

# Internet Security

Difesa della Rete

---

CORSO DI LAUREA TRIENNALE IN INFORMATICA (L-31)  
UNIVERSITÀ DEGLI STUDI DI CATANIA

DOTT. SERGIO ESPOSITO



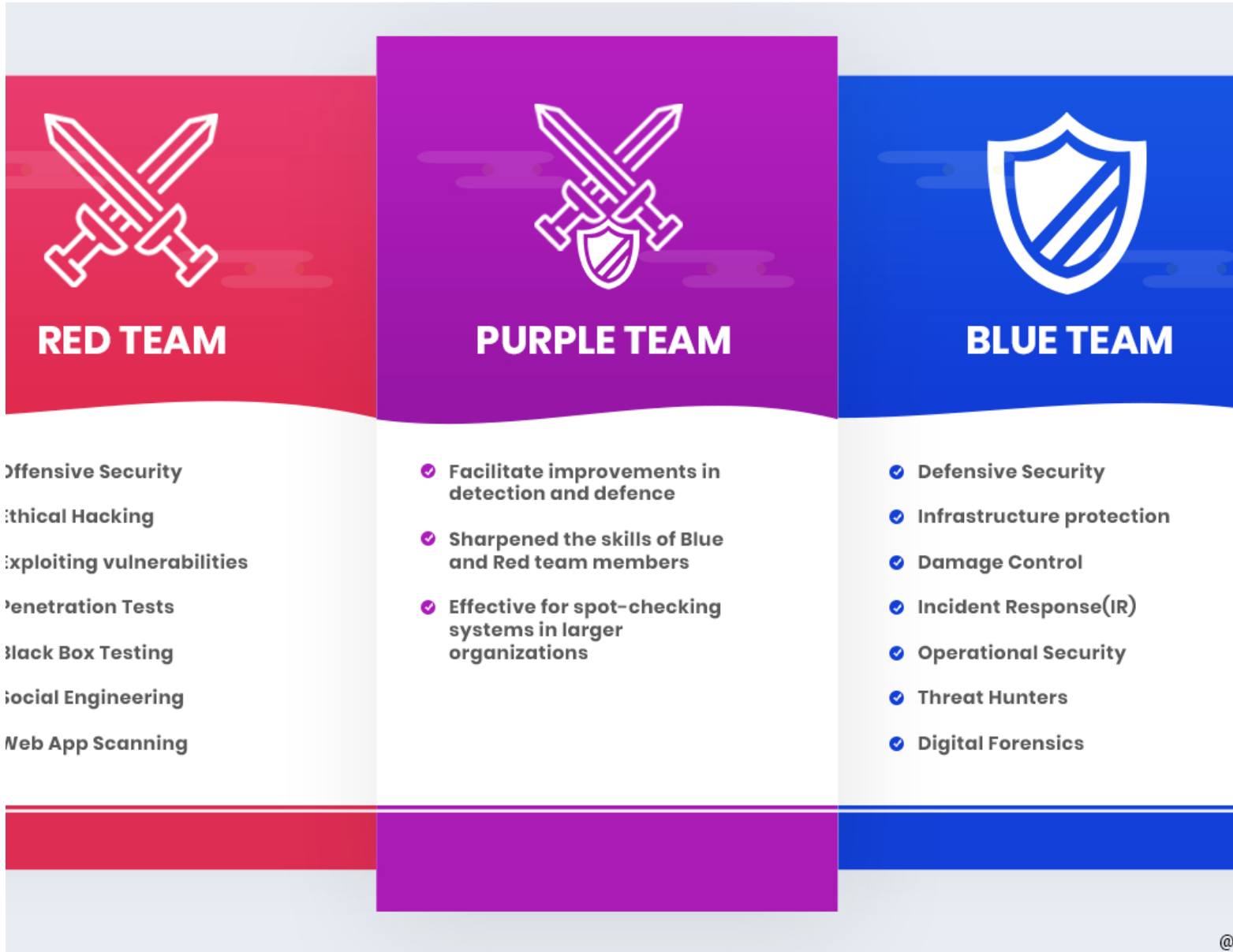
# Red vs Blue?

I loro compiti vanno ben oltre quelli riportati nell'infografica

Partita sempre in corso, scenario in continua evoluzione

Fonte immagine: @proxyblue,  
<https://hackernoon.com/introducing-the-infosec-colour-wheel-blending-developers-with-red-and-blue-security-teams-6437c1a07700>





## Red vs Purple vs Blue?

Potrebbe sembrare che ognuno giochi per sé, ma non è così!

Non è un gioco PvP, bensì PvE (Players vs Environment)

L'obiettivo comune è rafforzare il perimetro contro minacce reali.

Fonte immagine: @proxyblue,  
<https://hackernoon.com/introducing-the-infosec-colour-wheel-blending-developers-with-red-and-blue-security-teams-6437c1a07700>

# Intrusione

---

Dalle scorse slide di Internet Security:

"Ottenere illecitamente privilegi superiori a quelli posseduti lecitamente."

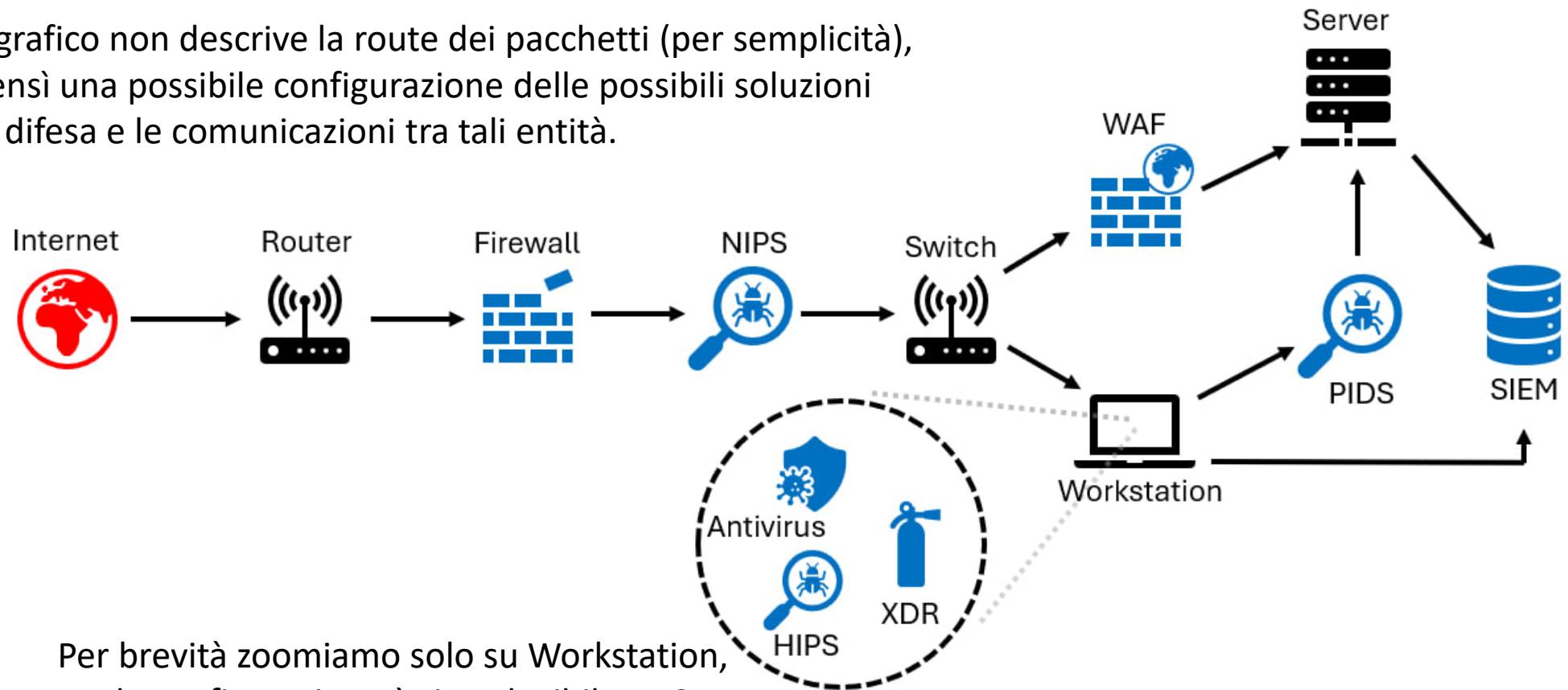
Innumerevoli modi per effettuarla,

Innumerevoli modi per difendersi!

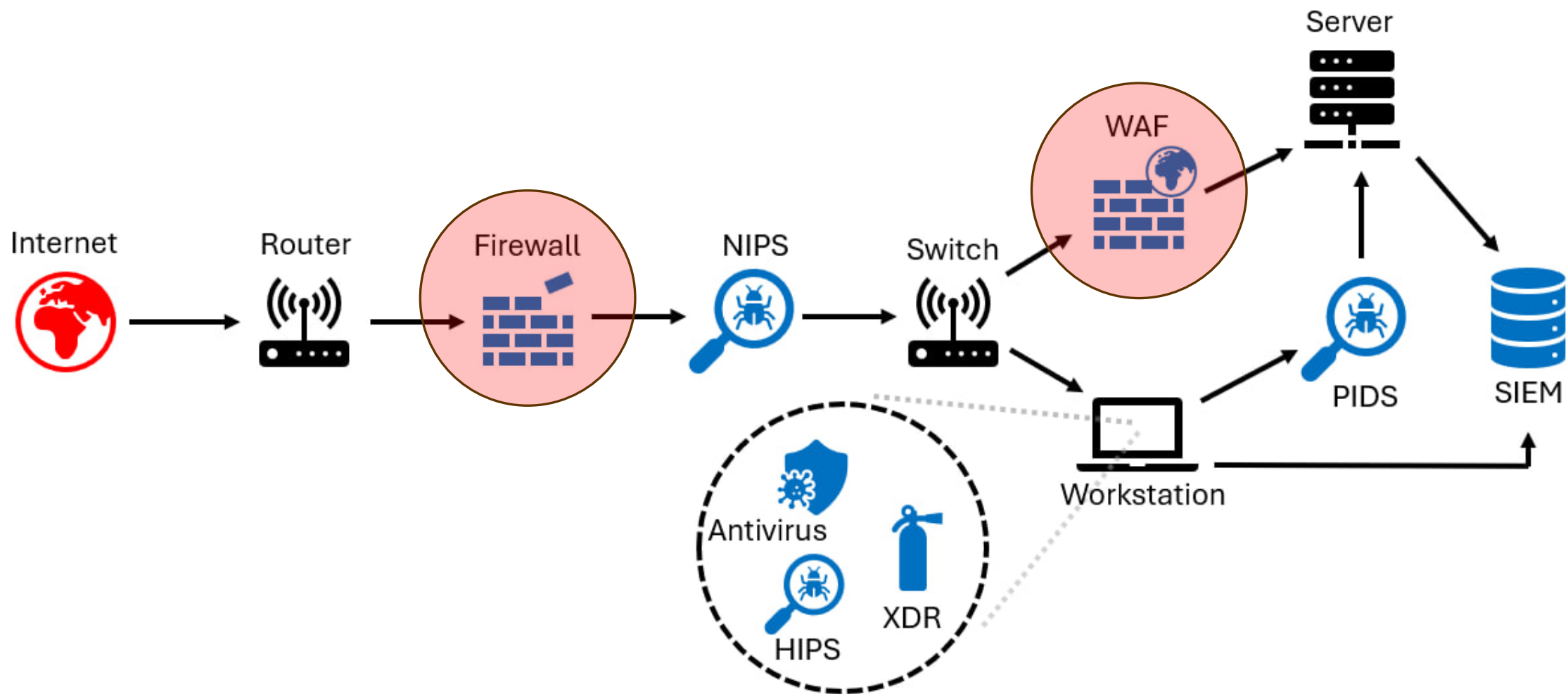
Impatto potenzialmente illimitato, a seconda dello scenario.

# Linee di Difesa Tipiche di una Rete

Il grafico non descrive la route dei pacchetti (per semplicità), bensì una possibile configurazione delle possibili soluzioni di difesa e le comunicazioni tra tali entità.



Per brevità zoomiamo solo su Workstation, ma la configurazione è riproducibile su Server.



# Firewall

---

Il firewall era originariamente concepito come un

"Dispositivo che controlla il flusso di traffico fra reti con diverse impostazioni di sicurezza."

Esempi:

- Fra una LAN e la rete Internet
- Fra due sottoreti interne
- Più in generale, fra una rete trusted e una untrusted

I firewall odierni si discostano un po' da questa idea, vedremo come

# Firewall

## Requisiti

---

- Tutto il traffico tra una rete e l'altra deve passare attraverso il firewall.
- Il passaggio di ogni tipo di traffico deve essere regolato da un'apposita regola di sicurezza all'interno del firewall.
  - Le regole possono essere specificate per gruppi. Ad esempio, "ammetti tutti i pacchetti TCP e UDP in entrata e uscita" copre tutti i tipi di pacchetti.
  - Forse non si tratta di una configurazione sicura?
  - Errori nella configurazione possono creare grossi problemi di sicurezza.
- Le best-practice per la gestione del firewall devono essere seguite.
  - Il firewall stesso potrebbe avere delle vulnerabilità.



# Vulnerabilità

---

"Violazione di una delle tre proprietà fondamentali della sicurezza (Confidenzialità, Integrità, Disponibilità) all'interno di un dato software o hardware."

Esistono diversi cataloghi delle vulnerabilità note, il più usato è il Common Vulnerabilities and Exposures (CVE), mantenuto dal MITRE ( <https://cve.mitre.org/> )

Una vulnerabilità non (ancora?) pubblica e non patchata viene chiamata 0-day.

# Debolezza (Weakness)

---

"Problema di sicurezza teorico, non collegato ad uno specifico software o hardware."

Il Common Weakness Enumeration (CWE) è anch'esso gestito dal MITRE, e da anni cataloga tutte le weakness. ( <https://cwe.mitre.org/> )

## Esempi:

CWE-78: Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')

CWE-89: Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')

CWE-121: Stack-Based Buffer Overflow

# Exploit

---

"Qualsiasi risorsa che sfrutta una vulnerabilità."

Se viene usato per sfruttare una vulnerabilità 0-day, viene chiamato exploit 0-day a sua volta

Può avere pressoché qualsiasi forma!

- Eseguitibile
- Stringa
- Immagine
- Audio
- Pagina web
- Archivio
- Codice
- Oppure altro!

# Esempi di Vulnerabilità su Firewall

---

## CVE-2024-3400

Arbitrary file creation vulnerability in the GlobalProtect feature of Palo Alto Networks PAN-OS software may enable an unauthenticated attacker to execute arbitrary code with root privileges on the firewall.

Dettagli: <https://nvd.nist.gov/vuln/detail/CVE-2024-3400>

Exploit: <https://github.com/Yuvvi01/CVE-2024-3400>

## CVE-2022-30525

A OS command injection vulnerability in the CGI program of Zyxel USG FLEX 100(W) could allow an attacker to modify specific files and then execute some OS commands on a vulnerable device.

