NEXT GENERATION FIREWALL

 @thepiratesshield

 @piratesshield

 @piratesshield

By
Anandraj Amaran
Piratesshield

Firewall Generations

Packet Filtering Firewall

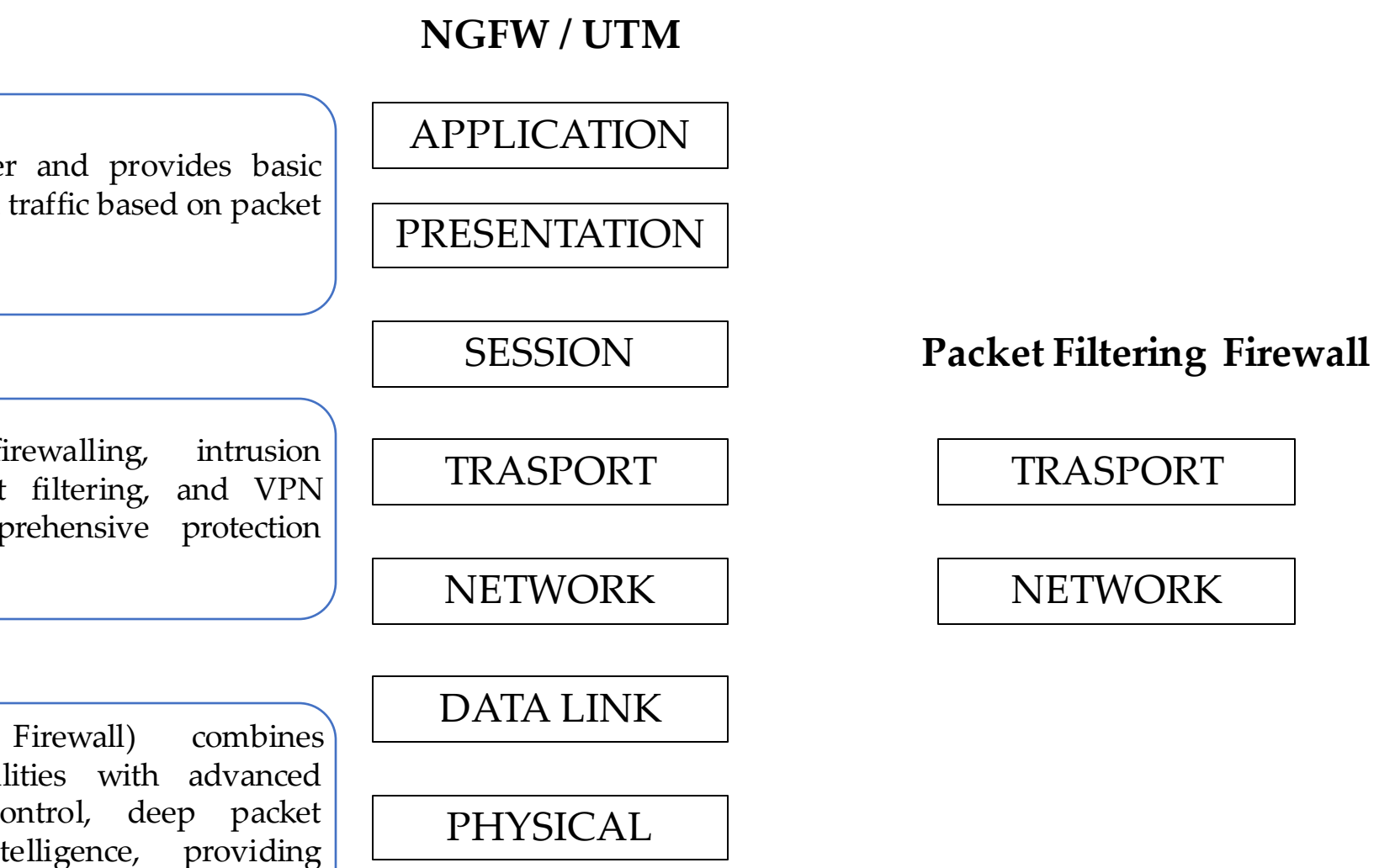
- operates at the network layer and provides basic security by controlling network traffic based on packet header information.

UTM

- offering functions like firewalling, intrusion prevention, antivirus, content filtering, and VPN connectivity, providing comprehensive protection against various threats

NextGen Firewall

- NGFW (Next-Generation Firewall) combines traditional firewall functionalities with advanced features like application control, deep packet inspection, and threat intelligence, providing enhanced security and visibility into network traffic.



Flow Based and Proxy Based Inspection

- Flow-based and proxy-based inspection are two different methods used by firewalls to inspect traffic.
- Flow-based inspection takes a snapshot of content packets and uses pattern matching to identify security threats in the content.
- Proxy-based inspection reconstructs content that passes through the firewall and inspects the content for security threats.

