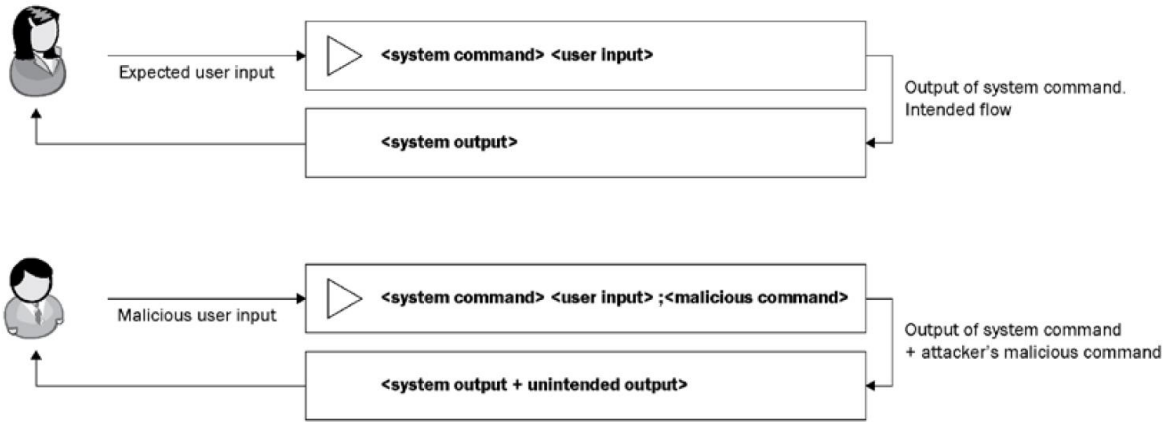
Chapter 12:Investigating External Threats

• Investigating web attacks

❖ The command injection vulnerability

Some web applications are designed to take input from users and then process it by invoking a shell to run a program to handle the input. An attacker may take advantage of this process and inject a command in their web request inputs to be executed on a vulnerable application. To do so, attackers usually use the ; character at the end of the normal input to be able to add their own injected command



❖ The SQL injection vulnerability

To gain initial access to the system, an attacker may exploit a **SQL injection** vulnerability in an application database. The simple way is for the attacker to enter `' or 1=1; --` in the `username` parameter, and the resulting SQL query will look like this:

```
SELECT username,password FROM users WHERE username='' or 1=1; --' and
password='';
```

By entering the preceding value in the username parameter, the attacker takes advantage of the SQL database, which needs a `true` condition to return a record. `1=1` is `true`, so the database thinks the username is `' '` or `true`. `;` is used to end the SQL statement, and `--` is used to comment the rest of the line. The preceding SQL statement retrieves all users from the database, or may get logged into the database with the administrator account if it is the first record in the database table.

❖ Path traversal vulnerability

The path traversal vulnerability (also known as directory traversal vulnerability) is a vulnerability that allows external attackers to access files and directories on a server. This might include web application code and data, configurations, and sensitive files stored on the disk. To do so, the attacker manipulates
variables that reference files with dot-dot-slash (../) sequences and their variations – for example,
http://vulnerable_site.com/websitefiles?file=../../../etc/passwd – or
tries to access absolute file paths – for example, .. In both these examples, the attacker tries to reach the password file that contains a list of a account of Linux OSs
Note: the attacker tried to access files that exist on UNIXbased
operating systems, so if they wanted to exploit path traversal vulnerability in Windows operating systems, they would use dot-dot-backslash (..\) instead of dot-dot-slash (../).

To evade detection by security controls such as the WAF, attackers usually use encoded characters, as shown in the following table:

Encoded value	Represented value
%2e%2e%2f	../
%2e%2e%5c	..\
%2e%2e/	../
%2e%2e\	..\

Table 12.1 – Encoded characters used by attackers

❖ XSS vulnerability

XSS is a web security vulnerability that allows attackers to steal web application information (such as web cookies) from users surfing a vulnerable website

- ❖ Reflected XSS:
  Reflected XSS is the simplest form of XSS vulnerability exploitation. In this type of XSS exploitation, the attacker sends the victim a phishing email or tricks the victim into visiting a website (such as his bank website) that is vulnerable to XSS, by clicking on a link that includes an embedded JavaScript to steal, for example, the user's bank cookies. After clicking on the link, the victim's browser sends the script to the vulnerable web application as user input. Then, the vulnerable web application reflects the user input (malicious JavaScript code to steal the cookies) back to the victim's browser, and finally, the script runs on the victim's browser to steal all cookies for this vulnerable website, which are sent via HTTP, email, and so on to the attacker's server.