Dettagli: <https://nvd.nist.gov/vuln/detail/CVE-2022-30525>

Exploit: [https://github.com/jbaines-r7/victorian\\_machinery](https://github.com/jbaines-r7/victorian_machinery)

# Firewall

## Funzionalità Principali

---

- Protezione dei servizi
  - L'amministratore potrebbe voler esporre alcune porte solo a chi è dentro una determinata rete
- Monitoraggio e filtraggio del traffico
  - L'amministratore potrebbe voler filtrare del traffico anche su porte aperte, analizzando il contenuto dei pacchetti
- Filtraggio dei dati
  - Ad esempio allegati email che vengano scaricati

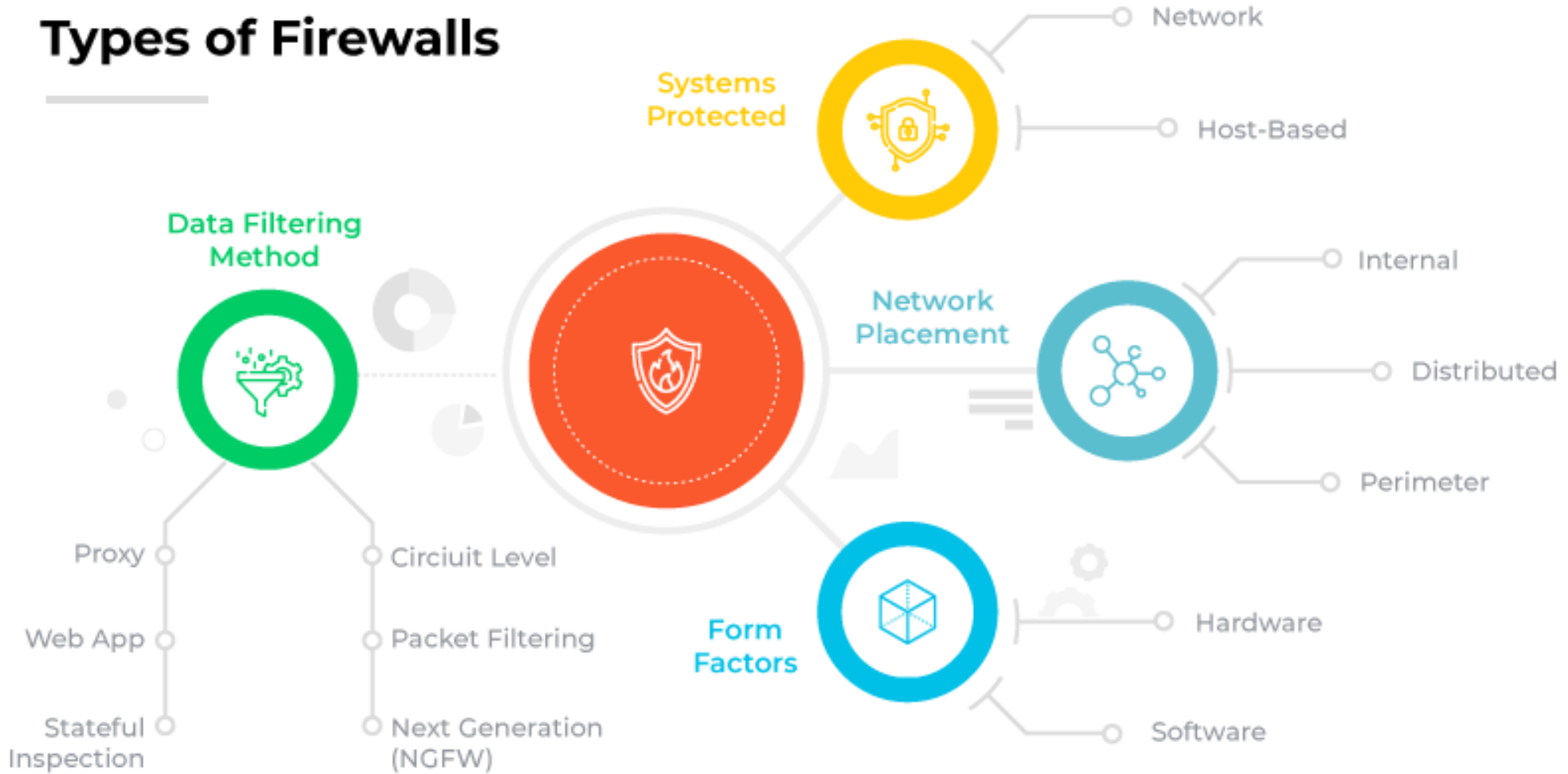
# Firewall

## Funzionalità Secondarie

---

- Creazione di VPN
  - Ad esempio, mediante IPSec in modalità tunnel
- Fornitura di indirizzi IP
  - Il firewall può fare sia da server che da client DHCP (Dynamic Host Configuration Protocol)
  - TunnelVision anyone? CVE-2024-3661
- Mappatura di indirizzi locali in indirizzi Internet
  - NAT (Network Address Translator)

# Types of Firewalls



Fonte immagini (questa e le successive sui firewall): <https://www.paloaltonetworks.com/cyberpedia/types-of-firewalls>

# Firewall

## Systems Protected

---

Prima differenza con la vecchia definizione: non tutti i firewall proteggono una rete!

**Network Firewall:** Si posiziona tra una rete e un'altra, e il suo ruolo è quello di monitorare la validità del traffico in entrata e in uscita secondo un determinato set di regole. Serve a proteggere una o più reti e a mantenerne l'integrità.

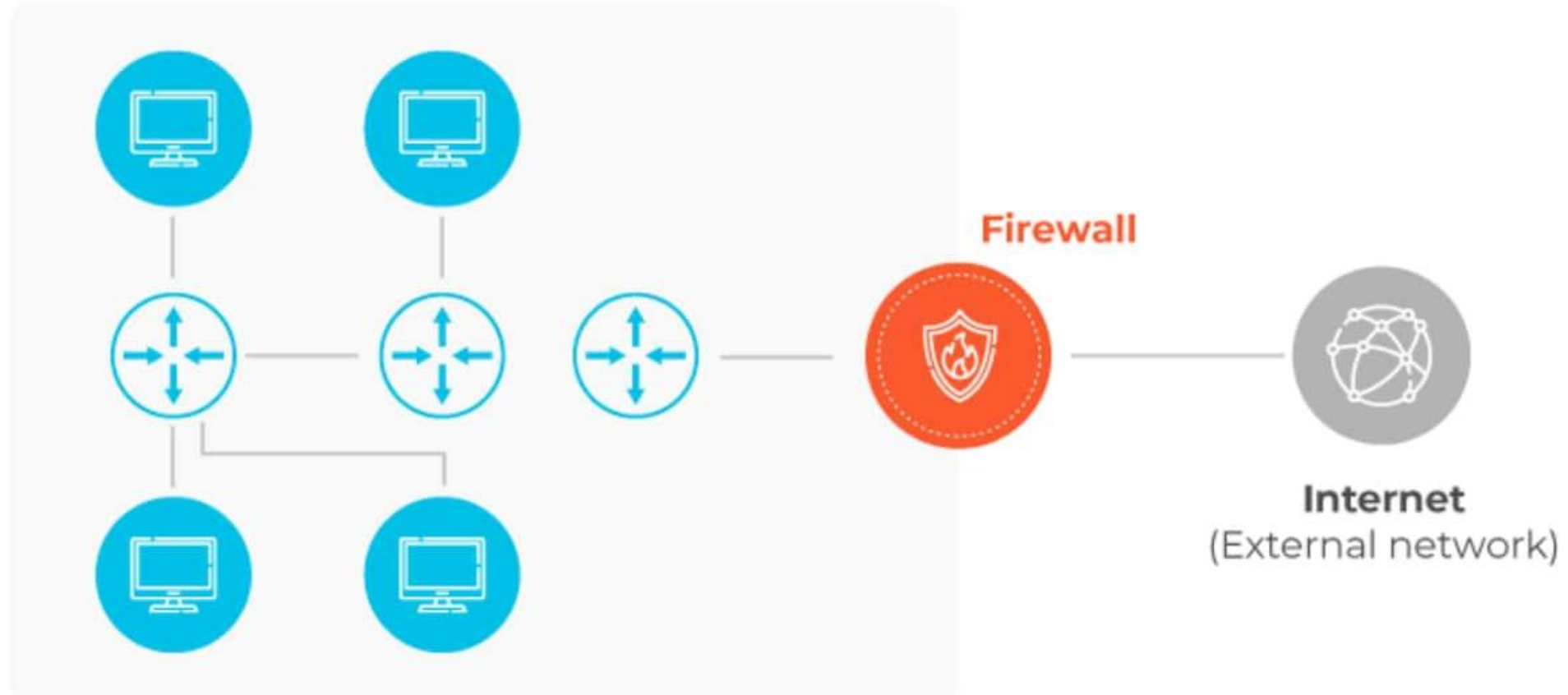
**Host-Based Firewall:** Esamina il traffico in entrata e in uscita del dispositivo su cui è installato, sulla base di un determinato set di regole. Anche se la rete viene violata, l'Host-Based Firewall può fornire una seconda linea di difesa per il dispositivo.



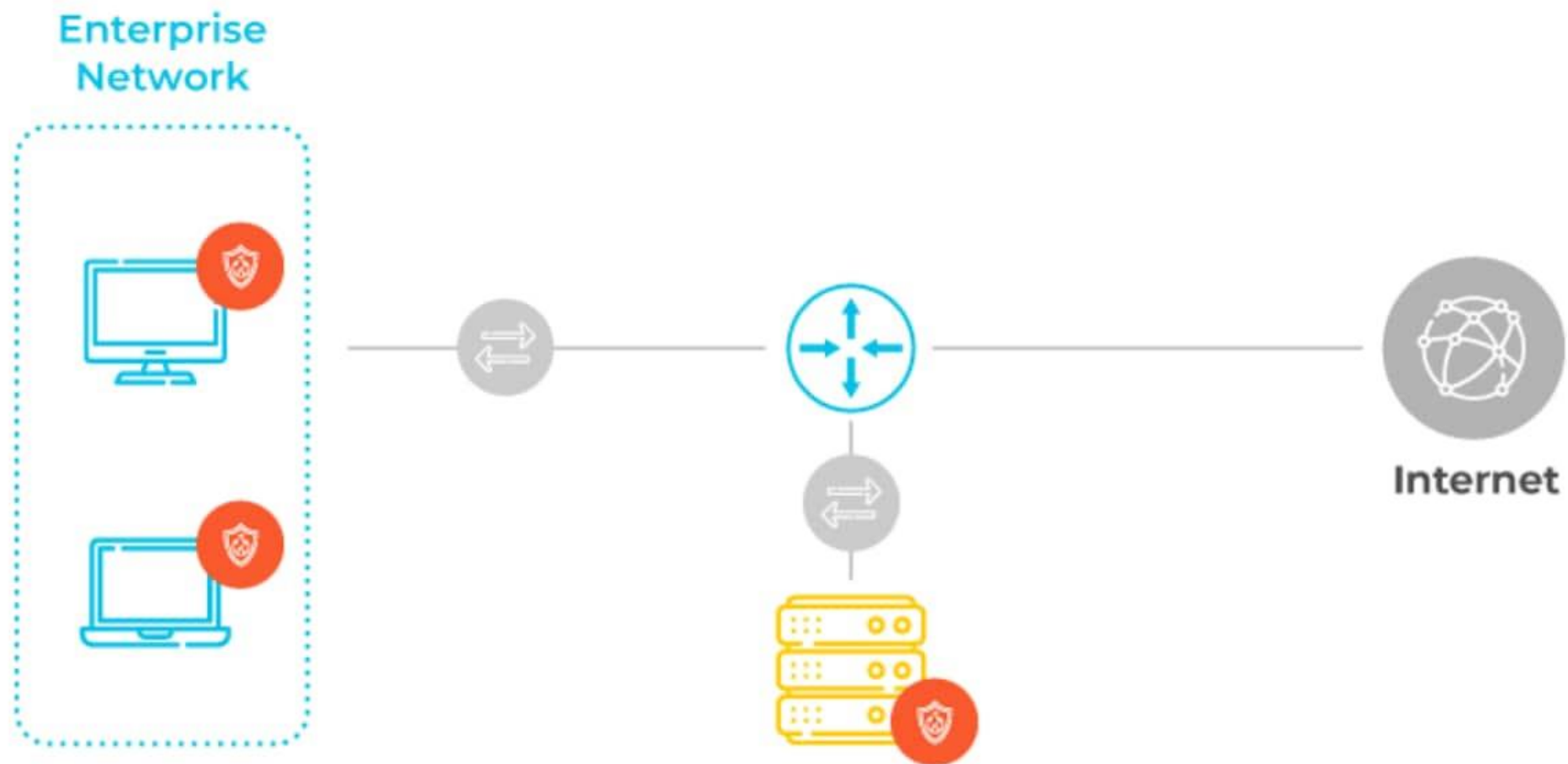


# Network Firewall

Internal Network



# Host-Based Firewall



# Firewall

## Network Placement

---



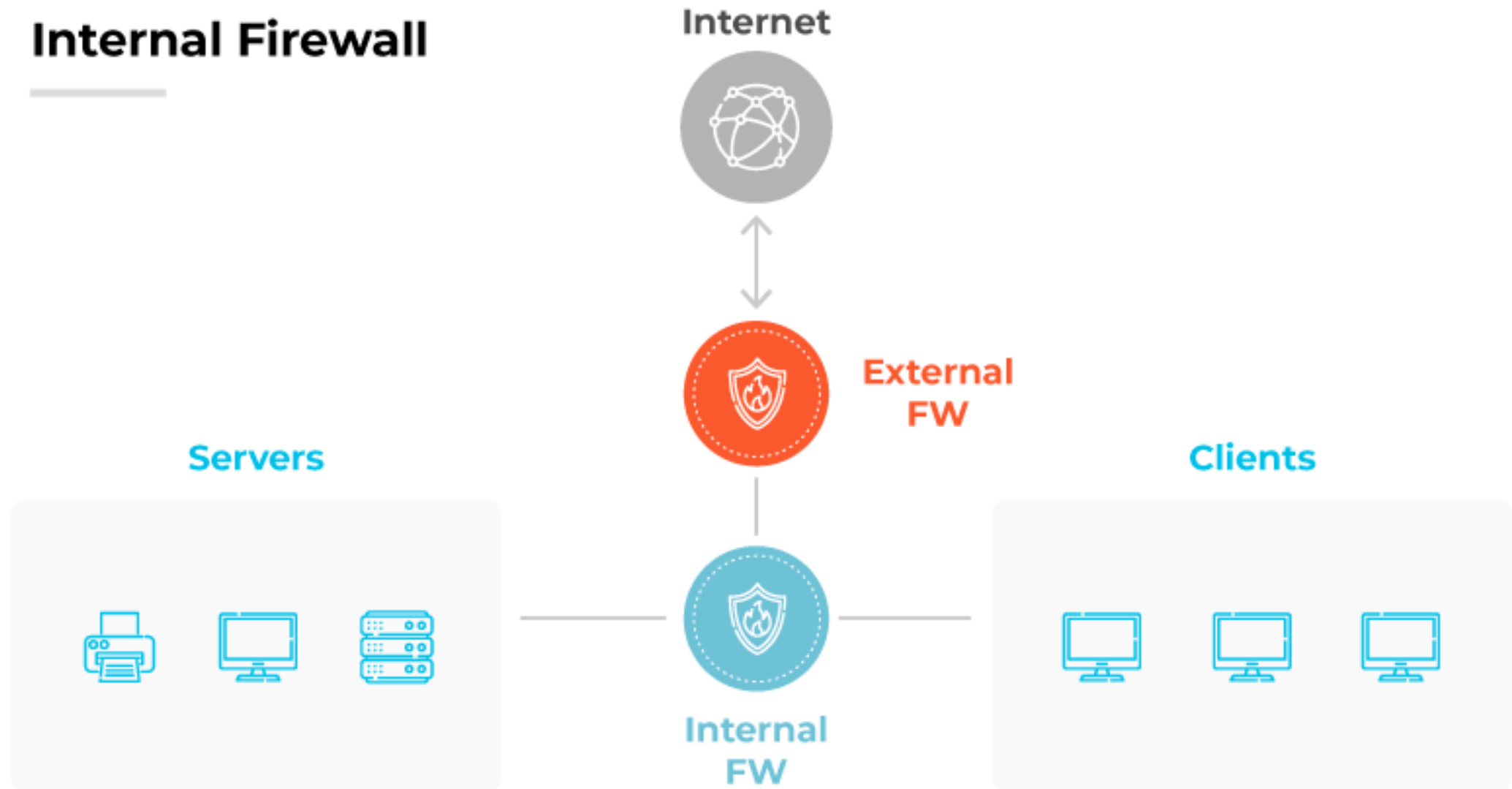
**Internal Firewall:** Analizza il traffico scambiato tra dispositivi della stessa rete, difendendo quindi dalle minacce che si trovano all'interno della stessa (e.g., insider threat, o script che girano su macchine già violate)

**Distributed Firewall:** A differenza degli altri firewall, non si trova su una specifica macchina, ma può essere installato su più macchine contemporaneamente in maniera distribuita, anche simultaneamente su più reti. Questo consente ad organizzazioni con tante reti e sottoreti di gestire in maniera scalabile la protezione di ciascuna di esse.

**Perimeter Firewall:** Il classico firewall che si trova "sul bordo" tra una LAN e la rete Internet. Analizza il traffico in entrata e in uscita, allo scopo di difendere la LAN dalle minacce esterne.

