp
80/tcp  open     http?
```

```
427/tcp filtered svrloc
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.c
Nmap done: 1 IP address (1 host up) scanned in 139.33 seconds

>> End of output
```

9 Conclusion

The constant fight between attackers and defenders in the digital world of cybersecurity reveals a complex web of evasion strategies and security countermeasures. While looking at different ways hackers try to sneak past security, a few different methods really stood out. Each one had its own approach and strategy. Decoy attacks, with the way they try to distract the Firewall and IDS by focusing their attention elsewhere, really showed how misdirection can be an art form. Manipulating the maximum transmission unit, or MTU, gets technical with how networks talk to each other. Then there's fragmentation, which breaks up data packets in a way that hides unauthorized users inside of them.

By taking these techniques apart piece by piece, I explored some shady corners of hacking that people didn't know about before. Breaking it down like this helps us protect ourselves better in the future and learn from it. This projects also consists of prototype description, a guide how it works, and it's testing which holds the information about various evasion techniques and attacks, which are accessible on the project repository.

10 Resources

Github Repository: `https://github.com/xtodorovic/BIT-project`

References

[1] CISSP Michael Swanagan. Intrusion detection vs prevention systems: What's the difference? https://purplesec.us/intrusion-detection-vs-intrusion-prevention-systems/, Mar 2023.

[2] What is siem? https://www.ibm.com/topics/siem.

[3] Ids system evasion techniques. https://www.geeksforgeeks.org/intrusion-detection-system-ids/, Mar 2023.

[4] What is an intrusion detection system? https://www.paloaltonetworks.com/cyberpedia/what-is-an-intrusion-detection-system-ids.

[5] Vít Bukac. Ids system evasion techniques. *Master. Masarykova Univerzita*, 2010.

[6] Intrusion prevention system (ips). https://www.geeksforgeeks.org/intrusion-prevention-system-ips/, Mar 2023.

[7] What is an ips? https://www.fortinet.com/resources/cyberglossary/what-is-an-ips.

[8] Paul Kirvan. What is a network intrusion protection system (nips)?: Definition from techtarget. https://www.techtarget.com/whatis/definition/network-intrusion-protection-system-NIPS, Jul 2023.

[9] TechTarget Contributor. What is wips (wireless intrusion prevention system)?: Definition from techtarget. https://www.techtarget.com/searchsecurity/feature/Introduction-to-wireless-intrusion-prevention-systems-in-the-enterprise, Mar 2015.

[10] What is network behavior analysis? definition, importance, and best practices. https://www.spiceworks.com/tech/networking/articles/network-behavior-analysis/, Feb 2022.

[11] Stephen J. Bigelow. What is host intrusion prevention system (hips)?: Definition from techtarget.

https://www.techtarget.com/searchenterprisedesktop/definition/host-intrusion-prevention-systems-HIPS, Jul 2023.

[12] Okta. Arp poisoning. https://www.okta.com/identity-101/arp-poisoning/.

[13] Buffer overflow. https://www.imperva.com/learn/application-security/buffer-overflow/, Dec 2019.

[14] What is a ddos attack? https://www.cloudflare.com/learning/ddos/what-is-a-ddos-attack/.

[15] Imperva. Ip fragmentation attack. https://www.imperva.com/learn/ddos/ip-fragmentation-attack-teardrop/.

[16] Andrew Harmon. Os fingerprinting, Jun 2020.

[17] What is port scanning? https://www.fortinet.com/resources/cyberglossary/what-is-port-scan.

[18] Server message block: Smb relay attack (the attack that always works). https://cqureacademy.com/blog/penetration-testing/smb-relay-attack.

[19] Smurf ddos attack. https://www.cloudflare.com/en-gb/learning/ddos/smurf-ddos-attack/.

[20] Syn flood attack. https://www.cloudflare.com/en-gb/learning/ddos/syn-flood-ddos-attack/.

[21] Daniel P. Newman and Kristina Maria Manalo. *Attack Types*, volume (Exam 642-531), page 528. Que.

[22] Nmap. https://www.simplilearn.com/tutorials/cyber-security-tutorial/what-is-nmap.

[23] Firewall/ids evasion and spoofing — nmap network scanning. https://nmap.org/book/man-bypass-firewalls-ids.html.