Feature	Flow-based inspection	Proxy-based inspection
Efficiency	More efficient	Less efficient
Performance	Faster	Slower
Impact on network latency	Less impact	More impact
Capabilities	Can inspect some types of traffic	Can inspect all types of traffic
Security	May miss some security threats	Less likely to miss security threats

Which one to use?

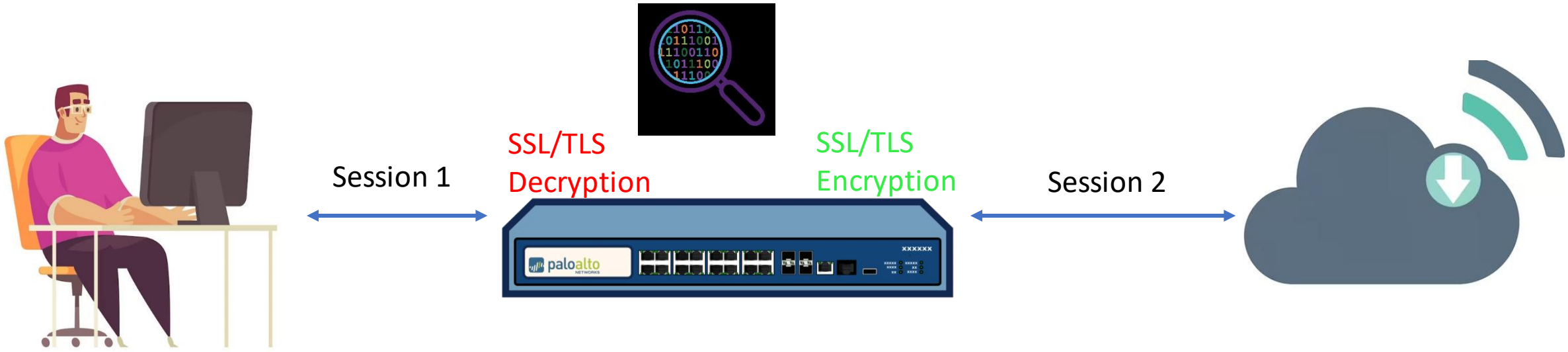
The best inspection mode to use depends on your specific needs and requirements. If you need the most comprehensive security and are willing to sacrifice some performance, then proxy-based inspection is the best option. If you need to maximize performance and are willing to accept some limitations on security, then flow-based inspection is the best option.

Features of NXFW

- Deep Packet Inspection (DPI)
- Intrusion Prevention System (IPS)
- Application Awareness and Control
- Advanced Threat Protection
- URL Filtering
- DNS Security
- WAF
- User Identity Awareness
- VPN Integration
- Centralized Management
- Logging and Reporting

Deep packet inspection

DPI DPI examines a larger range of metadata and data connected with each packet the device interfaces with. In this DPI meaning, the inspection process includes examining both the header and the data the packet is carrying.



*Custom CA cert
Installed in User PC

#Can not apply DPI on Bank and government website because of the Certificate pinning
#DPI is difficult , When Web-App uses the high level custom encryption

