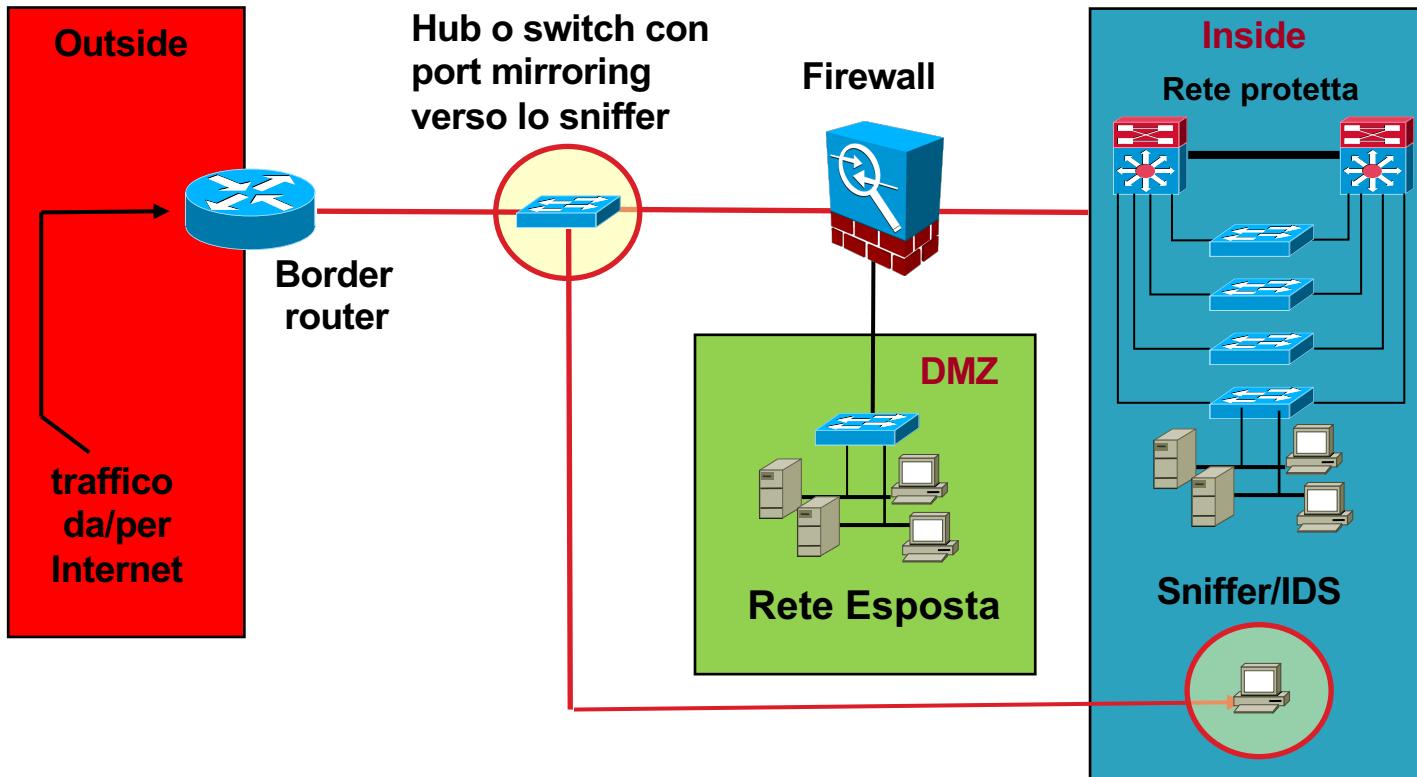


# Politiche di sicurezza e Controllo Accessi

*Francesco Palmieri*

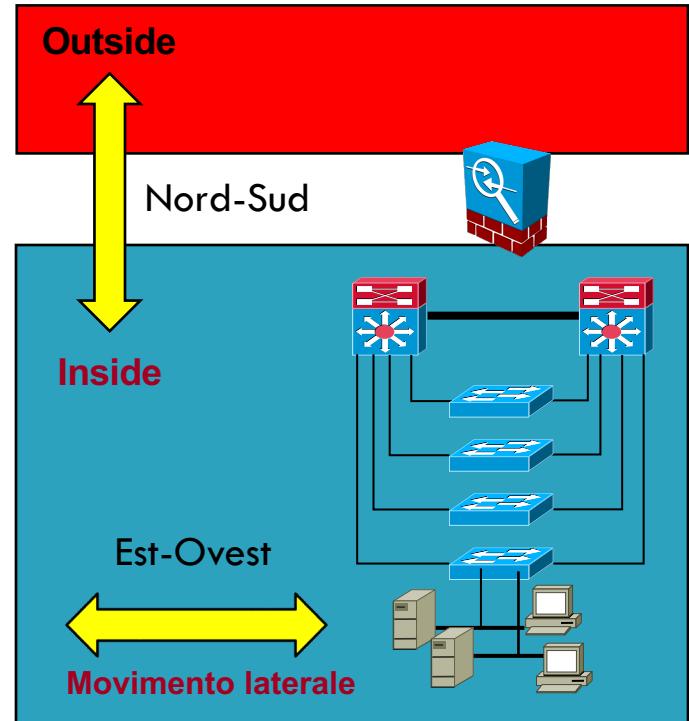
*fpalmieri@unisa.it*

# Architettura di riferimento



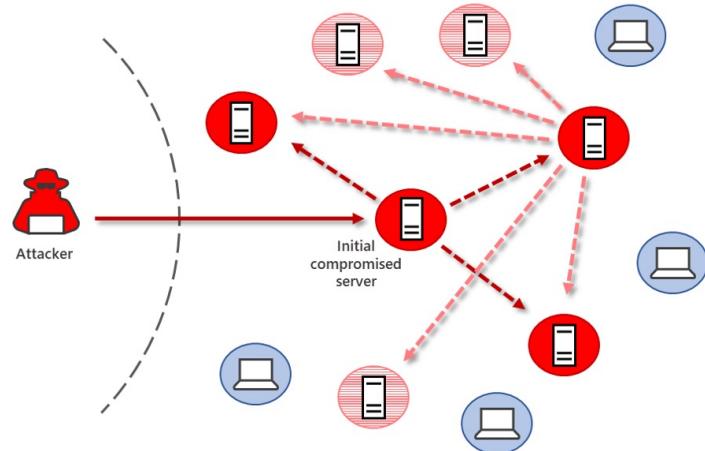
# Controllo del traffico

- Traffico **nord-sud**: fa riferimento a traffico che tipicamente attraversa il perimetro, e che quindi si muove fra diversi domini di sicurezza (es. Outside-Inside)
- Traffico **est-ovest**: si riferisce a traffico che si muove orizzontalmente all'interno dello stesso dominio di sicurezza (**movimento laterale**)



# Movimento Laterale

- Con il termine **Movimento Laterale** si fa riferimento a un'attività ostile che si muove all'interno di un dominio di sicurezza
- Una volta acquisito l'accesso a un dominio l'aggressore tipicamente usa la posizione come **base** da cui lanciare ulteriori attacchi.
- In assenza di punti di controllo e sezionamento l'aggressore può muoversi all'interno del dominio come se fosse un utente legittimo, senza che le misure di sicurezza convenzionali segnalino alcunchè.