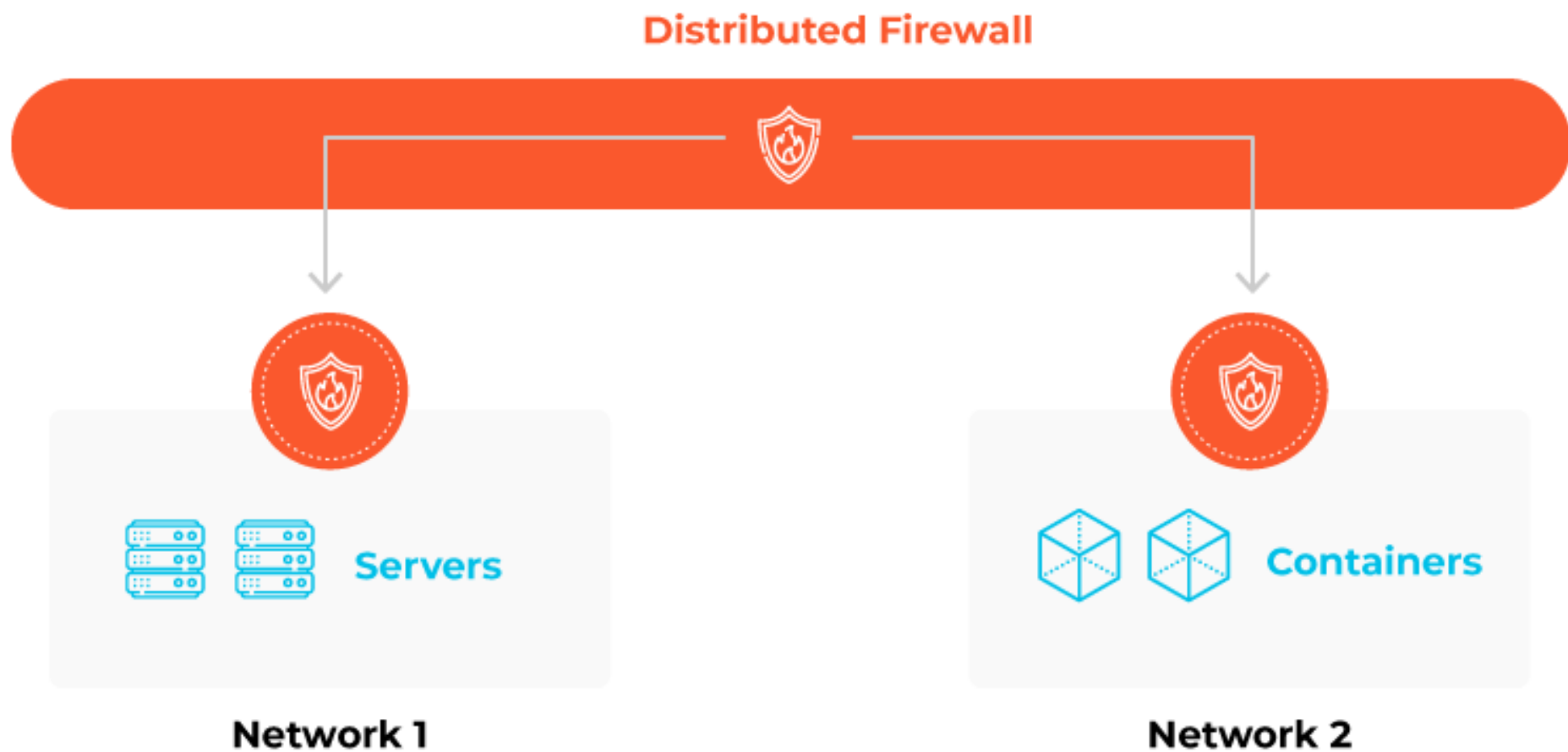
# Internal Firewall

---



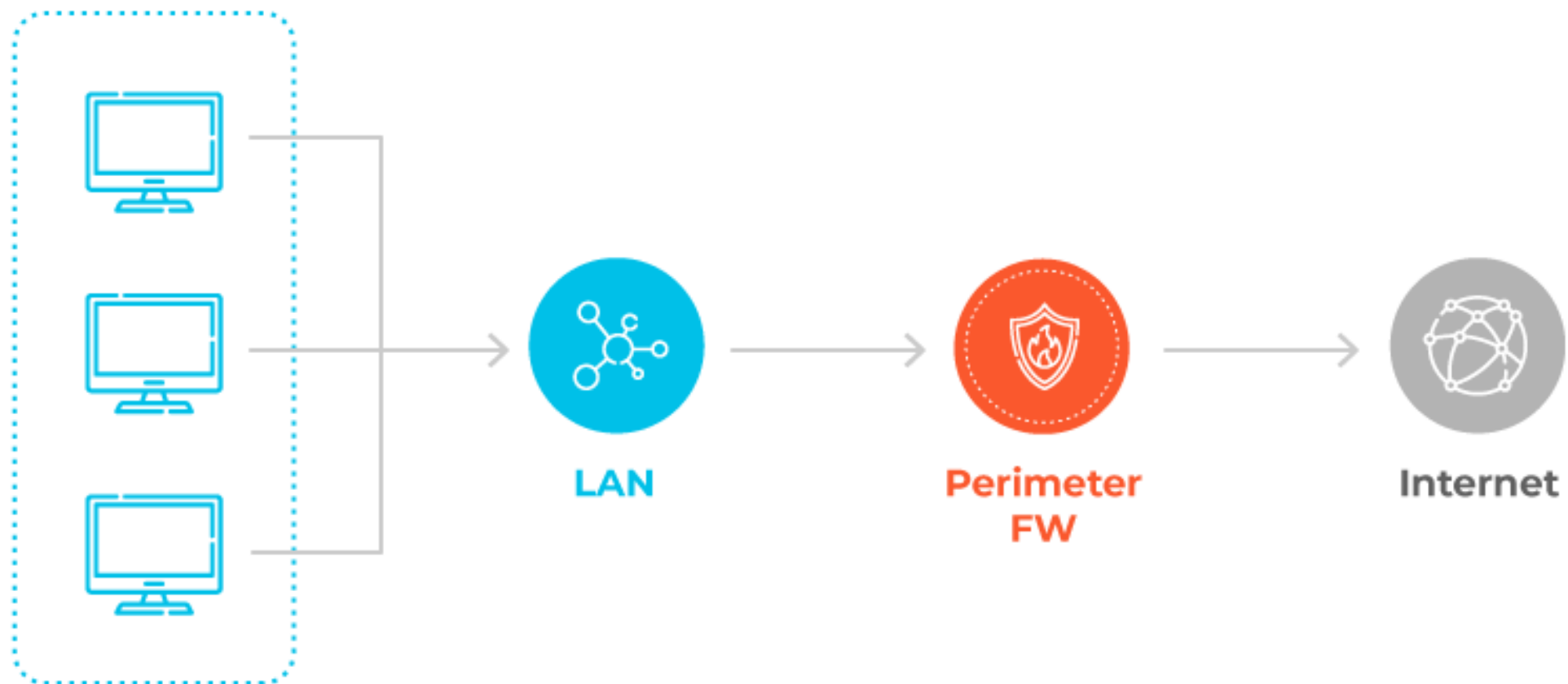
# Distributed Firewall

---



## Perimeter Firewall

---





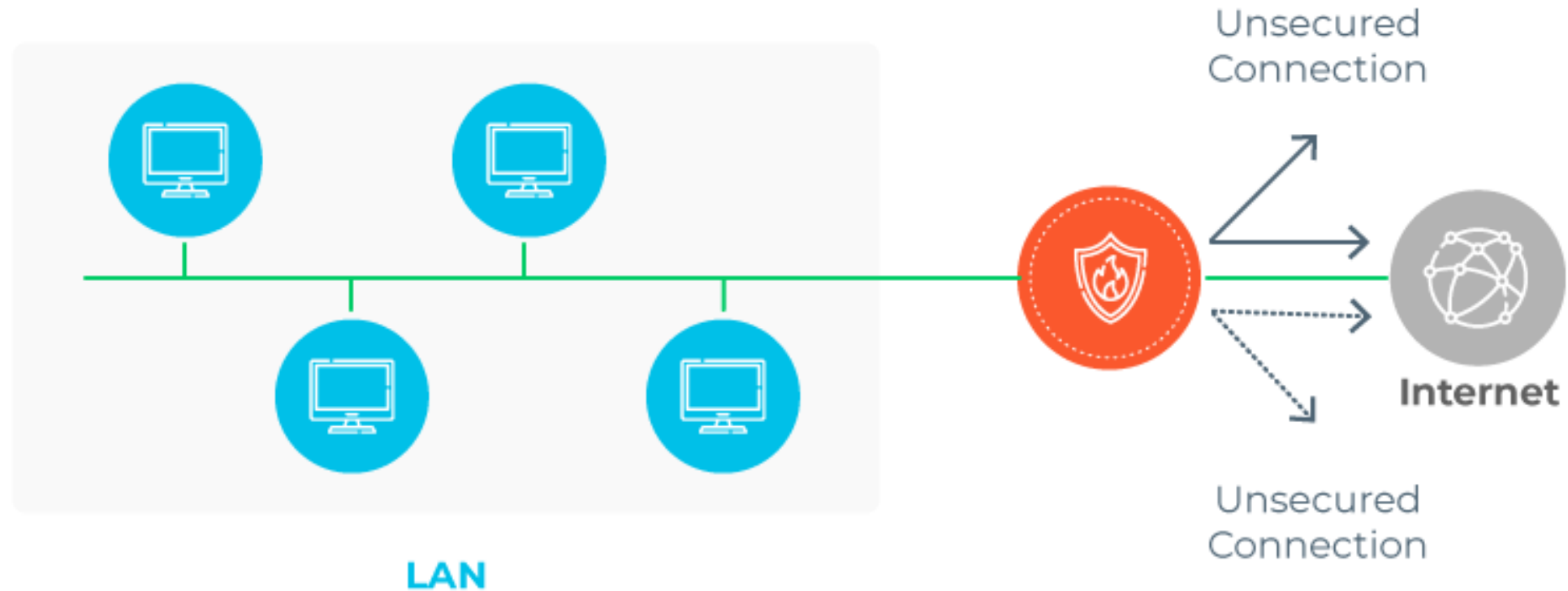
# Firewall

## Form Factors

**Hardware Firewall:** Dispositivo fisico che effettua le operazioni di analisi e filtraggio del traffico. Viene fisicamente interposto tra una rete e l'altra (e.g., connettendolo al router e permettendo ai dispositivi di accedere ad Internet solo attraverso il firewall).

**Software Firewall:** Può svolgere le stesse funzioni del firewall hardware, ma viene invece installato su una macchina come un normale software. Utile per scenari in cui non è possibile (o è difficile) usare firewall fisici (e.g. cloud, container). Essendo dei software a tutti gli effetti, è possibile estenderne le funzionalità nel tempo.

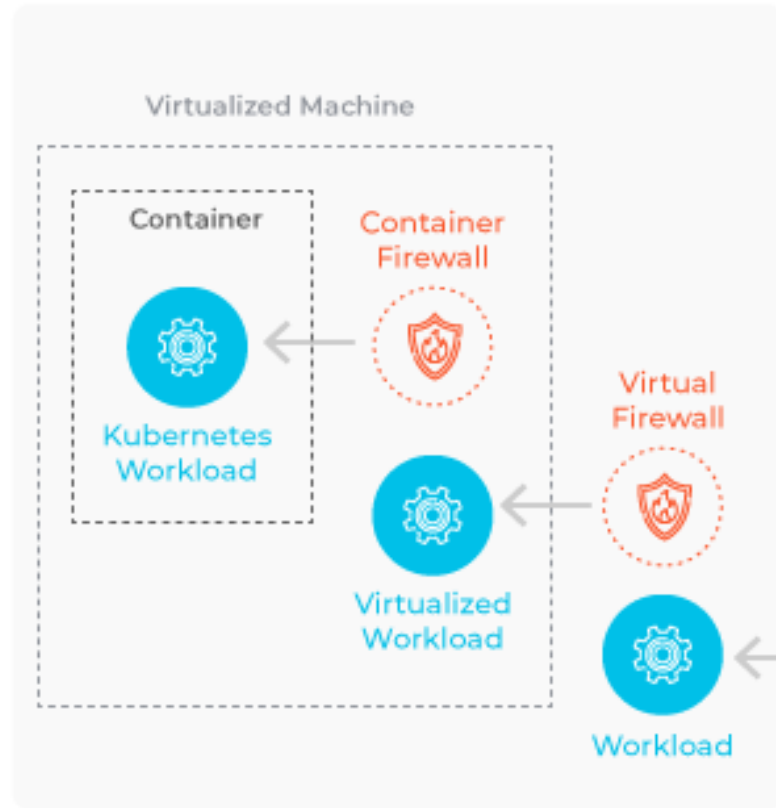
# Hardware Firewall





# Software Firewall

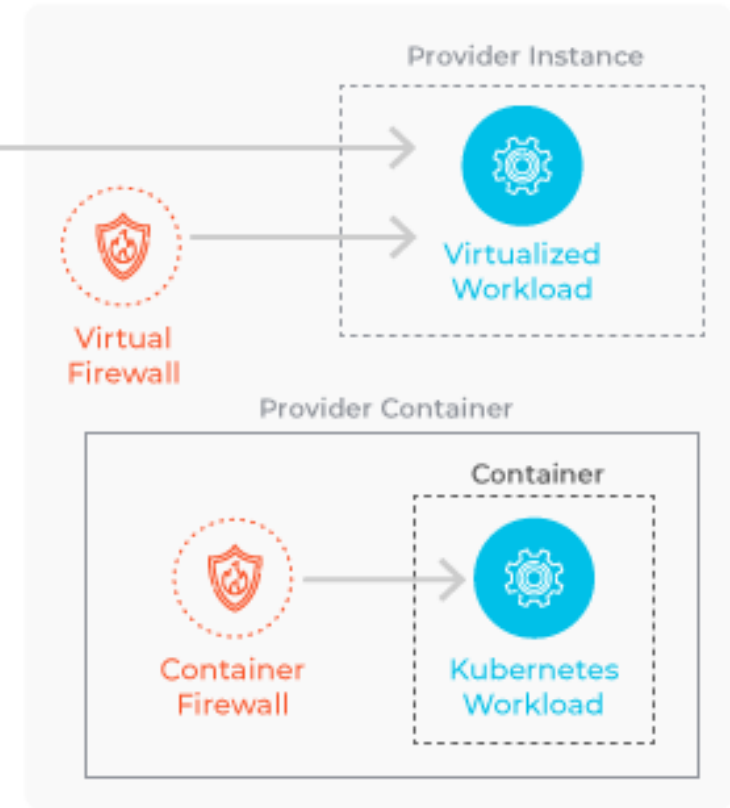
## Private Cloud

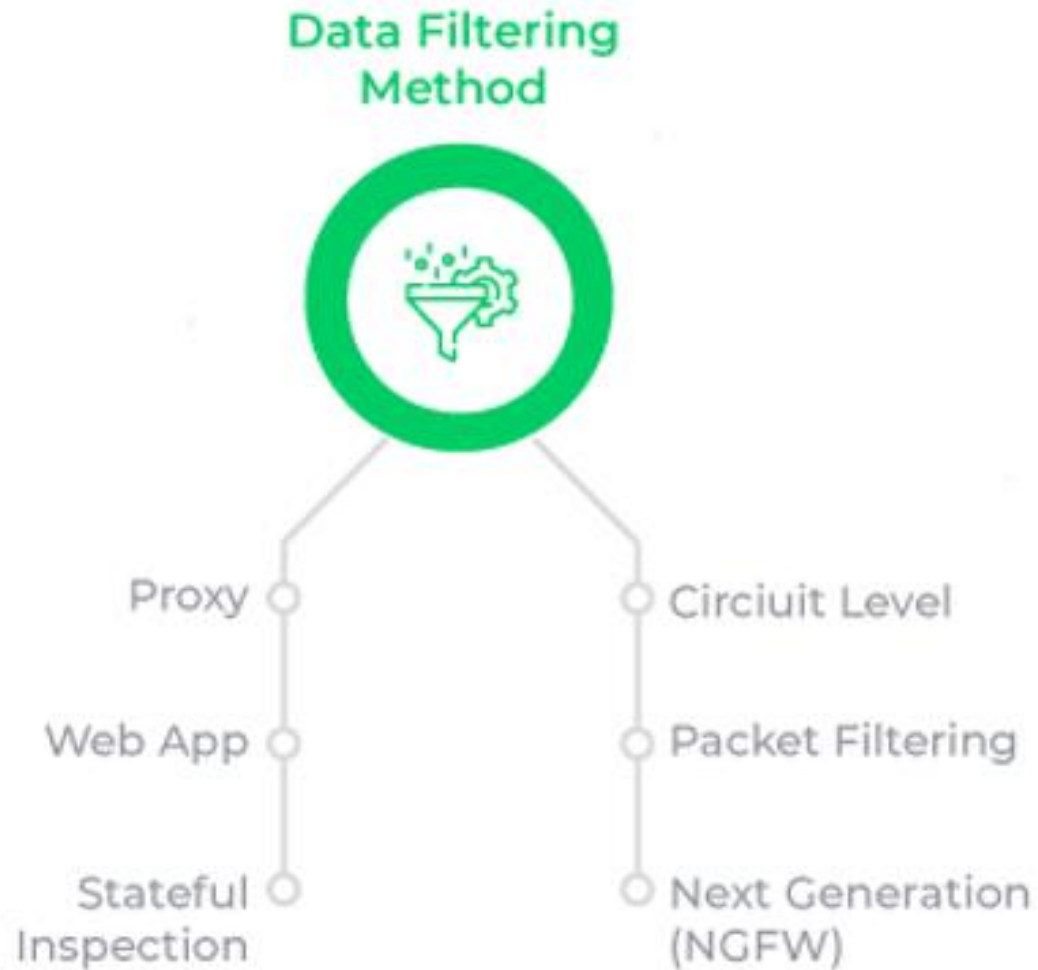


  
**Cloud Firewall (FWaaS)**

  
**Managed Service Firewall**

## Public Cloud





# Firewall

## Data Filtering Method

# Firewall

## Packet-Filtering Firewall

---

Opera al livello 3 del modello ISO/OSI, ovvero il livello di rete.