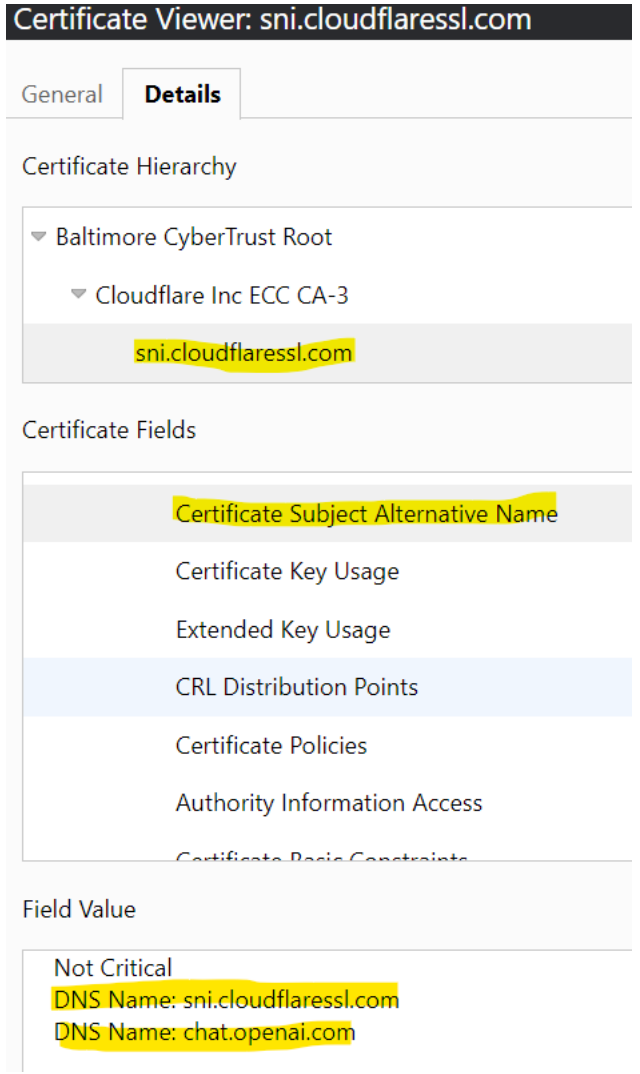
URL Filtering / Web Filtering

Web filtering in a firewall refers to the process of monitoring and controlling web traffic based on predefined rules or policies to block or allow access to specific websites or web content.

Capable to inspect the content inside the Encapsulated Tunnels QUIC and GRE

Method used :

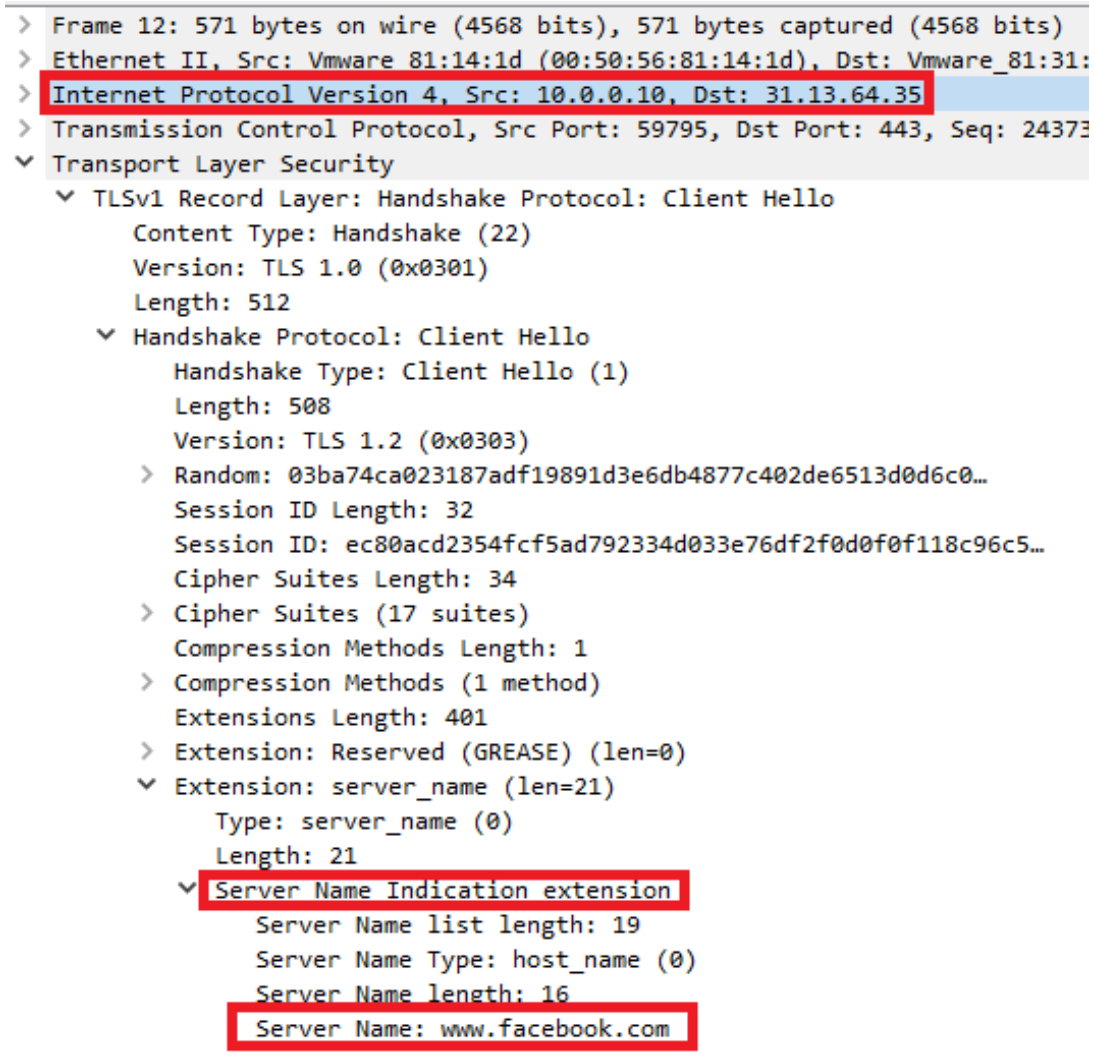
- Filter Over the HTTP headers
- Analyzing the SNI / SAN



Analyzing the SNI / SAN

The SNI information is sent in clear text during the initial handshake of the TLS protocol. It is part of the unencrypted *ClientHello* message that the client sends to the server. This allows the server to present the appropriate certificate for the requested hostname.