Compromissione iniziale/infiltrazione →  
Movimento laterale →  
Nuovo attacco →

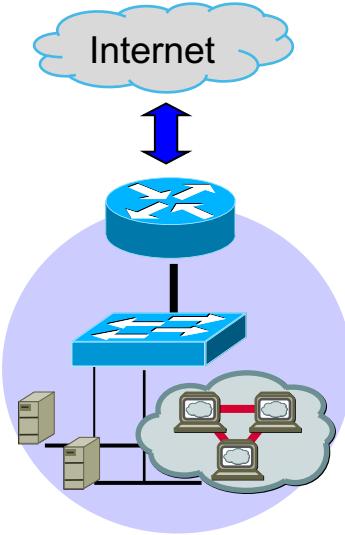
# Movimento Laterale: le fasi

Il movimento laterale si svolge in tre fasi principali:

- **Ricognizione**: esplorando il dominio, l'aggressore approfondisce la sua conoscenza della struttura e delle politiche di difesa dello stesso. L'individuazione di vulnerabilità gli consente o di formulare una strategia per addentrarsi nella rete.
- **Infiltrazione**: utilizzando le vulnerabilità individuate (credenziali di accesso, bugs, errori di protezione), l'aggressore impiega tecniche di exploitation ed escalation dei privilegi per ottenere l'accesso a altre componenti del dominio
- **Accesso**: una volta che individua le modalità di accesso, l'aggressore può iniziare il suo attacco vero e proprio, consegnando un payload malware, esfiltrando o distruggendo dati o eseguendo altre azioni indesiderate.