L'amministratore definisce delle regole (allow/deny) sul traffico della suite TCP/IP (TCP, UDP, ICMP, IGMP).

I parametri del pacchetto possono essere presi in considerazione per sviluppare le regole. Alcuni esempi includono:

- Indirizzo IP di origine
- Indirizzo IP di destinazione
- Numero porta di origine
- Numero porta di destinazione
- Protocollo utilizzato

# Firewall

## Packet-Filtering Firewall: Limiti

---

Si limita a far passare i pacchetti corrispondenti alle regole "allow", droppando gli altri.

Non ha memoria dei pacchetti ricevuti precedentemente, ovvero, è **stateless**.

Le informazioni di origine sul pacchetto, nonché altri campi, sono modificabili dall'attaccante e potrebbero bypassare le regole del firewall.

Vulnerabile ad attacchi di routing, in cui l'attaccante fa passare prima il pacchetto da un host che sa essere "allowed".

# Packet Filtering Firewall



# Firewall

## Stateful Inspection Firewall

---

Opera ai livelli 3 e 4 del modello ISO/OSI, ovvero il livello di rete e trasporto.

Oltre a quanto già fatto dal packet-filtering firewall, lo stateful inspection firewall memorizza i pacchetti processati precedentemente e tiene traccia delle connessioni aperte.

Analizza gli **header** dei pacchetti per verificare se possono essere malevoli (**packet inspection**). Questo in genere avviene verificando se tali header sono coerenti con altri pacchetti che sono già passati in precedenza dal firewall.

- Esempio: durante un port scan, lo SIF potrebbe identificare tutto il traffico dell'attaccante come sospetto e bloccarlo. Il PFF invece si atterrebbe alle regole specificate dall'amministratore, permettendo il completamento del port scan.

# Firewall

## Packet Inspection

---

Alcuni Stateful Inspection Firewall effettuano anche la **Deep Packet Inspection** (DPI)

**Packet Inspection:** Analisi del contenuto degli header dei pacchetti (e.g., indirizzi IP di sorgente e destinazione, porte di sorgente e destinazione, protocollo utilizzato).

**Deep Packet Inspection:** Analisi del contenuto degli header e del contenuto dei pacchetti. Il firewall decide poi, in base all'analisi effettuata, se far passare il pacchetto oppure no. Due tecniche principali di analisi:

- **Protocol Anomaly:** Di default non viene fatto passare nessun pacchetto ("default deny"). Solo i pacchetti che corrispondono ad un determinato set di regole possono passare.
- **Pattern/Signature Matching:** Di default vengono fatti passare tutti i pacchetti ("default permit"). I pacchetti che corrispondono ad un determinato set di regole vengono bloccati.

# Firewall

## Stateful Inspection Firewall: Limiti

---

Regole non sempre chiare e variabili nel tempo: rischio di falsi positivi (FP) e falsi negativi (FN).

Esempio di FP: uno SIF potrebbe identificare l'invio di SYN, ACK e/o FIN come traffico malevolo e non completare eventuali handshake o data transfer.

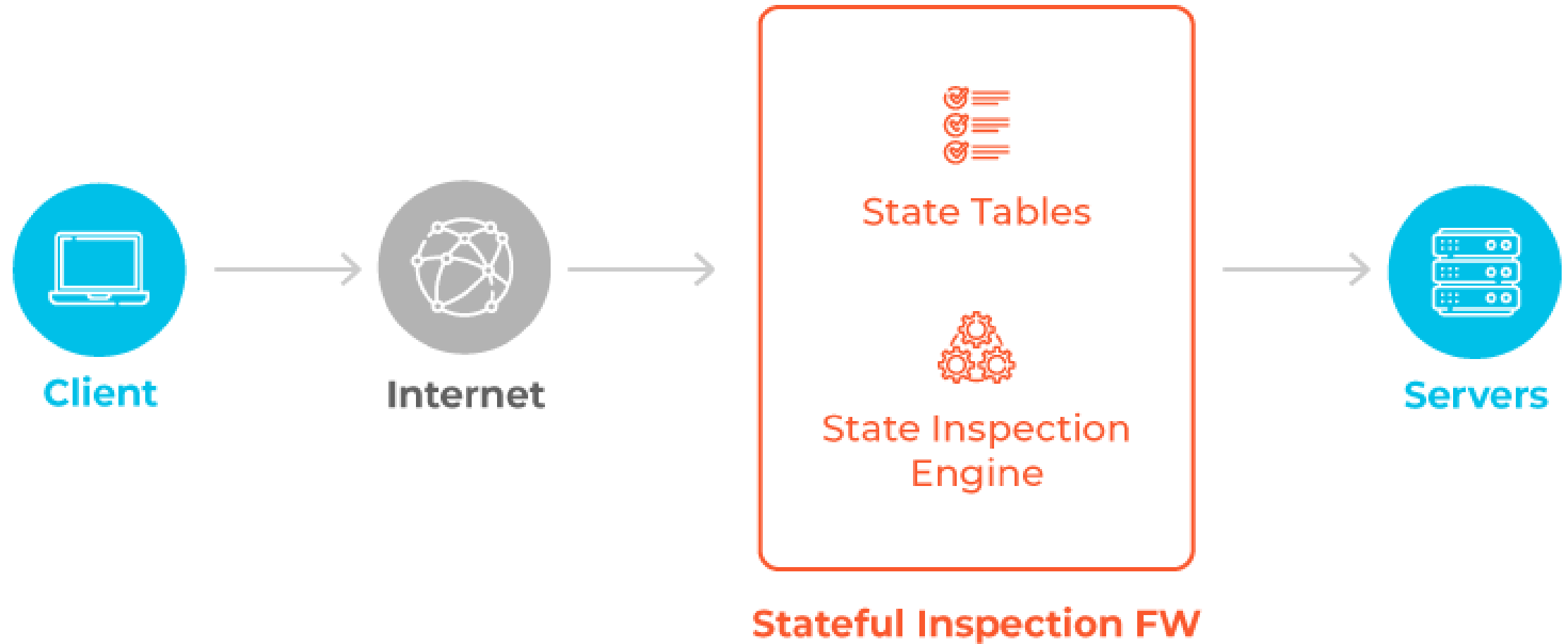
Sia SIF che PFF possono inoltre essere affetti da anomalie (Cuppens et al., Handling Stateful Firewall Anomalies, 2012)

- *Shadowing*: Let  $R$  be a set of filtering rules. Then, rule  $R_i$  is shadowed iff it never applies because all the packets that  $R_i$  may match, are previously matched by another rule, or combination of rules, with higher priority in order;
- *Redundancy*: Let  $R$  be a set of filtering rules. Then, rule  $R_i$  is redundant iff (1)  $R_i$  is not shadowed by any other rule (or combination of rules); and (2) when removing  $R_i$  from  $R$ , the filtering result does not change.



# Stateful Inspection Firewall

---



# Firewall

## Circuit Level Gateway

---

Opera al livello 5 del modello ISO/OSI, ovvero il livello di sessione.

Solitamente utilizzato per creare VPN.

Verifica se i pacchetti inviati sono parte legittima di una sessione mediante i seguenti step:

1. Verifica della genuinità degli handshake TCP e degli attori coinvolti;
2. Creazione di un circuito virtuale per la durata della sessione;
3. Filtraggio dei pacchetti sugli indirizzi e le porte specificate, a seconda della loro validità nel contesto della sessione.

# Firewall

## Circuit Level Gateway

---

### Pro:

Ogni computer esterno alla rete vede il traffico arrivare dal gateway, per cui si mantiene un buon livello di anonimità dei computer all'interno della rete.

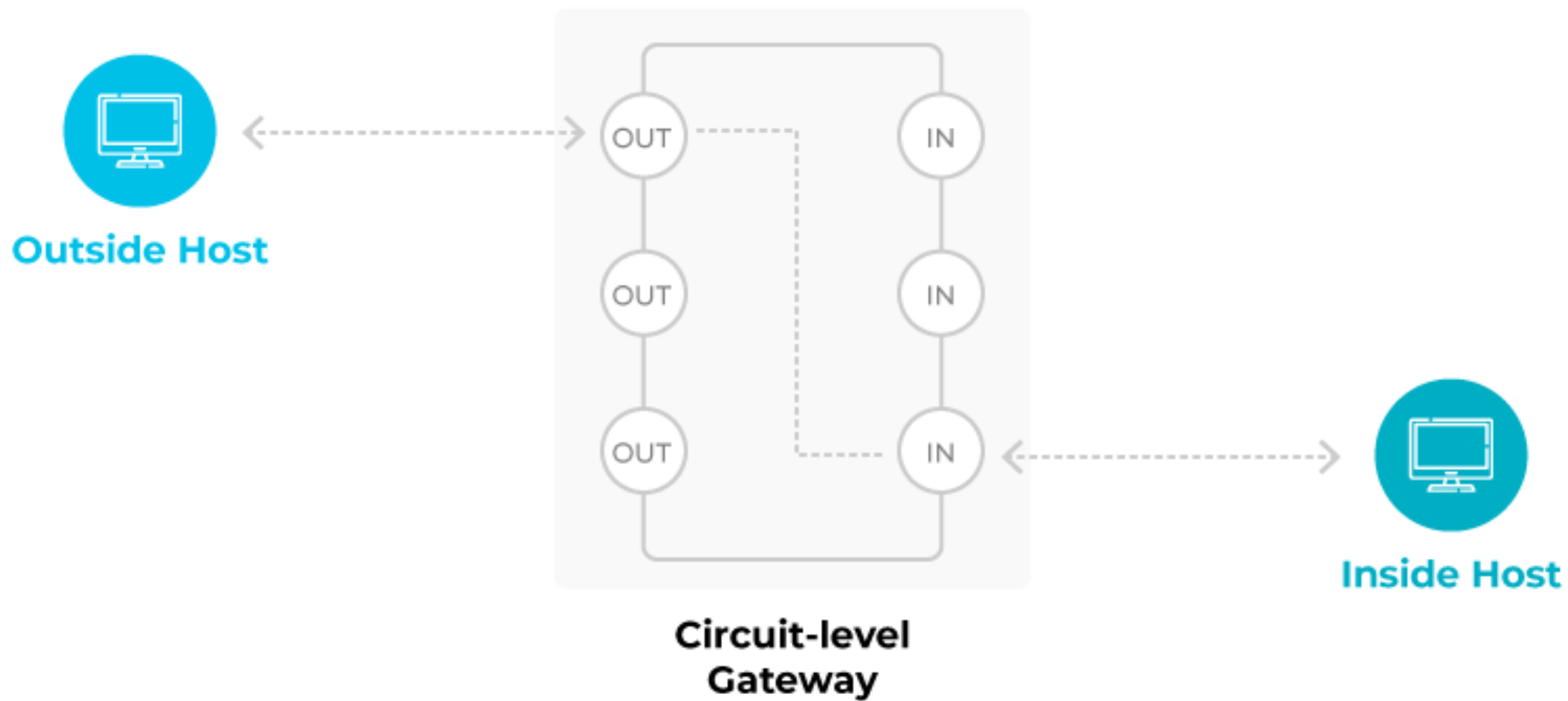
Configurazione facile, non degrada le prestazioni della rete (talvolta addirittura le migliora!).

### Contro:

Se il pacchetto è valido all'interno della sessione, non è possibile scartarlo in quanto non si possono impostare altre tipologie di filtri.

Questo permette a qualsiasi tipologia di contenuto (e.g., malware, payload malevoli) di passare indisturbato all'interno del canale.

## Circuit-level Gateway



# Firewall

## Proxy Firewall / Application Firewall

---

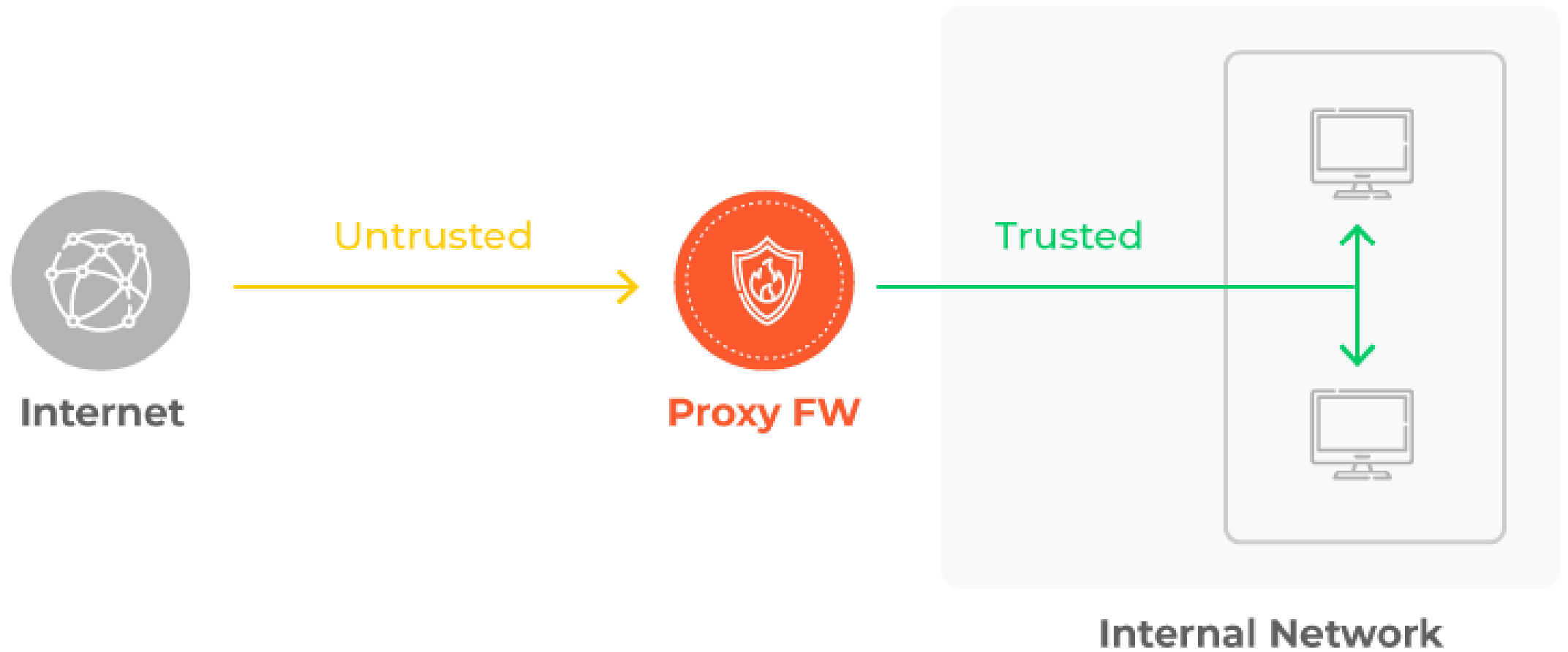
Opera al livello 7 del modello ISO/OSI, ovvero il livello applicativo.

Fa da proxy, ovvero effettua un Man-In-The-Middle (MITM) tra i **client interni alla rete** e i **server esterni**.

Dispone di un proprio IP, e protegge l'identità dei client interni, in quanto i server esterni vedranno arrivare le richieste dal proxy.

Il proxy firewall può anche analizzare il contenuto di ogni pacchetto prima di decidere se inviarlo o meno (**deep packet inspection**).

# Proxy Firewall



# Firewall

## Web Application Firewall

---

Opera al livello 7 del modello ISO/OSI, ovvero il livello applicativo.

Fa da **reverse proxy**, ovvero fa da MITM tra i **client esterni alla rete** e i **server interni**.

In questo caso, è l'identità dei server interni ad essere protetta da client esterni untrusted!

Dietro il WAF potrebbero esserci multipli server a rispondere alle richieste dei client. Il tutto è invisibile all'esterno, in quanto i client possono interagire solo col WAF.