*#**DPI** required in URL Filtering to do content/Malware/keyword analysis*



Process	Description
SNI Extraction	Firewall inspects initial handshake packets to extract the SNI sent by the client, providing the intended hostname or domain name for the secure connection.
Certificate Inspection	Firewall examines the server certificate presented during the handshake, specifically looking for the SAN field that contains additional domain names or IP addresses associated with the certificate.
Comparison	Firewall compares the extracted SNI with the values listed in the SAN of the certificate. If there is a match, the firewall associates the requested URL with the certificate.
Policy Enforcement	Based on the comparison results, firewall applies appropriate policies. If the SNI and requested URL align or there is a match in the SAN, the connection is allowed. Otherwise, the firewall may block the connection or apply additional scrutiny.

By examining both the SNI and the SAN, the firewall can accurately determine the relationship between the requested URL and the certificate presented by the server. This approach helps ensure secure and authorized connections while maintaining flexibility to handle scenarios where the SNI and SAN may differ.

Application Control

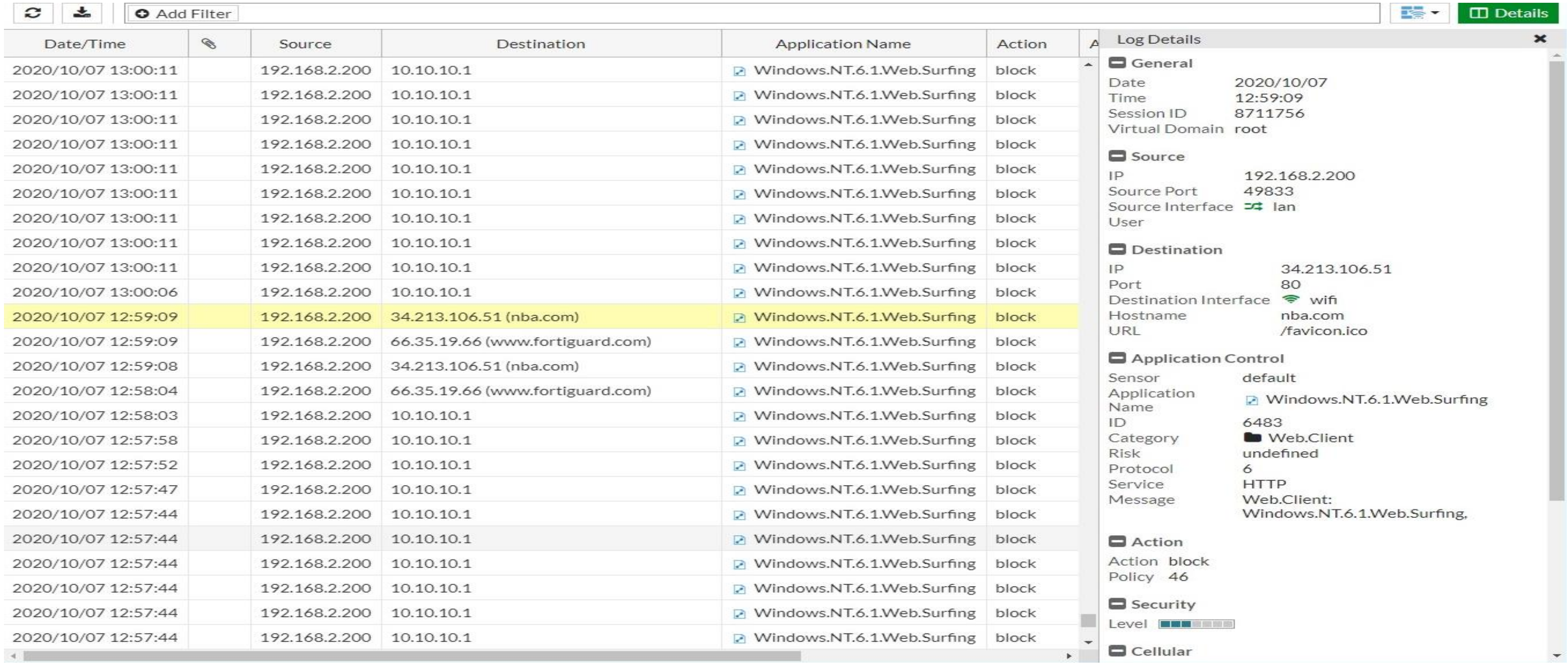
Application control in FortiGate provides granular control over network traffic based on specific applications, allowing administrators to define policies and enforce rules for application-level management and security. It enables identification, prioritization, and blocking of applications, along with bandwidth management and threat prevention. With comprehensive reporting and integration with security services, it enhances network visibility and protection against application-level risks.