- Stored XSS:
  Stored XSS is another form of XSS vulnerability exploitation that arises when a web application allows content to be posted by third parties; hence, the attacker can just post and “store” the malicious content (malicious JavaScript code to steal the users’ cookies) directly on the vulnerable application itself. The malicious content might be submitted to the web application via HTTP requests – for example, comments on social media posts, or user nicknames in a chat room. After clicking on the malicious content posted on the website, the victim’s browser sends the script to the vulnerable web application as user input. Then, the vulnerable web application reflects the user input (malicious JavaScript code to steal the cookies) back to the victim’s browser, and finally, the script runs on the victim’s browser to steal all cookies for this vulnerable website, which are sent via HTTP, email, and so on to the attacker’s server.

❖ Investigating WAF logs

The Web Application Firewall (WAF) is a security solution that comes in the form of an appliance,software, or cloud, serving to protect standard and custom web applications from web application attacks, such as SQL injection and XSS attacks. A WAF solution should have web application security knowledge and a deep understanding of the applications that protect. The WAF is also known as a reverse proxy because it acts as a proxy between an organization’s web applications and their visitors to filter out malicious traffic and protect the web application

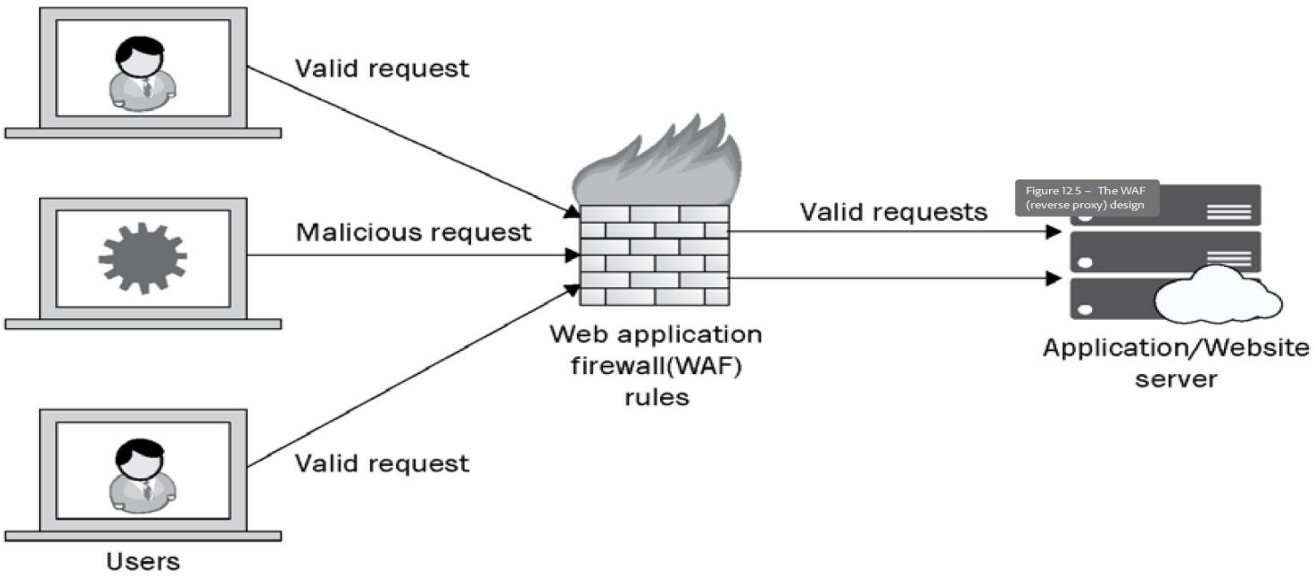


Figure 12.5 – The WAF (reverse proxy) design

traffic or traffic looks legit. The WAF nearly generates the same log attributes provided by the web proxy, such as the **source IP** and **port**, the **destination IP** and **port**, the **accessed URL**, **sent bytes**, **received bytes**, the **HTTP method**, the **user agent**, the **HTTP response**, and the **device action**. Also, the WAF provides the following features:

- Unique log attributes, such as the **violation type**, that indicate the attack type, such as SQL injection, DDoS, or XSS
- A **matched signature** that indicates the matched signature in the web traffic – for example, `1=1` strings that indicate SQL injection exploitations
- **Source geolocation** that indicates the traffic source geolocation

• Investigating suspicious external access to remote services

❖ Investigating unauthorized VPN and RDP access

To detect and investigate such attacks, you should monitor and investigate the following:

- Investigate the allowed RDP traffic to the organization's published servers by analyzing the firewall logs.
- Investigate the multiple login failure attempts against the organization's VPN or RDP accounts by analyzing either the application or OS logs.
- Investigate the suspicious successful authentications to your environment's VPN or RDP accounts from unexpected geolocations, or from two different geolocations in a short time period. Use an accurate IP geolocation database such as `https://ipgeolocation.io/` to make sure that the geolocations are different.
- Investigate large amounts of data sent from internal IPs over a VPN or RDP channels to an external IP by analyzing the firewall logs.

❖ Investigating compromised mailboxes

- Investigate multiple login failure attempts against an organization's mailboxes.

- Investigate suspicious access to your environment's mailboxes from unexpected geolocations or from two different geolocations in a short time period. Use an accurate IP geolocation database to make sure that the geolocations are different.
- Investigate several emails sent from an internal mailbox to either a group of internal employees or another organization's email addresses.
- Investigate sent emails from internal email addresses that have suspicious subjects.
- Investigate large emails sent from an internal email address to a suspicious external email address.

❖ Investigating suspicious authentications to web services

- Investigate multiple login failure attempts against the organization's web services.
- Investigate suspicious access to your environment's web services from unexpected geolocations or from two different geolocations in a short time period. Use an accurate IP geolocation database to make sure that the geolocations are different, and verify with the customer whether their account was shared with anyone or uses a VPN application.
- Investigate user login and activities from different user agents.
- Investigate excessive browsing activities from authenticated users.

Chapter 13, Investigating Network Flows and Security Solutions Alerts

• Investigating network flows

Most SIEM solutions provide an integration capability to receive flows from different network devices. As an SOC analyst, you should take advantage of the network session information (NetFlow) generated from the network devices to detect and investigate the following:

- Suspicious communications from/to blacklisted IPs
- Suspicious communications over suspicious ports
- A high number of transferred bytes between two IPs
- Outbound communications during unusual times – for example, outside of working hours