# Firewall

## Web Application Firewall

---

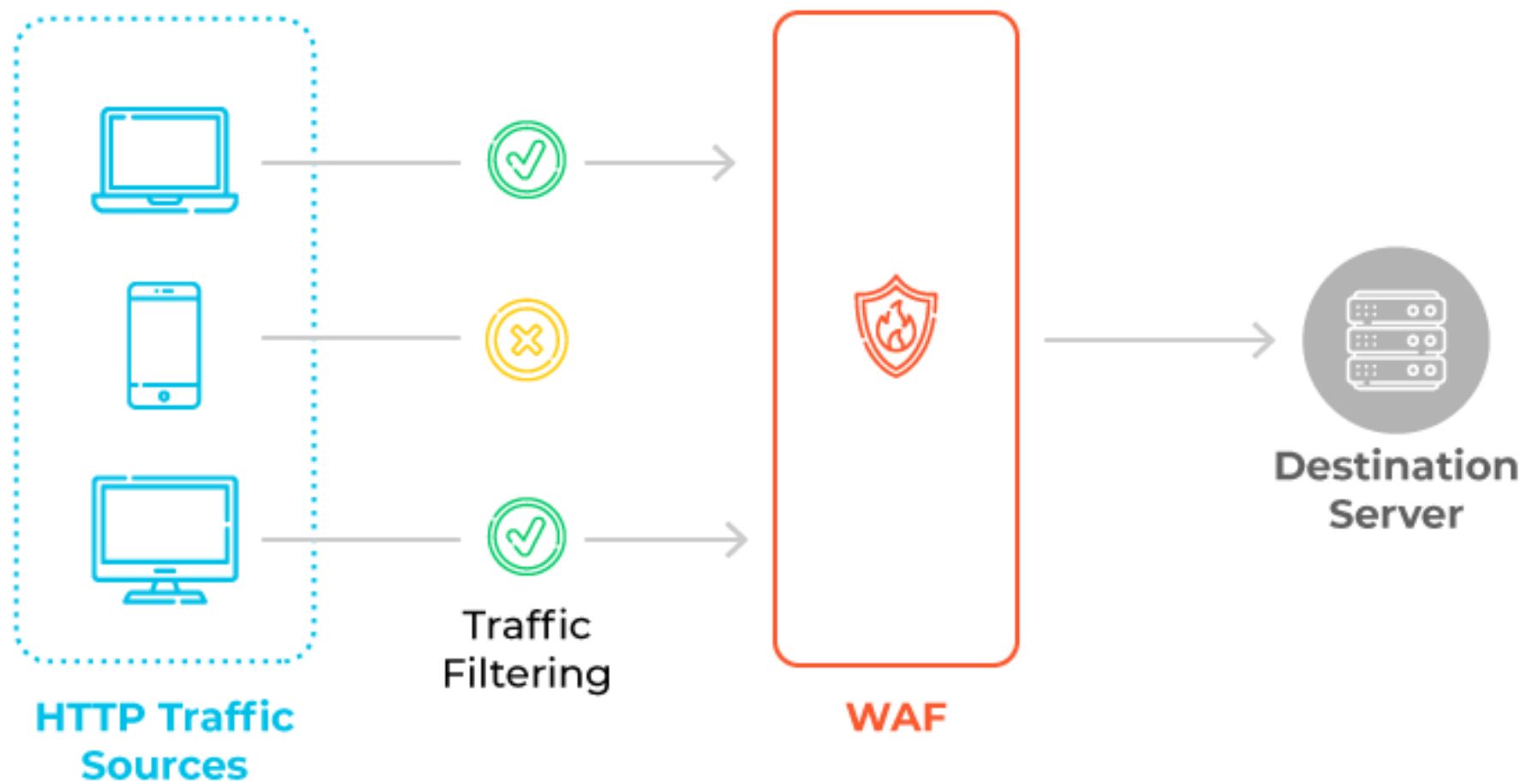
Anche il WAF effettua solitamente la Deep Packet Inspection, spesso cercando payload noti che servono a sfruttare vulnerabilità comuni.

Esempio 1: ' or 1=1 --+ è un tipico payload per SQL Injection, serve a bypassare i login.

Esempio 2: ;cat /etc/passwd è un tipico payload per OS Command Injection, serve più che altro come Proof of Concept (PoC) in quanto oggi non ci sono più molte informazioni rilevanti in /etc/passwd.



# Web Application Firewall



# Firewall

## Next Generation Firewall (NGFW)

---

Combinano in un'unica soluzione quasi tutto quello che abbiamo visto.

Al contrario degli altri firewall, non vi è una definizione precisa in letteratura di cosa sia un NGFW.

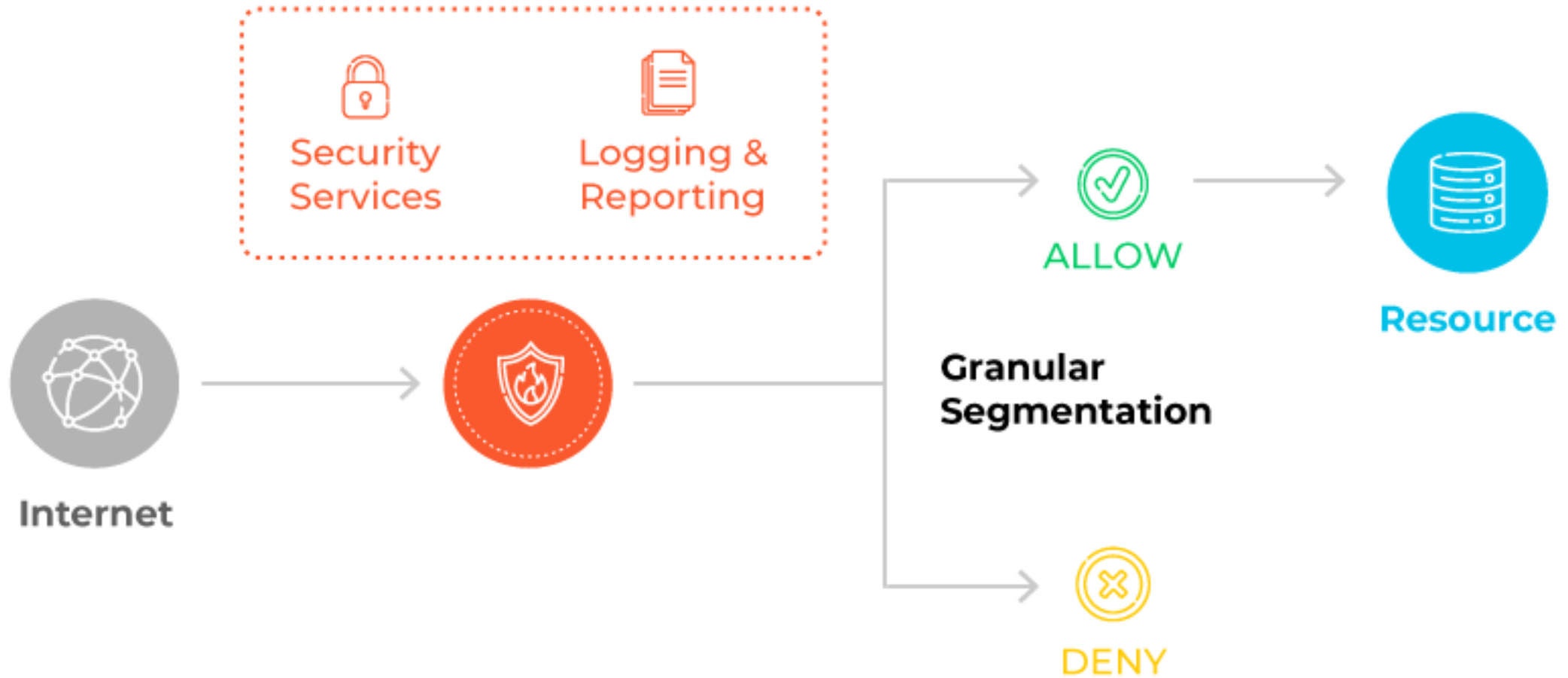
Potremmo dire che un NGFW è "un firewall che tenta di combinare due o più firewall tra quelli mostrati", ma anche questo sembra riduttivo.

Sembrano fare qualsiasi cosa! :)

<https://www.fortinet.com/content/dam/fortinet/assets/data-sheets/fortigate-7000f-series.pdf>

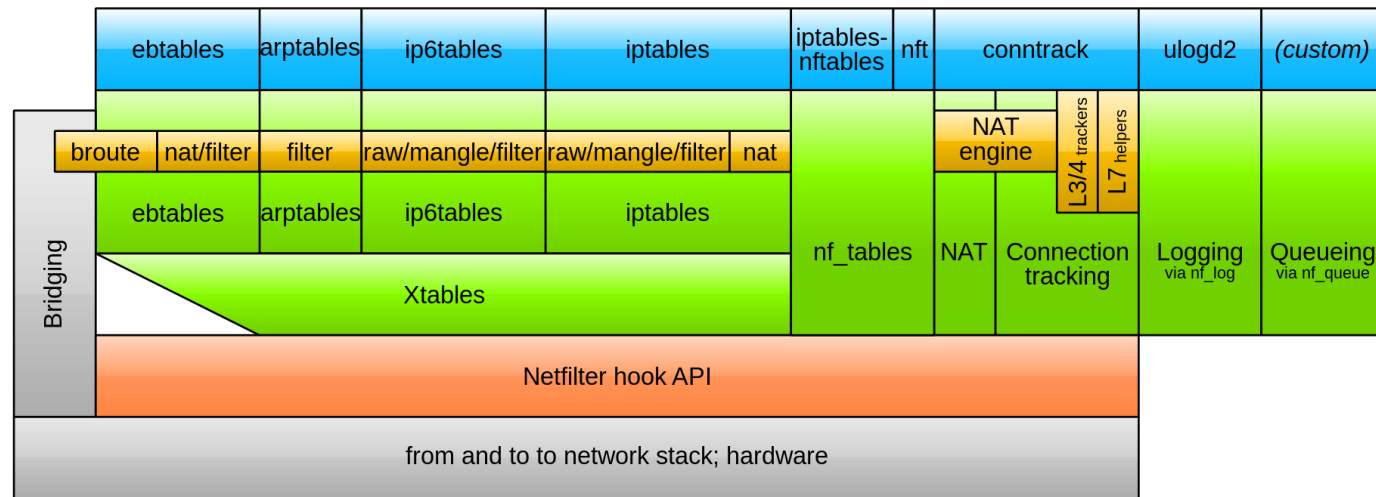
Ad esempio, il firewall linkato combina il classico packet filtering con TLS inspection, threat protection, IPS e molto altro.

# Next-Generation Firewall



# Netfilter components

Jan Engelhardt, last updated 2014-02-28 (initial: 2008-06-17)



- Userspace tools
- Netfilter kernel components
- other networking components

## Netfilter

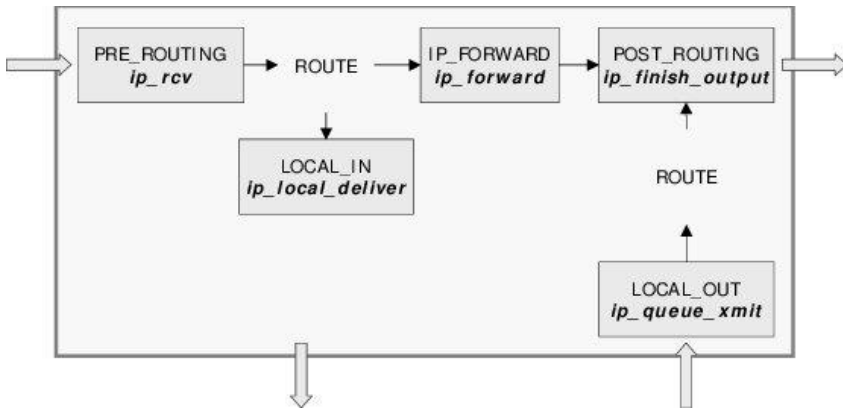
Packet filtering framework fornito dal kernel Linux

Dispone di cinque hook nel networking stack, ai quali è possibile agganciare delle funzioni personalizzate.

Una funzione verrà richiamata quando un pacchetto attraverserà il rispettivo hook all'interno dello stack di rete.

Tali funzioni non fanno altro che gestire il filtraggio dei pacchetti!

# Netfilter Hooks



**NF\_IP\_PRE\_ROUTING:** Aggancia i pacchetti in entrata, subito dopo il loro ingresso nel network stack.

**NF\_IP\_LOCAL\_IN:** Aggancia i pacchetti in entrata, se questi sono destinati al sistema locale.

**NF\_IP\_FORWARD:** Aggancia i pacchetti in entrata che devono essere inoltrati ad un altro host.

**NF\_IP\_LOCAL\_OUT:** Aggancia i pacchetti in uscita che vengono creati dal sistema locale, subito dopo il loro ingresso nel network stack.

**NF\_IP\_POST\_ROUTING:** Aggancia i pacchetti in uscita (outgoing + forward) subito prima che questi vengano spediti.

Fonte immagine: Aguero et al., On the Implementation and Experimental Characterization of the Dynamic Source Routing Protocol for Mobile Ad Hoc Networks, 2003

# Iptables

---

Firewall su Linux che interagisce con Netfilter per filtrare i pacchetti.

**System Protected:** Prevalentemente Host-Based

**Network Placement:** Prevalentemente Internal, in rari scenari Perimeter (dipende dalla posizione della macchina su cui gira Iptables!)

**Form Factor:** Software

**Data Filtering Method:** Packet Filtering + Stateful Inspection (con conntrack)

# Iptables

## Tabelle

---

Iptables utilizza, come suggerisce il nome, delle tabelle per organizzare le regole di gestione dei pacchetti.

Ogni tabella ha uno specifico scopo. Il numero cinque è ridondante (i.e., ci sono 5 tabelle).

**Filter:** La tabella più usata, contiene le regole per decidere se un pacchetto può essere inoltrato alla sua destinazione o deve essere droppato.

**NAT:** In questa tabella risiedono le regole per il natting. Quando i pacchetti entrano nel network stack, le regole in questa tabella determinano come gli indirizzi di sorgente e destinazione debbano cambiare.

**Mangle:** Contiene le regole per l'alterazione degli header dei pacchetti. Ad esempio, si potrebbero voler aumentare o diminuire il numero di hop che un pacchetto può fare, modificando il suo TTL (Time To Live).

# Iptables

## Tabelle

---

**Raw:** Utilizzata solo per la stateful inspection, serve a configurare eccezioni nel connection tracking. Ovvero, tiene traccia di quali pacchetti NON devono essere analizzati mediante stateful inspection.

**Security:** Utilizzata per definire i parametri che possono essere utilizzati per marcare pacchetti e connessioni con dei security contexts aggiuntivi, ovvero SECMARK (per i pacchetti) e CONNSECMARK (per le sessioni).

- SECMARK viene piazzato su un pacchetto se corrisponde ad una regola nella security table
- CONNSECMARK in maniera simile, viene piazzato su tutti i pacchetti di una sessione.

Tale security context può essere utilizzato poi da altre soluzioni (e.g., il Mandatory Access Control di SELinux) per capire cosa fare con il pacchetto o con la sessione.

SELinux è un modulo di sicurezza del kernel Linux che offre funzionalità di sicurezza.



# Iptables

## Chains

---

Dentro ogni tabella, le regole sono organizzate in delle catene (chains) separate.

Ad ogni hook corrisponde una catena: quindi, potremmo dire che le regole dentro una catena definiscono la funzione callbackata dall'hook.

Le catene sono quindi cinque:

- PREROUTING (agganciata all'hook `NF_IP_PRE_ROUTING`)
- INPUT (agganciata all'hook `NF_IP_LOCAL_IN`)
- FORWARD (agganciata all'hook `NF_IP_FORWARD`)
- OUTPUT (agganciata all'hook `NF_IP_LOCAL_OUT`)
- POSTROUTING (agganciata all'hook `NF_IP_POST_ROUTING`)

L'utente può creare altre catene, ma non essendoci un hook apposito, queste vanno raggiunte mediante un jump da una delle cinque catene sopracitate.

# Iptables

## Conntrack (Connection Tracking)

---

- Modulo di Iptables che implementa la stateful inspection
- Permette di filtrare i pacchetti sulla base di regole aggiuntive basate sullo stato di tali pacchetti
- In Conntrack, i pacchetti possono avere quattro stati:
  - NEW: Il pacchetto è relativo ad una nuova connessione
  - ESTABLISHED: Il pacchetto è relativo ad una connessione già esistente
  - RELATED: Il pacchetto è correlato ad una connessione già esistente, ma non ne fa parte
  - INVALID: Il pacchetto non è relativo ad una nuova connessione o ad una connessione già esistente

# Iptables

## Comandi e Tutorial

---

Comandi utili in generale:

<https://www.digitalocean.com/community/tutorials/iptables-essentials-common-firewall-rules-and-commands>

Tutorial: <https://www.digitalocean.com/community/tutorials/how-to-set-up-a-firewall-using-iptables-on-ubuntu-14-04>

Firewall Strategy e Templates:

<https://www.digitalocean.com/community/tutorials/how-to-implement-a-basic-firewall-template-with-iptables-on-ubuntu-14-04>

# Iptables

## Problemi

---

Iptables ha diversi problemi che col tempo si sono acuiti. Di seguito alcuni:

- Tutte le tabelle e le chain di default vengono sempre create, anche se viene effettivamente utilizzata solo una tabella e solo una chain. La sola presenza delle catene inutilizzate apparentemente riesce a degradare le prestazioni.
- IPv6 non viene gestito direttamente da Iptables, ma serve utilizzare un diverso tool chiamato Ip6tables. Questo vuol dire che le modifiche alla configurazione di un tool non si propagano direttamente nell'altro, e che l'operazione va fatta manualmente (con possibili errori umani in mezzo).

# Iptables

## Problemi

---

- Il supporto di nuovi protocolli richiede un upgrade del kernel. Questo può creare problemi di compatibilità con altri tool e servizi che girano sulla macchina (oltre a richiedere tempo che potremmo non avere, se stiamo amministrando un server di produzione).