- Control the Dynamic IP and URLs usage of Application
- Dynamic content of Application
- URL Obfuscation Techniques
- Behavior/category based Control
- Control Functions of Application
- Bandwidth Optimization
- Able to create custom Application signatures

Custom Application Signature Example

F-SBID(--attack_id 6483; --name "**Windows.NT.6.1**.Web.Surfing"; --default_action drop_session; --service HTTP; --protocol tcp; --app_cat 25; --flow **from_client**; --pattern !"FCT"; --pattern "Windows NT 6.1"; --no_case; --context header; --weight 40;)

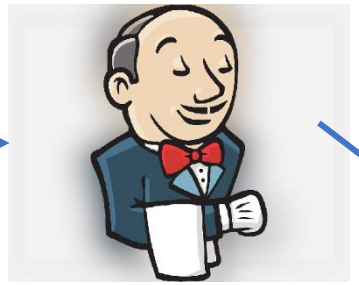
This signature scans HTTP and HTTPS traffic that matches the pattern Windows NT 6.1 in its header. For blocking older versions of Windows, such as Windows XP, you would use the pattern Windows NT 5.1. An attack ID is automatically generated when the signature is created.



Date/Time		Source	Destination	Application Name	Action
2020/10/07 13:00:11		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block
2020/10/07 13:00:11		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block
2020/10/07 13:00:11		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block
2020/10/07 13:00:11		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block
2020/10/07 13:00:11		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block
2020/10/07 13:00:11		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block
2020/10/07 13:00:11		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block
2020/10/07 13:00:11		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block
2020/10/07 13:00:11		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block
2020/10/07 13:00:06		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block
2020/10/07 12:59:09		192.168.2.200	34.213.106.51 (nba.com)	Windows.NT.6.1.Web.Surfing	block
2020/10/07 12:59:09		192.168.2.200	66.35.19.66 (www.fortiguard.com)	Windows.NT.6.1.Web.Surfing	block
2020/10/07 12:59:08		192.168.2.200	34.213.106.51 (nba.com)	Windows.NT.6.1.Web.Surfing	block
2020/10/07 12:58:04		192.168.2.200	66.35.19.66 (www.fortiguard.com)	Windows.NT.6.1.Web.Surfing	block
2020/10/07 12:58:03		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block
2020/10/07 12:57:58		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block
2020/10/07 12:57:52		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block
2020/10/07 12:57:47		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block
2020/10/07 12:57:44		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block
2020/10/07 12:57:44		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block
2020/10/07 12:57:44		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block
2020/10/07 12:57:44		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block
2020/10/07 12:57:44		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block
2020/10/07 12:57:44		192.168.2.200	10.10.10.1	Windows.NT.6.1.Web.Surfing	block