- Investigating IPS/IDS alerts

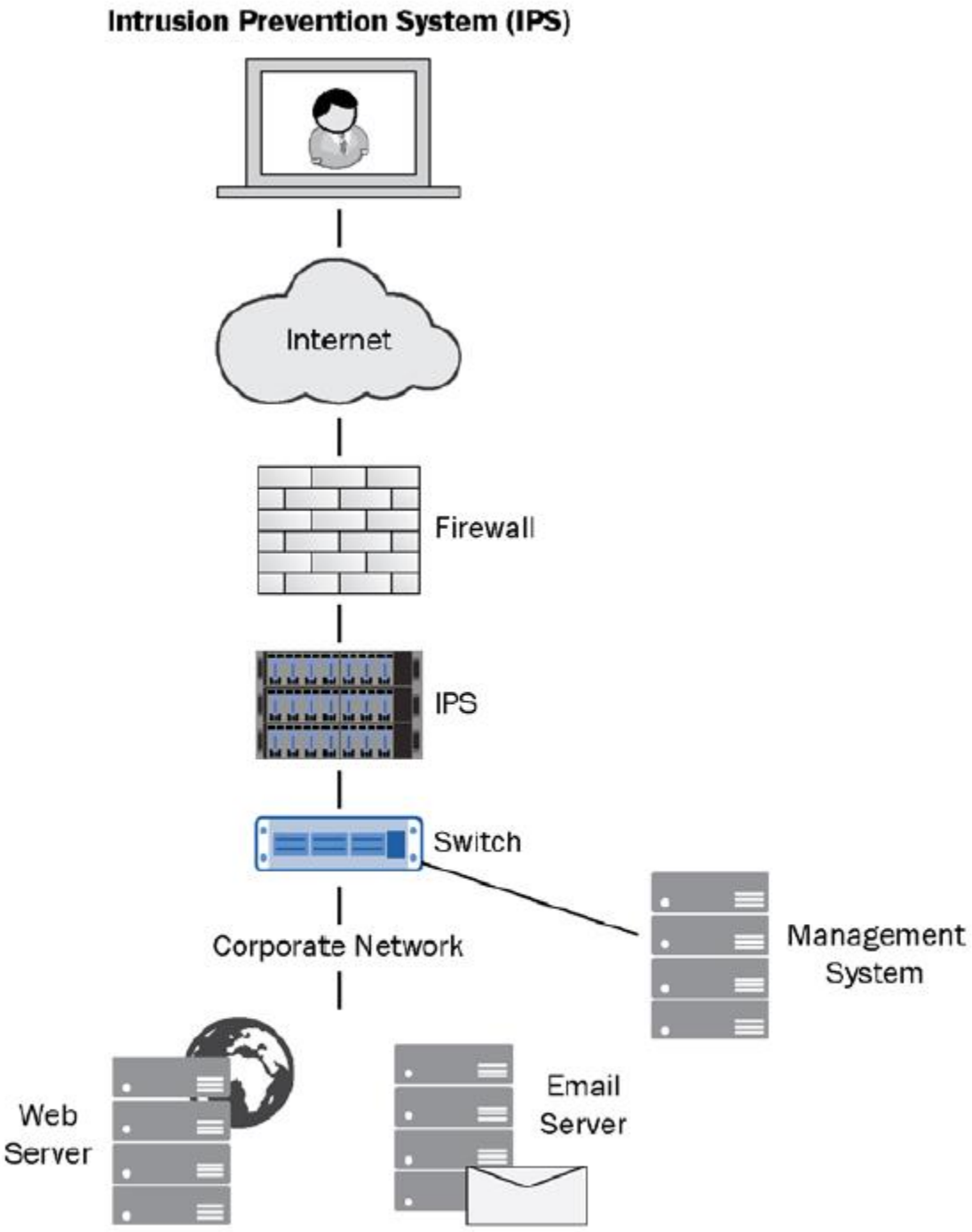
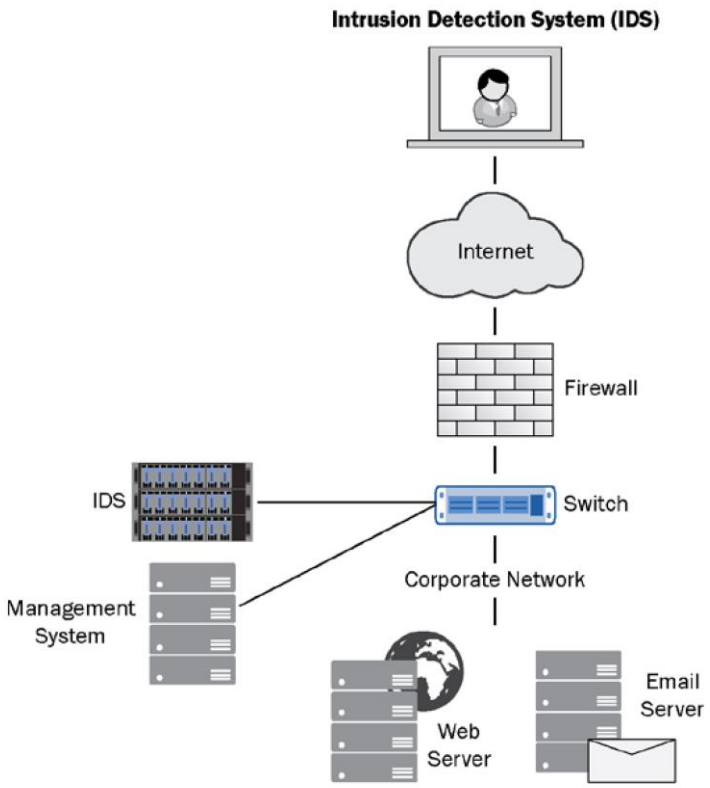


Figure 13.1 – An IPS layout



While most of the alerts received from the IPS/IDS should be investigated,you should pay more attention to

- Alerts with medium, high, and critical severity
- Several exploitation attempts from internal machine against internal machine/s
- IDS alerts about the exchange of malicious executable files
- Alerts of downloading malicious executables from external systems
- Several exploitation attempts from external IP against web-facing system(s)