Questi e altri problemi sono stati risolti nel 2014 dal successore di Iptables (sempre all'interno di Netfilter), ovvero Nftables

# Nftables

---

- Le basi sono le stesse di Iptables, cambia la sintassi e il fatto che nessuna catena o tabella è creata di default.
- Dovremo quindi creare noi le tabelle e le catene, e agganciare queste ultime ad un hook!
- Un hook aggiuntivo! "Ingress" aggancia i pacchetti dopo che questi vengono rilasciati dal NIC (Network Interface Controller), ovvero, ancora prima del prerouting.

# Nftables

## Tabelle

---

- Tre **tipi** di tabelle (per quale scopo verrà usata):
  - Filter
  - Route (simile al vecchio "mangle" di Iptables, per il routing in uscita)
  - NAT
- Sei **famiglie** di tabelle (in che ambito verrà usata):
  - ip, ip6, bridge, arp li avevamo come tool singoli in Iptables!
  - inet (ip+ip6)
  - netdev (unica vera novità: aggancia i pacchetti su una specifica interfaccia e possiamo usare l'hook ingress, ad esempio per droppare DDoS.)

# Nftables

## Catene


---

- Due **tipi** di catene, base e regular:
  - **Base chain:** viene agganciata ad un hook di Netfilter. Dato che con questa funzionalità è possibile agganciare più catene ad un hook, vi è un sistema di priorità, per cui possiamo decidere quale catena si attiva per prima, tra quelle assegnate allo stesso hook
  - **Regular chain:** come le user-defined chains di Iptables, si attivano solo se c'è un esplicito jump da un'altra catena.
- Troppi nuovi concetti? Esiste un tool "iptables-translate" per i sysadmin pigri che non vogliono imparare un'altra sintassi!




# Tipiche Conversazioni da Sysadmin

---

 **user\_n0mad** • 2 anni fa


I prefer to use plain iptables.

⊖ ↑ 21 ↓ 🏆 Premio 📌 Condividi ...

 **maybegeek** • 2 anni fa


Wasn't iptables deprecated. Do you mean nftables, it's successor?

⊖ ↑ 14 ↓ 🏆 Premio 📌 Condividi ...

 **user\_n0mad** • 2 anni fa

No, I still use iptables cause I haven't [REDACTED] sat down to learn nftables STILL.

⊖ ↑ 9 ↓ 🏆 Premio 📌 Condividi ...

 **maybegeek** • 2 anni fa

Yeah, me neither 🤔

↑ 5 ↓ 🏆 Premio 📌 Condividi ...

Per questo motivo, se utilizzate Iptables oggi, comunque interagite con nftables!

# Nftables

## Comandi e Tutorial

---

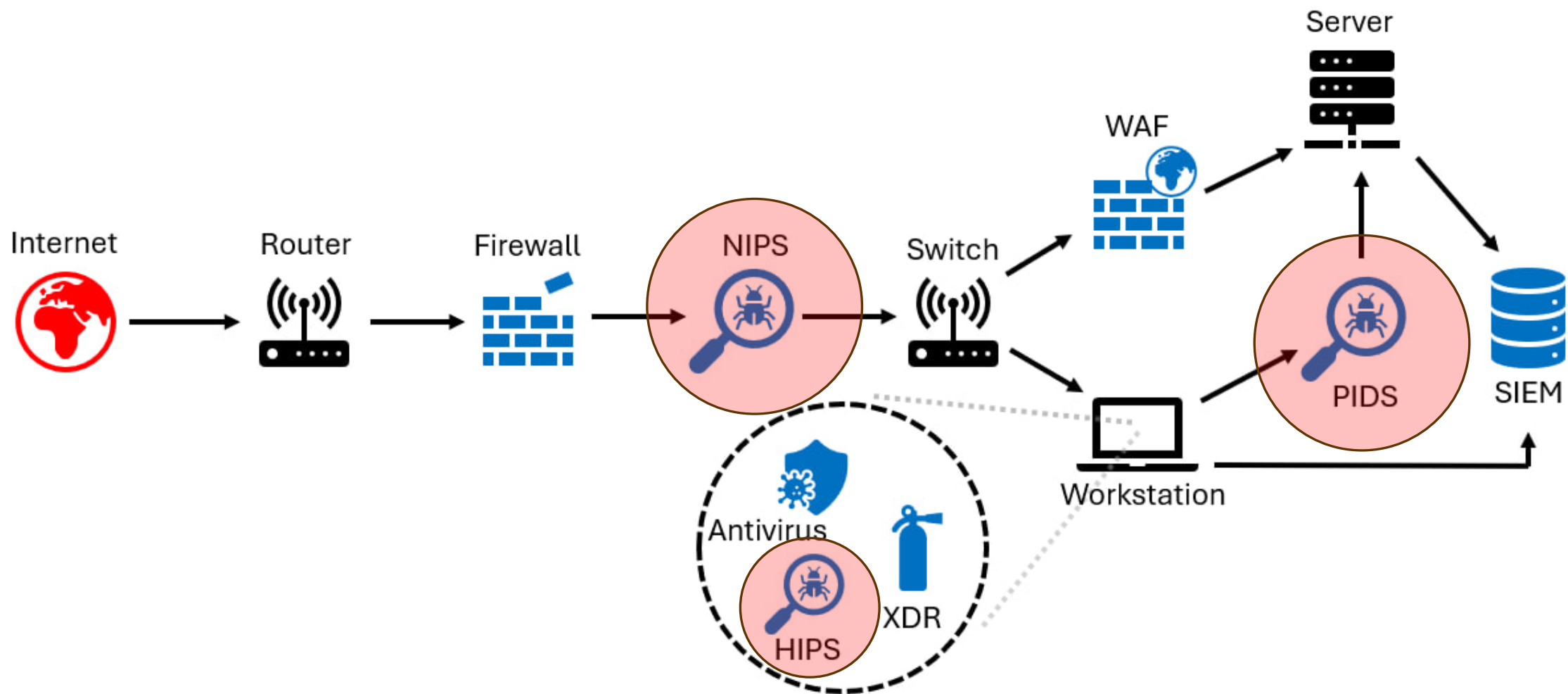
Tutorial: <https://www.linode.com/docs/guides/how-to-use-nftables/>

Documentazione ufficiale: [https://wiki.nftables.org/wiki-nftables/index.php/Main Page](https://wiki.nftables.org/wiki-nftables/index.php/Main_Page)

# Interfacce

---

- Non tutti hanno il tempo e l'esperienza necessaria per configurare correttamente un firewall.
- Per questo, esistono delle interfacce che semplificano l'operazione di gestione dei firewall per gli utenti meno esperti.
- Su Linux, due interfacce principali: UFW (Uncomplicated Firewall) e FirewallD.
- Esempio (Tutorial UFW):  
<https://www.digitalocean.com/community/tutorials/how-to-set-up-a-firewall-with-ufw-on-ubuntu>
- Esempio (Comandi utili UFW):  
<https://www.digitalocean.com/community/tutorials/ufw-essentials-common-firewall-rules-and-commands>



# IDS vs IPS (in teoria)

<https://www.paloaltonetworks.com/cyberpedia/what-is-an-intrusion-detection-system-ids>

	Intrusion Prevention System	IDS Deployment
<b>Placement in Network Infrastructure</b>	Part of the direct line of communication (inline)	Outside direct line of communication (out-of-band)
<b>System Type</b>	Active (monitor & automatically defend) and/or passive	Passive (monitor & notify)
<b>Detection Mechanisms</b>	1. Statistical anomaly-based detection 2. Signature detection: <ul style="list-style-type: none"><li>- Exploit-facing signatures</li><li>- Vulnerability-facing signatures</li></ul>	1. Signature detection: <ul style="list-style-type: none"><li>- Exploit-facing signatures</li></ul>

# Tipi di IDS

---

- Network-based (NIDS)
  - Monitora il traffico in entrata e in uscita da una determinata sottorete.
- Host-based (HIDS)
  - Viene installato su uno specifico endpoint (e.g., un computer) e monitora solo il suo traffico.
- Protocol-based (PIDS)
  - Viene solitamente installato sui server, analizza i messaggi scambiati mediante uno o più protocolli di rete per identificare anomalie o stati teoricamente irraggiungibili.
- Application Protocol-Based (APIDS)
  - Analizza i messaggi scambiati mediante uno o più protocolli a livello di applicazione, ad esempio il Tabular Data Stream Protocol viene utilizzato per interrogare database MS-SQL.  
([https://learn.microsoft.com/en-us/openspecs/windows\\_protocols/ms-tds/b46a581a-39de-4745-b076-ec4dbb7d13ec](https://learn.microsoft.com/en-us/openspecs/windows_protocols/ms-tds/b46a581a-39de-4745-b076-ec4dbb7d13ec) )

# Tipi di IPS

---

- Network-based (NIPS)
  - Come il NIDS, viene posizionato dentro una sottorete e monitora il traffico in entrata e in uscita. Può bloccare eventuale traffico che venga identificato come sospetto.
- Host-based (HIPS)
  - Come il HIDS, viene installato su un endpoint e monitora il suo traffico, eventualmente bloccandolo.
- Wireless-based (WIPS)
  - Si connette ad una rete wireless e cerca di identificare dispositivi non autorizzati che vi siano connessi, allo scopo di deautenticarli.
- Network Behaviour Analysis (NBA)
  - Viene posizionato all'interno di una sottorete per identificare traffico insolito. È in grado di rivelare la presenza di malware con funzionalità di rete o lo sfruttamento di vulnerabilità 0-day.

# IDS vs IPS (in pratica)

---

Difficile trovare oggi un IDS che non abbia anche funzionalità IPS (magari disattivate di default, ma attivabili su richiesta dell'utente).

Ciò è probabilmente dovuto al fatto che oggi la comunità informatica è più abituata alla gestione di minacce ed intrusioni, rispetto a qualche decennio fa.

Prevenzione dell'intrusione come requisito funzionale di un sistema (**security by design**).



# Snort

---

<https://www.snort.org/>

Network-Based IPS (NIPS)

Sviluppato nel 1998 da Martin Roesch, ed acquistato da CISCO nel 2013.

Open source, regole community-driven, aggiornamenti rilasciati più volte a settimana.

La versione attuale è Snort 3, rilasciata nel 2021.

# Snort

## Pros and Cons

---

- Snort 1 e 2 erano single-threaded, la versione 3 è multi-threaded.
- Ottima gestione della RAM.
- In condizioni operative "normali" è più preciso dei suoi competitor, ma ha un tasso di falsi positivi abbastanza alto.
  - È possibile reportare facilmente i falsi positivi attraverso il sito web.
- Sotto stress (e.g., attacchi DoS, scenari con molto throughput) tende a perdere pacchetti e quindi a dare falsi negativi.

[https://pure.port.ac.uk/ws/portalfiles/portal/79753845/A Comparative Analyses of Snort 3 and Suricata.pdf](https://pure.port.ac.uk/ws/portalfiles/portal/79753845/A_Comparative_Analyses_of_Snort_3_and_Suricata.pdf)



### Rule Header

alert tcp \$EXTERNAL\_NET \$HTTP\_PORTS -> \$HOME\_NET any

### Rule Options

#### Message

msg: "BROWSER-IE Microsoft Internet Explorer  
CacheSize exploit attempt";

#### Flow Detection

flow: to\_client,established;  
file\_data,  
content:"recordset"; offset:14; depth:9;  
content:".CacheSize"; distance:0; within: 100;  
pcr:"/CacheSize\s\*/";

#### Metadata References Classification Signature ID

policy max-detect-ips drop, service http;  
reference:cve,2016-8077;  
classtype: attempted-user;  
sid:65535;rev:1;

# Snort

## Azioni Base

---

- **Alert:** Genera un alert per il pacchetto analizzato (ovvero, si comporta come un NIDS), + log
- **Block:** Blocca il pacchetto analizzato e tutti i pacchetti seguenti dello stesso flusso, + log
- **Drop:** Blocca il pacchetto analizzato, + log
- **Log:** Logga il pacchetto analizzato
- **Pass:** Ignora il pacchetto analizzato (senza loggarlo!)

# Snort

## Active Responses

---

3 azioni aggiuntive, che non rientrano tra le "azioni base" già definite:

- **React:** Invia una risposta al client ed interrompi la sessione
- **Reject:** Interrompi la sessione con un TCP reset o un ICMP unreachable
- **Rewrite:** Sovrascrive i contenuti del pacchetto, sulla base dell'opzione "replace" definita insieme alla regola rewrite.

<https://docs.snort.org/rules/headers/actions>

# Snort

## Database delle Regole disponibile sul Web

---

È possibile visualizzare una descrizione di tutte le regole online. Alcuni esempi:

- [https://www.snort.org/rule\\_docs/1-1122](https://www.snort.org/rule_docs/1-1122)
  - Regola che controlla se l'utente sta provando a visualizzare /etc/passwd attraverso l'URL della pagina web
  - Bonus: Se proviamo a cercare una regola che contiene /etc/passwd sul sito di snort, veniamo bloccati dal WAF! :D
  - Black-box: il WAF si trova davanti l'IPS, oppure Snort non ha bloccato la nostra richiesta?!
    - Risposta verosimile: Snort si trova davanti il WAF, ma non ha bloccato la nostra richiesta perché non decripta TLS
- [https://www.snort.org/rule\\_docs/1-2063](https://www.snort.org/rule_docs/1-2063)
  - Regola che verifica se l'utente sta provando ad exploitare una vulnerabilità nota (CVE-2002-0539)
- È anche possibile scaricare il file contenente tutte le regole fatte dalla community.

# IPS o WAF?

---

Il fatto che la nostra richiesta sul sito di Snort sia stata bloccata non dall'IPS, ma dal WAF, potrebbe farci chiedere se le due soluzioni siano ridondanti o meno.

Infatti, le regole di Snort che abbiamo visto potrebbero essere implementate anche attraverso un WAF.

Ricordiamo però che il WAF opera solo al livello ISO/OSI 7 (applicativo), mentre **l'IPS opera anche al livello 4 (trasporto)**.

Inoltre, gli IPS tendono a non decriptare il traffico cifrato (e.g., TLS) per ottimizzare le prestazioni, per cui **questo compito può effettuarlo il WAF**.

# Suricata

---

<https://suricata.io/>

Network-based IPS (NIPS)

Rilasciato nel 2010, è diventato velocemente popolare quanto Snort

Anche Suricata è open-source, community veloce e reattiva

Circa 200 contributors



# Suricata

## Pros and Cons

---

- Multi-threaded
- Consuma più RAM di Snort, ma è comunque molto efficiente
- Anche in condizioni operative "stressanti" la packet/alert loss è ridotta al minimo
- Meno preciso di Snort, ma ha anche meno falsi positivi (vuol dire che ha più falsi negativi, però)
  - Accuracy:  $(TP + TN) / (TP + TN + FP + FN)$

[https://pure.port.ac.uk/ws/portalfiles/portal/79753845/A Comparative Analysis of Snort 3 and Suricata.pdf](https://pure.port.ac.uk/ws/portalfiles/portal/79753845/A_Comparative_Analysis_of_Snort_3_and_Suricata.pdf)

# Suricata

## Pros and Cons

---

- Suricata fa anche da NSM (Network Security Monitoring), per cui logga e interpreta tutto il traffico, che viene reso disponibile all'interno di un comodo file JSON che è facile interfacciare con altri programmi per la visualizzazione (e.g., Elastic, o anche Tulip se siete CTF players)

<https://docs.suricata.io/en/latest/output/eve/eve-json-format.html>

<https://www.stamus-networks.com/blog/suricata-myths-alerts-and-nsm>

- È possibile scrivere degli script in Lua per personalizzare le regole di detection

# Suricata

## Regole

---

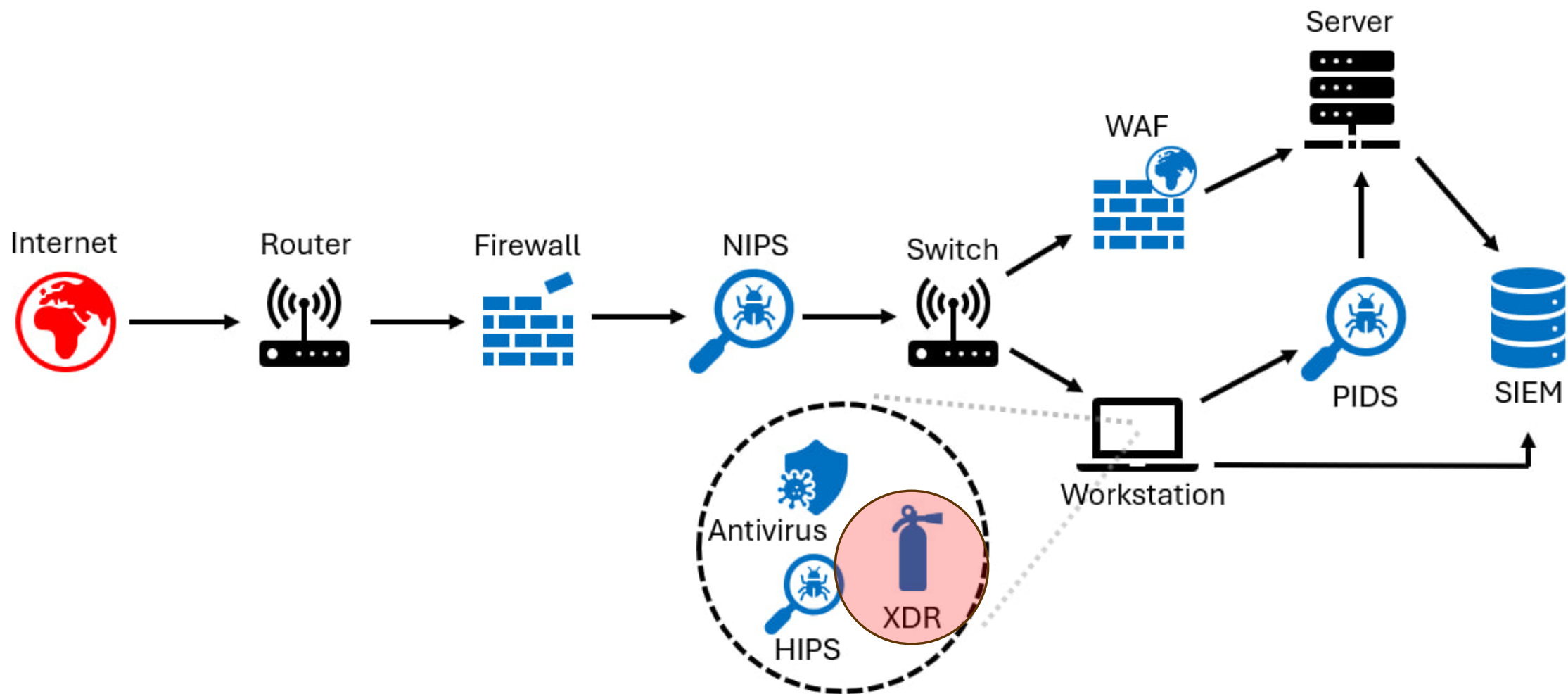
Esempio di regola in Suricata:

```
alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"HTTP GET  
Request Containing Rule in URI"; flow:established,to_server;  
http.method; content:"GET"; http.uri; content:"rule"; fast_pattern;  
classtype:bad-unknown; sid:123; rev:1;)
```

Vi ricorda qualcosa?!