Automations

SOC TEAM



Jenkin

API

WebHook

SHODAN

Threat Intelligence

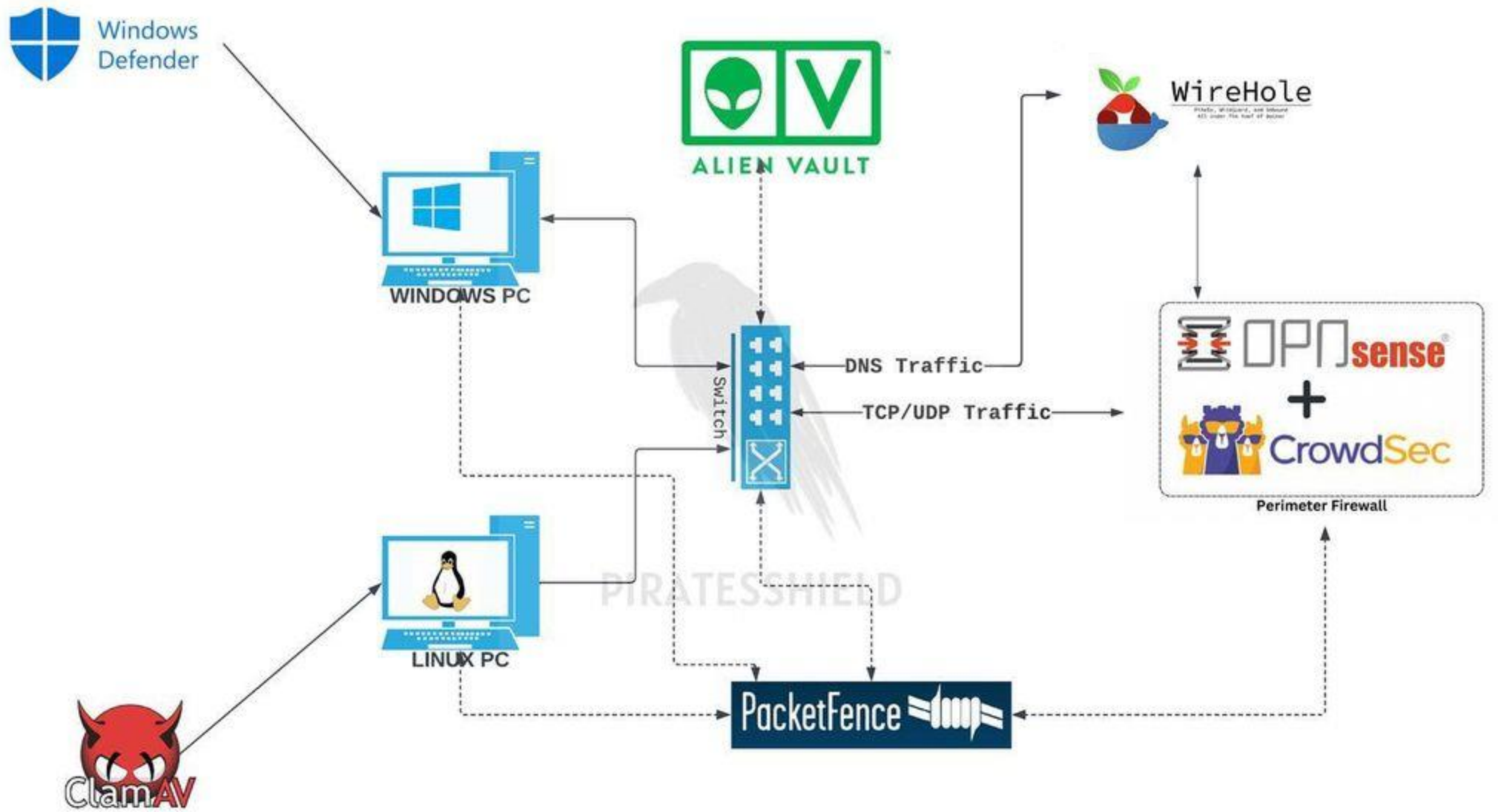
API & proprietary protocol

WebHook

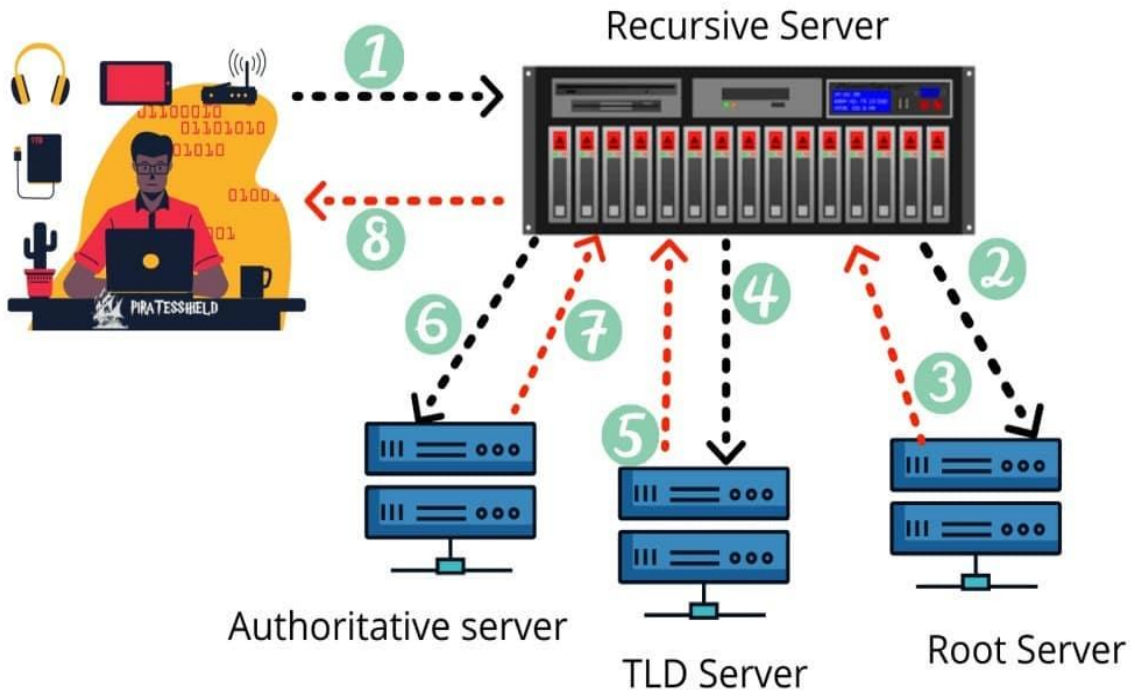
NAC

Network Access Control

Secure Infrastructure with FOSS




DNS
SECURITY



User's Input:
The user enters a URL, such as "www.example.com," into the browser's address bar.