• Investigating endpoint security solutions alerts

❖ Investigating AV alerts

The alerts received from the AV solutions contain at least the following details:
• An infected machine name
• An infected filename
• An infected file path
• An infected file hash
• A malware name
• A malware category

❖ Investigating EDR alerts

Endpoint Detection and Response (EDR) is an endpoint security solution that provides continuous monitoring and collection of endpoint data with automated rules to detect suspicious behaviors. EDR also provides live analysis and response capabilities, aiding threat hunters to discover undetected threats and Digital Forensics and Incident Response (DFIR) analysts to analyze infected systems and respond to cyber threats.

• Investigating network sandbox and AV alerts

The network AV solution is a crucial network security control that organizations implement to scan all files and URLs that are either transferred internally or sourced from external resources, such as emails and web servers.

The network sandbox solution is a network security solution implemented in an organization's network to render or execute and analyze the behavior of files and URLs, including those internally transferred and downloaded from external resources such as email and a web server in an isolated environment, before sending them to an end user.

Chapter 14, Threat Intelligence in a SOC Analyst's Day

In cyber security, threat intelligence represents sharing contextual threat information on attacks and threat actors across defense environments.

The information shared in threat intelligence, which is also known as threat intelligence feeds, is divided into three levels:

1. Strategic
   The strategic threat intelligence level is information about the organization's threat landscape. This type of information usually does not contain technical information
2. Operational
   The operational threat intelligence level includes the Tactics, Techniques, and Procedures (TTPs) used by threat actors.

3. Tactical
   The tactical threat intelligence level is information about threat actors' Indicators of Compromise (IOCs).This type of data is useful to network defense teams such as SOCs for detecting uncovered threats.

The Sigma rules are threat detection rules designed to analyze system logs. Sigma was built to allow collaboration between the SOC teams as it allows them to share standardized detection rules regardless of the SIEM in place to detect the various threats by using the event logs.
To know more about sigma rule :https://www.picussecurity.com/resource/glossary/what-is-sigma-rule

the following table highlights the preferred platforms to use to investigate different types of threat artifacts:

Artifact type	VirusTotal	X-Force	AbuseIPDB	Google
Web domain	✓	✓		✓
Outbound IP	✓	✓		
File hash	✓	✓		
Inbound IP		✓	✓	
User agent				✓
Filename				✓
Email subject				✓

Chapter 15, Malware Sandboxing – Building a Malware Sandbox

Introducing the sandbox technology

In cybersecurity, sandbox technology is an isolated test environment that looks like end user operating systems to safely execute and analyze suspicious files and investigate their behavior. A sandbox is also useful if you are dealing with zero-day malware.

Sandbox types

- Cloud sandboxes
- On-premises sandboxes

Static analysis

Static analysis tools are the tools that will be used to collect and analyze information about the suspected file without execution. The static analysis tools that we will install on our private sandbox
are as follows:

- YARA: YARA is a tool aimed at (but not limited to) helping malware researchers identify and classify malware samples. We will use the YARA tool to scan the suspected files for certain malware characters to identify the malware category and family if detected

- Exeinfo: A great GUI tool for analyzing the Portable Executable (PE) file header information. We will use this tool to verify whether we are dealing with a packed file, and if so, it helps to identify the packer and how to unpack it.
- Compute hash: A suggested tool for calculating the file hash
- PEstudio: This is the most useful tool in the static analysis phase, as it has been made specifically for static malware analysis.

Dynamic analysis tools

Dynamic analysis tools are the tools that are used to analyze suspicious file behavior after the execution as it enables you as a security analyst to watch the malware in action

- FakeNet: This tool simulates a fake network and internet so that malware interacting with a remote host continues to run, allowing the analyst to observe the malware's network activity
- Wireshark: A free and open source network packet analysis tool that will be used to analyze the FakeNet output PCAP file.
- ProcMon: This is a tool that records real-time system activity such as process creation, registry key editing and adding, file touching, network connections, and so on with a great filtering capability
- ProcDot: This tool visualizes the ProcMon output.
- Autoruns: This is a very useful free tool from Microsoft that checks for suspicious entries and the code signing certificate on persistence locations such as registry paths and scheduled tasks.