C'è effettivamente qualche differenza con Snort (e.g., azioni):

<https://docs.suricata.io/en/latest/rules/intro.html>



# Endpoint Detection and Response (EDR)

---

Riconosciuto da Gartner nel 2013.

Software progettato per cercare di superare i limiti degli antivirus, e proteggere gli endpoint aziendali (laptop, smartphone, etc.) da varie minacce.

Questo viene fatto principalmente mediante la raccolta continua di dati sull'utilizzo degli endpoint, e l'analisi continua di tali dati, anche mediante tecniche di machine learning.

Molto utile, quindi, in quei casi in cui stabilire la firma di un attacco per il rilevamento da parte dell'antivirus è molto difficile o impossibile (e.g., social engineering che non prevede malware), per rilevare intrusioni o exploit 0-day.

# EDR

## Funzioni

---

Un EDR in genere dispone di cinque funzionalità:

1. Raccolta dei dati dagli endpoint
2. Analisi dei dati raccolti per il rilevamento delle minacce
3. Risposta automatizzata alla minacce rilevate
4. Isolamento e risoluzione delle minacce
5. Supporto per la ricerca manuale delle minacce

# EDR

## Raccolta Dati

---

Raccolta di tutti i dati di utilizzo del dispositivo, per vedere se ci sono scostamenti rispetto alla normalità che l'EDR ha osservato nel tempo. Esempi:

- Processi attivi (o anche inattivi - vedi [process hollowing](#))
- Creazione o modifica di file
- Siti web visitati
- Trasferimento di file
- Applicazioni utilizzate
- Comportamento dell'utente

# EDR

## Analisi per il Rilevamento di Minacce

---

Cerca di identificare minacce note o comportamenti sospetti attraverso algoritmi di machine learning, perlopiù.

Ricerca due tipi di informazioni:

- **Indicator of Compromise (IOC)**, ovvero eventi che possono essere correlati con una violazione in corso. Esempio: tentativi di connessione ad SSH in corso da un noto indirizzo IP malevolo.
- **Indicator of Attack (IOA)**, ovvero eventi associati a minacce o attori noti. Esempio: la modifica di 10.000 file nell'arco di pochi secondi, e il rename a **filename.crypt**.



# EDR

## Interfaccia con MITRE

---

Alcuni EDR si interfacciano con MITRE ATT&CK, una knowledge base di tecniche d'attacco, per ottenere informazioni su come operano i reali attaccanti.

Link: <https://attack.mitre.org/>

Esiste anche l'opposto, ovvero una knowledge base di tecniche di difesa, ovvero MITRE D3FEND.

Link: <https://d3fend.mitre.org/>

La ricerca nel settore della "malware attribution" si occupa di stabilire quali attacchi sono stati portati avanti da quali organizzazioni criminali.

Anche questo può aiutare l'EDR (e.g., anche i criminali riutilizzano parti di codice)

# EDR

## Risposta Automatizzata

---

Una volta identificata la minaccia, un EDR potrebbe, ad esempio:

- Sollevare uno o più alert
- Analizzare la catena di eventi sospetti, mapparli e determinare la causa scatenante
- Isolare un dispositivo, un account o un utente dalla rete
- Bloccare specifiche funzionalità sulla macchina infetta (e.g. la possibilità di creare file, connettersi ad Internet, leggere dispositivi rimovibili)
- Impedire l'esecuzione di codice malevolo (in maniera simile ad un antivirus)
- Attivare scansioni su altri endpoint non ancora violati per verificare la presenza della stessa minaccia o di minacce simili

# EDR

## Risoluzione Minacce e Supporto per la Ricerca

---

EDR può fornire supporto alle analisi forensi necessarie per individuare la causa scatenante dell'evento di sicurezza.

Lo scopo è ottenere informazioni su:

- File infetti
- Vulnerabilità utilizzate per accedere alla macchina (o per spostarsi verso altre macchine), e patch applicabili per ripararle, o modifiche necessarie alle regole di Access Control
- Credenziali violate
- Data e ora in cui il sistema esce dallo stato safe
- Come configurare i sistemi di protezione per evitare la minaccia in futuro.

# EDR

## Esempi

---

Ce ne sono molti a pagamento forniti dai soliti player nell'ambito della cybersecurity: Kaspersky, Checkpoint, Sophos...

Esiste qualche EDR free ed open source

Poco mantenuti, oppure piuttosto sospetti

Un esempio che appare affidabile è OpenEDR, ma non vengono rilasciati spesso aggiornamenti

<https://github.com/ComodoSecurity/openedr>

# eXtended Detection and Response (XDR)

---

Differisce da EDR in quanto non protegge solo gli endpoint dell'azienda, bensì tutti gli asset, da quelli on-premises a quelli in cloud.

Monitoraggio continuo di:

- Infrastruttura di rete
- Email
- Applicazioni
- Server (anche quelli in cloud)
- Altri sistemi di protezione, al fine di raccogliere più dati
- Dati di telemetria in generale
  - Telemetria: collezione di dati da diversi sistemi aziendali

# Protezione data dai Sistemi di Difesa

---

"Quindi, se utilizzo Firewall, IPS, Antivirus e XDR sono al sicuro, giusto?"

Purtroppo, la risposta a questa domanda è "no".

È importante comprendere che il Blue Team **reagisce** agli attacchi del Red Team, con pochissime eccezioni (ovvero, raramente ci si difende da un attacco che non è ancora stato ideato).

In pratica, il Blue Team agisce quasi sempre in modalità **best effort** perché nuove problematiche di sicurezza nascono ogni giorno (e.g., nuove vulnerabilità vengono scoperte, nuove campagne malware vengono attivate, nuovi tentativi di intrusione vengono effettuati).

Ne consegue che un **incidente di sicurezza** è **sempre possibile**.

# Incident Response

---

IBM (<https://www.ibm.com/topics/incident-response>) definisce l'Incident Response come:

"I processi e le tecnologie appartenenti ad un'organizzazione che hanno lo scopo di individuare e rispondere alle minacce informatiche, intrusioni o attacchi cibernetici."

Anche standard come ISO/IEC 27001 prescrivono la necessità di dotarsi di procedure al riguardo.

# Tipologie di Incidenti Comuni

## Ransomware

---

"Malware che blocca l'utilizzo del SO del computer vittima, chiedendo un pagamento in denaro (riscatto) per restituire le funzionalità bloccate."

I ransomware odierni criptano il contenuto dei dischi collegati alla macchina violata: pagando il riscatto, si ottiene la chiave per decriptarli (si spera).

Sempre più spesso, i gruppi criminali usano i ransomware anche per ottenere accesso **interattivo** ai sistemi violati.



# Tipologie di Incidenti Comuni

## Phishing

---

"Messaggi, in qualunque formato, che hanno lo scopo di manipolare il comportamento del destinatario e fargli eseguire azioni che lo danneggeranno."

Azioni tipiche richieste all'interno di messaggi di phishing:

- Scaricare ed aprire file (che potrebbero nascondere malware).
- Effettuare bonifici verso un determinato conto, spesso estero.
- Cliccare su link (che reindirizzano a domini malevoli, truffa simile al typosquatting).

# Tipologie di Incidenti Comuni

## Phishing

---

Spesso il mittente fa finta di essere un parente della vittima (e.g., il nipote) o una persona autorevole (e.g., forze dell'ordine, principale dell'azienda per cui la vittima lavora).

Quando il contenuto del messaggio di phishing è **attentamente personalizzato** dall'attaccante, per il seguente invio ad una vittima **ben specifica**, si parla di **spear-phishing**.

Il phishing non è altro che la più comune forma di Social Engineering.

# Tipologie di Incidenti Comuni

## Social Engineering

---

"Insieme di tecniche che hanno lo scopo di manipolare il comportamento delle persone e fare loro eseguire azioni che porteranno alla compromissione della sicurezza dei loro asset personali o aziendali."

- Phishing (già visto!)
- Baiting (e.g., truffa del principe Nigeriano, free music CD-ROM)
- Tailgating
  - Fisico: L'attaccante segue qualcuno all'interno di un'area ad accesso riservato
  - Digitale: Un dipendente va in pausa pranzo e dimentica di bloccare lo schermo, l'attaccante ne approfitta

# Tipologie di Incidenti Comuni

## Social Engineering

---

- Pretexting

- L'attaccante crea un finto pretesto e si offre per volerlo risolvere (e.g., classica pubblicità che sostiene che il pc di chi la visualizza sia infetto, e propone di scaricare un "antivirus")

- Quid pro quo (latino: "qualcosa al posto di qualcos'altro", "scambio")

- In italiano, "qui pro quo" viene usato anche come sinonimo di "equivoco"
- L'attaccante mostra alla vittima un premio, il cui ottenimento è vincolato ad una certa condizione (e.g., "Congratulazioni, sei il milionesimo visitatore! Installa il nostro software e vinci un coupon da 100€!")
- ~~Il coupon potrebbe non arrivare~~
- Simile al baiting, definizioni miste in letteratura: alcuni considerano il QPQ una sottocategoria del baiting, altri affermano che si tratta di baiting solo se la vittima è attratta dalla curiosità e non dal premio (in questo caso, il free music CD-ROM è baiting, il principe nigeriano è QPQ)

# Tipologie di Incidenti Comuni

## Social Engineering

---

- Scareware
  - Software (ad esempio un sito web: se si tratta di un semplice messaggio, ricade nel phishing) che usa la paura come metodo di manipolazione.
  - Ad esempio, la vittima viene reindirizzata su un dominio malevolo che mostra una pagina con il logo dei carabinieri. La pagina accusa l'utente di un crimine e chiede di installare un software per verificare l'innocenza dell'utente.
- Watering hole attack
  - L'attaccante riesce, mediante una vulnerabilità, ad iniettare del codice malevolo all'interno di un software, ad esempio un sito web, permettendogli così di rubare credenziali, scaricare ed eseguire ulteriore codice malevolo, etc.

<https://www.ibm.com/topics/social-engineering>

## Identification of cyber security breaches and attacks

Cyber security breaches and attacks remain a common threat.

Half of businesses (50%) and around a third of charities (32%) report having experienced some form of cyber security breach or attack in the last 12 months. This is much higher for medium businesses (70%), large businesses (74%) and high-income charities with £500,000 or more in annual income (66%).

By far the most common type of breach or attack is phishing (84% of businesses and 83% of charities). This is followed, to a much lesser extent, by others impersonating organisations in emails or online (35% of businesses and 37% of charities) and then viruses or other malware (17% of businesses and 14% of charities).

## Importanza di Social Engineering

---

Si potrebbe pensare che le precedenti slide diano troppa importanza al Social Engineering, e che si tratti perlopiù di tecniche che hanno effetto su una minoranza della popolazione che non è abituata all'utilizzo della tecnologia.

Tuttavia, i dati mostrano un quadro diverso.

Ad esempio, sulla sinistra un estratto dal Cyber Security Breaches Survey 2024 rilasciato ad Aprile dal governo UK.

# Possibili Impatti di un Incidente

---

- Perdite finanziarie
  - Esempio incidente: violazione di un account email aziendale
  - È più facile cadere vittima di phishing se l'email proviene effettivamente dall'indirizzo violato di un collega o del capo!
- Danni alla business continuity
  - Esempio incidente: diversi sistemi condividono la stessa vulnerabilità e vengono violati tutti; oppure l'attaccante, una volta dentro, ha trovato altre vulnerabilità ed ha effettuato movimento laterale
  - Diventa complicato identificare tutte le macchine violate e **pulirle tutte**
  - Anche se ci fosse una sola macchina violata, serve tantissima esperienza in forensics per rimuovere tutto il malware probabilmente droppato dall'attaccante
  - È più facile e sicuro ripristinare il sistema ad uno stato che è certamente **safe** (mediante backup o format)

# "Nuke It From Orbit"

---



*"I say we take off and nuke the entire site from orbit"*



*"..."*



*"It's the only way to be sure"*

Aliens - Scontro Finale (1986)



# Possibili Impatti di un Incidente

## Altri Esempi

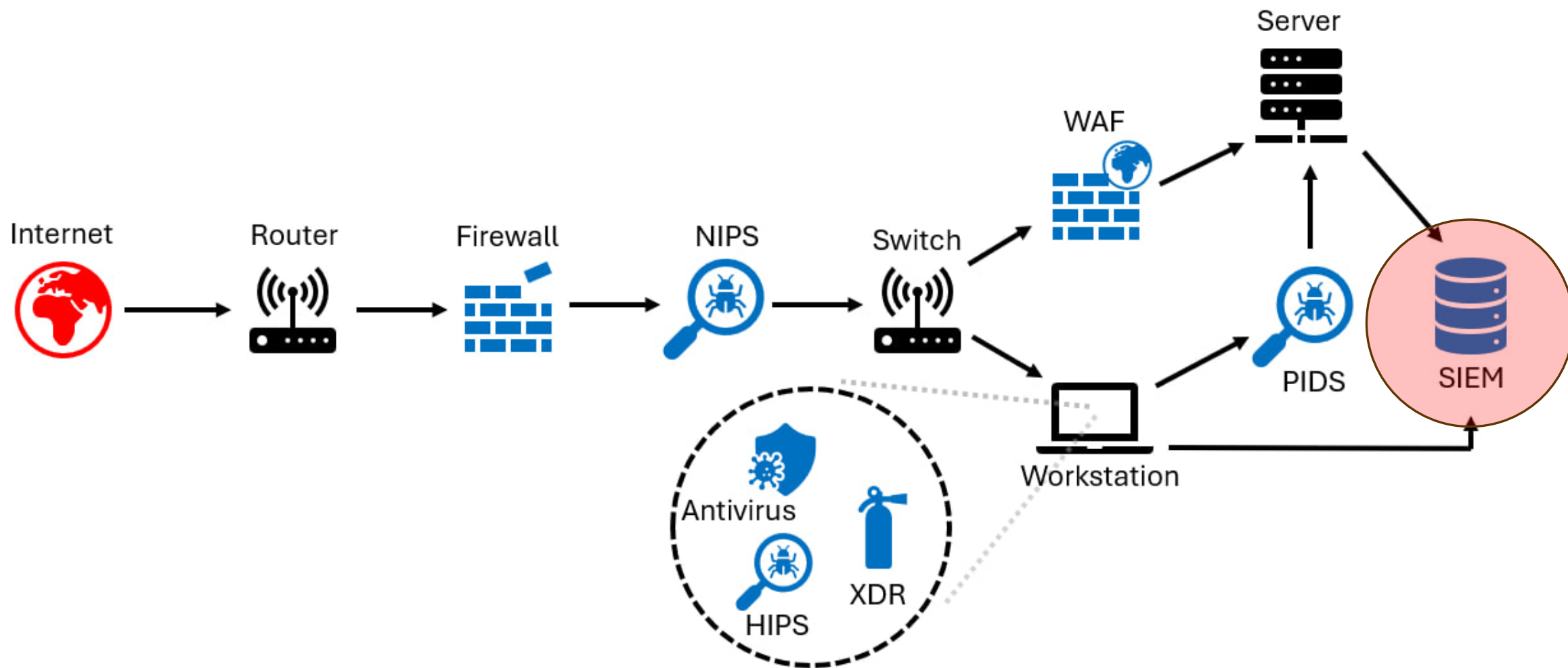
---

- Esfiltrazione dei dati

- Esempio incidente: dati sugli ultimi progetti aziendali (o peggio, dati personali, o documenti governativi segretati) vengono rubati e messi in vendita nelle darknet.
- I dati potrebbero anche diventare di dominio pubblico
- Bonus: danni di immagine, multe (ad esempio dal garante privacy), denunce, perdite finanziarie

- Perdita dei dati

- Esempio incidente: ransomware
- Raramente i ransomware odierni criptano e basta, quindi i sistemi sono comunque da ripristinare (nuke them from orbit!)
- Bonus: in Italia, per le aziende, pagare il riscatto **appare illegale!**  
<https://www.cyberlaws.it/2018/e-legale-pagare-riscatto-ransomware/>



# Log di Sistema

---

- Ogni azione effettuata da ogni utente all'interno di un sistema viene loggata.
- Spesso, il sistema operativo e le applicazioni dispongono dei loro log.
- A seguito di una violazione, tuttavia, questi log non sono da considerarsi affidabili!
  - L'attaccante potrebbe cancellare i log, oppure
  - L'attaccante potrebbe modificare i log per far ricadere la colpa su qualcun altro
- Serve quindi qualche strumento che possa garantirne l'integrità

# SIEM

---

## Security Information and Event Monitoring.

- Termine coniato nel 2005 da Gartner, unendo le definizioni di SIM (Security Information Monitoring) e SEM (Security Event Monitoring).
- Si trova solitamente su una macchina separata, e colleziona i log che gli vengono inviati dal resto delle macchine all'interno della rete.
- Gestisce i log, ne garantisce l'integrità e può correlare gli eventi loggati mediante regole predefinite, oppure impostate dai sysadmin.