Local Host File Lookup:
The operating system (OS) checks the local host file.
Wnindows: C:\Windows\System32\drivers\etc\host(file)
Linux : /etc/resolv.conf

Recursive DNS Query:
If the IP address is not found in the local host file, the OS sends a recursive DNS query to the configured DNS resolver . ***E.g. 8.8.8.8 , 1.1.1.1***

Resolver Interaction: (8.8.8.8 , 1.1.1.1)

- The resolver initiates the DNS resolution process by sending iterative queries to various DNS servers. It starts by contacting the root DNS servers to obtain information about the top-level domain (TLD) server responsible for the requested domain.
- Resolver sends a query to the authoritative name server identified in the previous step.
- The authoritative name server responds to the resolver with the IP address associated with the requested domain. The resolver caches this response to expedite future queries for the same domain.
- The resolver returns the IP address to the user's browser, which can now initiate a connection to the desired web server using the obtained IP address.

TYPES OF DNS

DNS over HTTPS (DoH): (TCP/UDP Port 443, 8443)
Encrypts DNS queries and responses using the HTTPS protocol, providing enhanced privacy and preventing eavesdropping or tampering with DNS traffic.

DNS over TLS (DoT): (TCP PORT 853)
DNS over TLS establishes a secure connection between the client and DNS resolver using the Transport Layer Security (TLS) protocol, ensuring encrypted communication and protecting against interception or manipulation.

DNSCrypt: (TCP Port 443)
DNSCrypt is a protocol that encrypts DNS traffic, adding cryptographic signatures for authentication and preventing DNS hijacking or unauthorized modifications.

Split DNS:
Split DNS, or split-horizon DNS, allows for different DNS resolutions based on network location or context, enabling customized DNS policies and filtering for internal and external networks.

DNS over Blockchain (DoB)
Uses combines the principles of DNS (Domain Name System) with blockchain technology. It proposes using a blockchain network to decentralize and secure the DNS infrastructure.

Types of Records

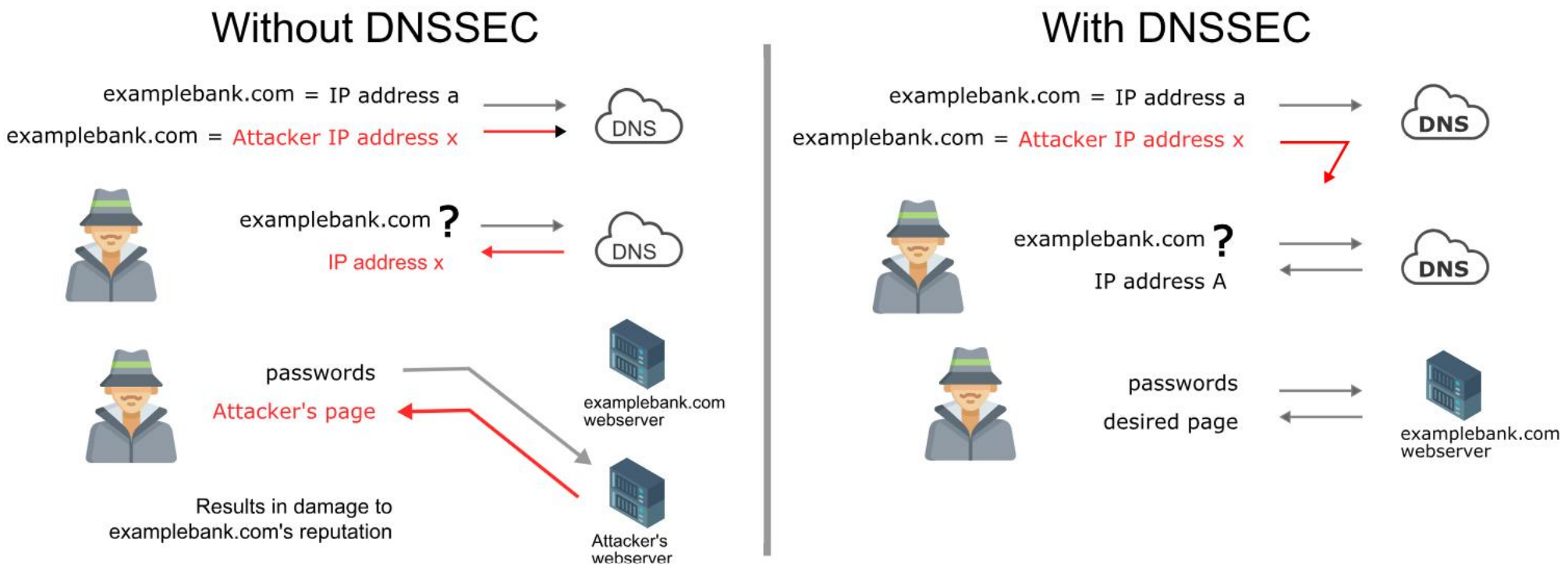
- **A Record:** Maps a domain name to an IPv4 address.
- **AAAA Record:** Maps a domain name to an IPv6 address.
- **CNAME Record:** Creates an alias for a domain or subdomain.
- **MX Record:** Specifies the mail server responsible for domain's email delivery.
- **TXT Record:** Allows arbitrary text to be added to a domain's DNS records for various purposes.
- **PTR Record:** Maps an IP address to a domain name in reverse DNS lookup.