# SIEM

## Requisiti

---

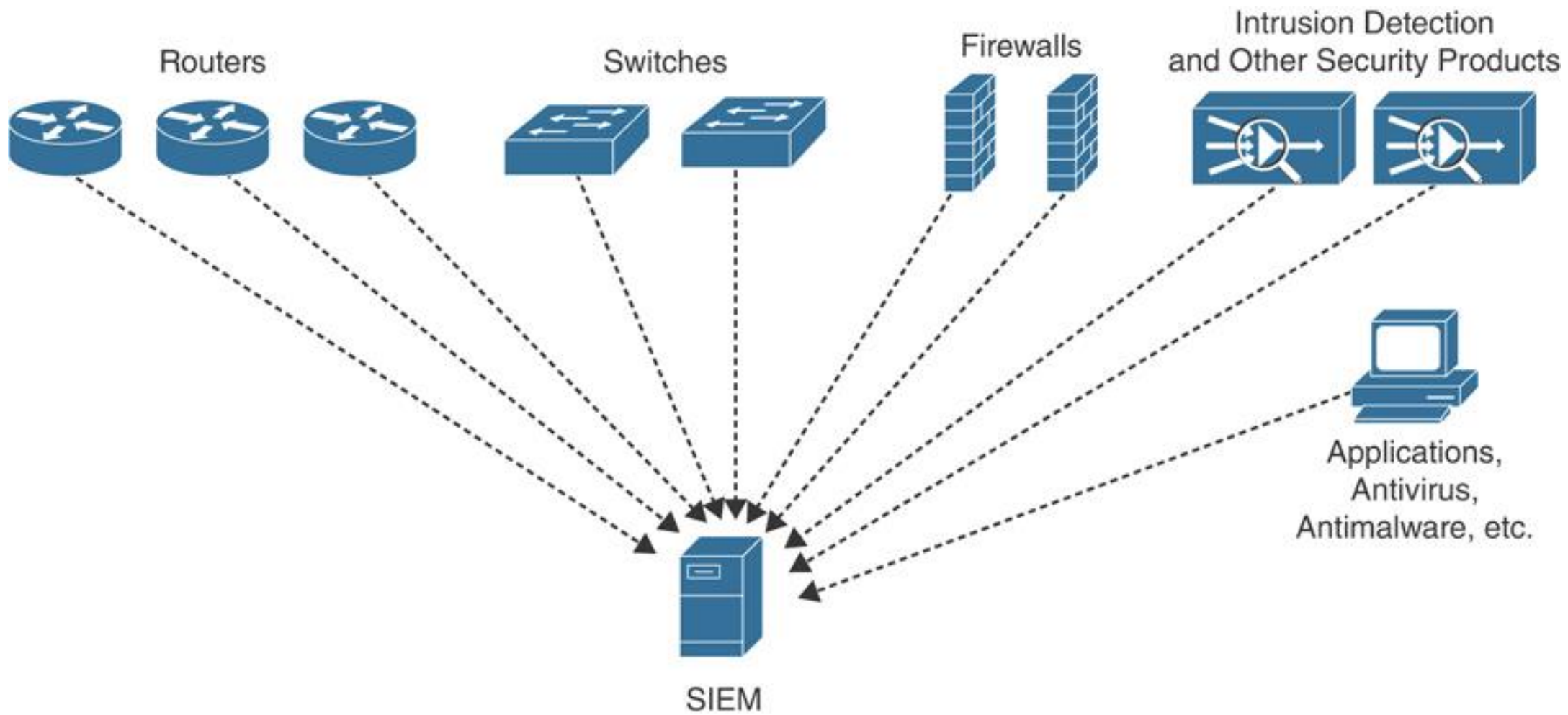
- Perché un SIEM funzioni correttamente, certi requisiti devono essere rispettati:
  - Il SIEM deve trovarsi in una macchina separata dal resto degli asset
  - I log di sistema devono essere inviati al SIEM non appena vengono generati, idealmente prima di venire scritti sul disco della macchina.
    - WinCollect, ad esempio, si interfaccia con IBM QRadar e raccoglie i log dalle API del sistema operativo delle macchine su cui è installato. Una volta raccolte delle informazioni, le invia subito a QRadar e **solo dopo** ne salva una copia su disco
  - I log dovrebbero essere criptati sia in transito che at rest
  - L'integrità dei file di log dovrebbe essere garantita **almeno** mediante hash crittografici, meglio se con **HMAC**.

# SIEM

## Integrità dei log

---

- Dato che i log e i relativi hash/MAC sono in doppia copia (una sul SIEM e una sulla macchina che ha generato il log), un attaccante dovrebbe violare sia il SIEM che la macchina che genera il log, al fine di poter nascondere le proprie tracce
- Visto che il SIEM si trova separato dal resto, e solitamente non è esposto sulla rete Internet, ne consegue che un attaccante dovrebbe, nell'ordine:
  1. Violare una macchina
  2. Esplorare la sottorete e notare la presenza del SIEM
  3. Violare il SIEM
  4. Procedere alla manomissione dei log
- Tra il punto 1 e 3 potrebbe passare diverso tempo, per cui il Blue Team dovrebbe avere la possibilità di accorgersi dell'intrusione prima della compromissione del SIEM



Fonte immagine: <https://www.pearsonitcertification.com/articles/article.aspx?p=3128871&seqNum=5>

Rule	Goal	Trigger	Event Sources
Repeat Attack-Login Source	Early warning for brute force attacks, password guessing, and misconfigured applications.	Alert on 3 or more failed logins in 1 minute from a single host.	Active Directory, Syslog (Unix Hosts, Switches, Routers, VPN), RADIUS, TACACS, Monitored Applications.
Repeat Attack-Firewall	Early warning for scans, worm propagation, etc.	Alert on 15 or more Firewall Drop/Reject/Deny Events from a single IP Address in one minute.	Firewalls, Routers and Switches.
Repeat Attack-Network Intrusion Prevention System	Early warning for scans, worm propagation, etc.	Alert on 7 or more IDS Alerts from a single IP Address in one minute	Network Intrusion Detection and Prevention Devices
Repeat Attack-Host Intrusion Prevention System	Find hosts that may be infected or compromised (exhibiting infection behaviors)	Alert on 3 or more events from a single IP Address in 10 minutes	Host Intrusion Prevention System Alerts
Virus Detection/Removal	Alert when a virus, spyware or other malware is detected on a host	Alert when a single host sees an identifiable piece of malware	Anti-Virus, HIPS, Network/System Behavioral Anomaly Detectors
Virus or Spyware Detected but Failed to Clean	Alert when >1 Hour has passed since malware was detected, on a source, with no corresponding virus successfully removed	Alert when a single host fails to auto-clean malware within 1 hour of detection	Firewall, NIPS, Anti-Virus, HIPS, Failed Login Events

Fonte tabella: [https://en.wikipedia.org/wiki/Security\\_information\\_and\\_event\\_management](https://en.wikipedia.org/wiki/Security_information_and_event_management)



# IBM QRadar

---

<https://www.ibm.com/it-it/qradar>

Suite commerciale di tool modulari sviluppata da IBM, che comprende:

- SIEM
- EDR
- "Log Insights", ovvero una piattaforma cloud per vedere e gestire dei log
- SOAR (Security Orchestration Automation and Response), soluzione simile a XDR che raccoglie eventi da vari endpoint, li correla e cerca di identificare e mitigare autonomamente le minacce. In generale è più complesso da gestire e configurare rispetto a XDR, e costa di più -- ma dipende dal SOAR utilizzato

QRadar Community Edition include solo SIEM e gestisce max 100 eventi/sec

# Wazuh

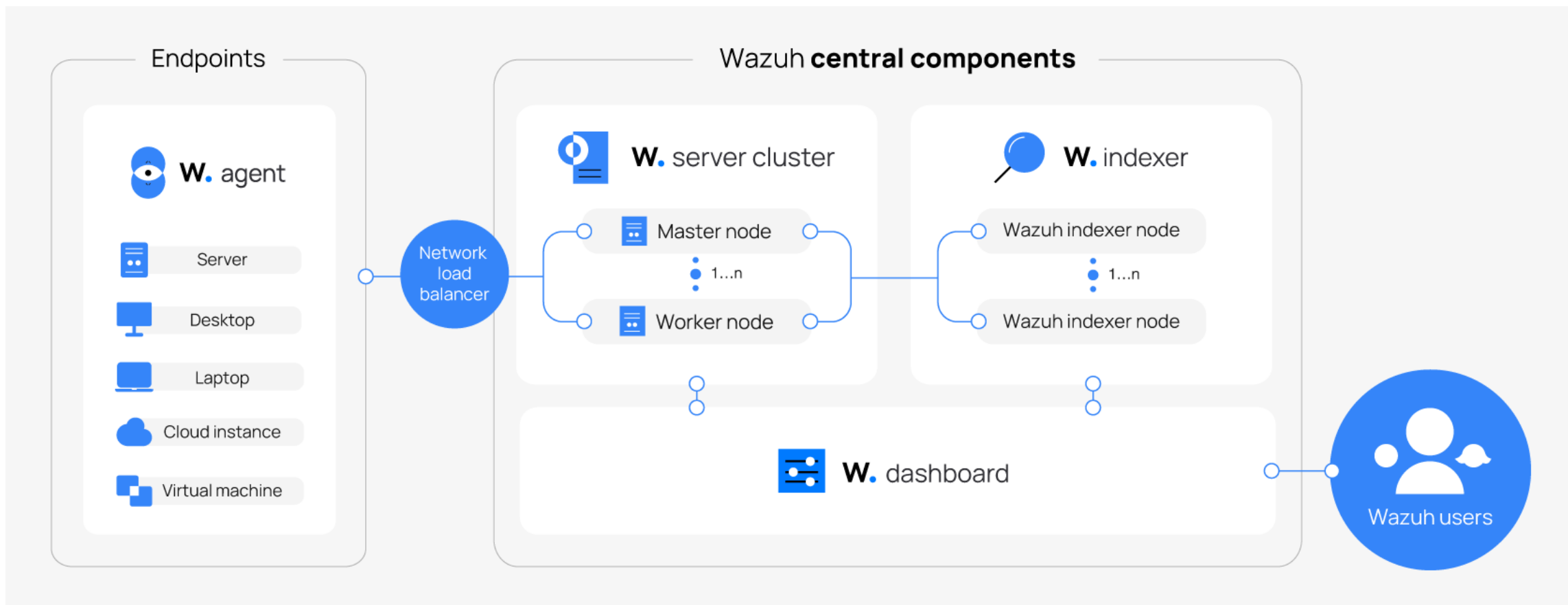
---

<https://wazuh.com/>

SIEM e XDR totalmente gratuito, rilasciato al pubblico nel 2017.

Dispone di quattro moduli:

- **Indexer:** indicizza e conserva gli alert, allo scopo di poterli recuperare mediante ricerche rapide da parte dell'utente
- **Server:** gestisce gli agent, ovvero li configura, li aggiorna e analizza i dati ricevuti per cercare indicatori di compromissione
- **Dashboard:** interfaccia utente per la visualizzazione dei dati
- **Agent:** componente installabile sull'endpoint da proteggere e monitorare



# Wazuh

## Regole

---

```
<rule id="5700" level="0" noalert="1">
  <decoded_as>sshd</decoded_as>
  <description>SSHD messages grouped.</description>
</rule>
```

```
<rule id="5716" level="5">
  <if_sid>5700</if_sid>
  <match>^Failed|^error: PAM: Authentication</match>
  <description>sshd: authentication failed.</description>
  <mitre>
    <id>T1110</id>
  </mitre>
  <group>authentication_failed,gdpr_IV_35.7.d,gdpr_IV_32.2,g
    ,tsc_CC7.3,</group>
</rule>
```

```
<rule id="5760" level="5">
  <if_sid>5700,5716</if_sid>
  <match>Failed password|Failed keyboard|authentication error</match>
  <description>sshd: authentication failed.</description>
  <mitre>
    <id>T1110.001</id>
    <id>T1021.004</id>
  </mitre>
  <group>authentication_failed,gdpr_IV_35.7.d,gdpr_IV_32.2,gpg13_7.1,hipaa
    ,tsc_CC7.3,</group>
</rule>
```

```
<rule id="5763" level="10" frequency="8" timeframe="120" ignore="60">
  <if_matched_sid>5760</if_matched_sid>
  <same_source_ip/>
  <description>sshd: brute force trying to get access to the system. Authentication failed.</de
  <mitre>
    <id>T1110</id>
  </mitre>
  <group>authentication_failures,gdpr_IV_35.7.d,gdpr_IV_32.2,hipaa_164.312.b,nist_800_53_SI.4,n
    .1,tsc_CC6.8,tsc_CC7.2,tsc_CC7.3,</group>
</rule>
```

# SOC

---

Dietro i tool ci sono sempre delle persone!

Il SOC (Security Operations Center) è il team che monitora 24/7 lo stato dell'infrastruttura mediante tutti i tool a disposizione, e reagisce se necessario.

Viene gestito dal CISO (Chief Information Security Officer), oppure da un ruolo separato chiamato SOC Manager.

Il team può essere interno all'azienda o essere gestito da consulenti.

SOC **distribuito** vs SOC **remoto** vs SOC **interno** (e.g., Control Room)

# SOC

## Compiti

---

Chi fa parte di un SOC non sta a guardare log per tutta la durata del suo turno (si spera!)

Oltre a gestire i vari tool presentati, chi fa parte di un SOC svolge anche le seguenti attività:

- Manutenzione dell'inventario degli asset e verifica della protezione di tutti gli asset mediante le tecnologie implementate
- Attività di routine come aggiornamenti di sistema, aggiornamento delle regole firewall/IPS/SIEM, verifica della corretta esecuzione dei backup e loro integrità

# SOC

## Compiti

---

- Gestione delle procedure di Incident Response: questo include anche test regolari mediante i cosiddetti cyber exercises
- Partecipazione a corsi di aggiornamento per stare al passo con le ultime soluzioni di difesa o gli ultimi attacchi recenti
- Utilizzo di altre soluzioni non descritte precedentemente (e.g., tool di Threat Intelligence)
- Talvolta si occupano anche della compliance, ovvero della verifica della conformità delle soluzioni di sicurezza alle normative vigenti (e.g., PCI DSS per la gestione dei dati di pagamento, GDPR per la privacy, NIS2 per le infrastrutture critiche, etc.)

# Threat Intelligence

## Greynoise

---

<https://www.greynoise.io/>

<https://viz.greynoise.io/>

<https://viz.greynoise.io/trends/trending>

Rete di honeypot diffusa per il web

Simula la presenza di servizi vulnerabili (Personae) allo scopo di raccogliere informazioni sugli attaccanti e sulle loro metodologie di attacco

Dati molto utili al SOC per decidere se alcuni alert borderline fanno parte di reali attacchi oppure si tratta di falsi positivi

Interrogazione del database gratuita!