Use of TXT Record :

SPF (Sender Policy Framework),DKIM (DomainKeys Identified Mail),DMARC (Domain-based Message Authentication, Reporting, and Conformance),Domain Ownership Verification

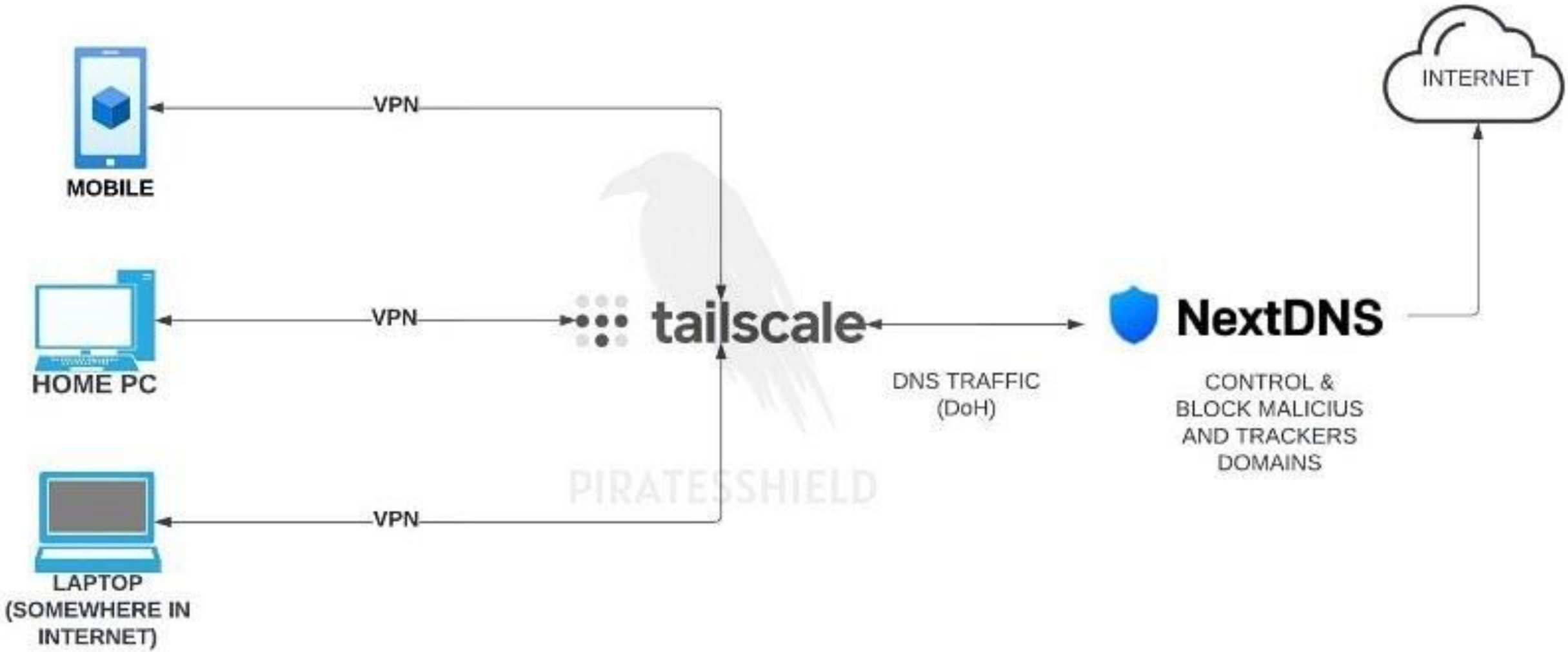
Use of **PTR record :**
Some email servers use reverse DNS lookup and PTR records to validate the sending server's identity. If a PTR record is missing, it may negatively impact email deliverability, as some receiving servers might consider the lack of a PTR record as suspicious or potentially indicate spam.

DNSSEC



DNSSEC (Domain Name System Security Extensions) is a security feature that adds cryptographic protection to the DNS. It uses digital signatures to verify the authenticity and integrity of DNS data, preventing DNS-related attacks. DNSSEC ensures that DNS responses are trustworthy and prevents DNS spoofing, providing a more secure and reliable DNS resolution process.

OPENSOURCE (FOSS) DESIGN FOR PROTECTING PERSONAL/FAMILY PRIVACY



THANK YOU !