# Router di frontiera

- Il router di frontiera il primo sbarramento della propria rete
  - difficile l'aggiramento da parte dell' end-user.
- Permette di centralizzare un buon numero di controlli di sicurezza.
- Fondamentale la sua protezione
  - una compromissione può aprire l'accesso alla rete interna;
  - una inadeguata politica di filtraggio può esporre la stessa ad attacchi;
  - la corruzione delle tabelle di routing può provocare disservizi e accesso non autorizzato a dati.
- Un router correttamente configurato può minimizzare effetti derivanti da siti interni compromessi da attacchi.



# Firewall

- Firewall è un termine inglese dal significato originario di muro tagliafuoco
- E' il principale componente attivo di difesa perimetrale
- Ha compiti di security enforcing, nel senso più ampio del termine, con lo scopo di controllare il traffico fra due o più reti:
  - permettendo solo quello autorizzato dalla politica di sicurezza
  - rilevando e segnalando eventuali tentativi di violazione della politica di sicurezza
  - svolgendo eventualmente funzioni aggiuntive di auditing e accounting
  - Può anche svolgere funzioni di collegamento tra due o più segmenti di rete.



# Perché installare un firewall

- Per implementare una politica di sicurezza:
  - In grado di permettere l'accesso controllato ai sistemi o servizi di una rete protetta:
    - Solo agli utenti autorizzati
    - Solo ai sistemi autorizzati
  - In grado di permettere agli utenti e sistemi di una rete protetta di accedere in maniera controllata ai sistemi e servizi di una rete non protetta:
    - solo se il rischio è accettabile
    - registrando le attività



# Firewall: Vantaggi

- Centralizzazione delle politiche di sicurezza
  - Può tradursi in un Single point of failure (può essere uno svantaggio)
- Sistema specializzato in grado di ottimizzare le operazioni di filtraggio del traffico (tramite HW opportuno)
- Possibilità di ispezionare il traffico fino al livello di applicazione
- Controllo “stateful” delle sessioni



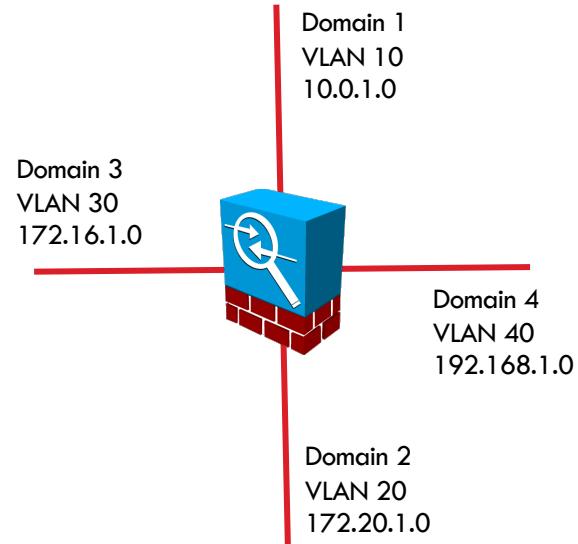
# Firewall: Svantaggi

- Difficoltà con protocolli non banali
- Prestazioni/throughput
  - Può trasformarsi in un bottleneck
  - la percezione dell'utente può essere negativa
- Gestione complessa
  - configurazione
  - Verifica e analisi dei log
- Senso eccessivo di fiducia e insicurezza interna
- Costi elevati in caso di prestazioni oltre il Gigabit



# Implementazione e funzioni di base

- Network device con almeno 2 interfacce di rete
- Ogni interfaccia individua un **dominio di sicurezza distinto** su un segmento (VLAN) diverso
- Può effettuare remapping indirizzi (NAT)
- Filtra traffico fra le diverse zone/domini tramite regole predefinite (politiche controllo accessi)
- Può mediare l'accesso a specifiche applicazioni a scopo di controllo ed ispezione
  - Proxy
  - Content filtering (filtraggio selettivo su base contenuti)
  - Deep packet Inspection e analisi del traffico
  - Limitazione in banda



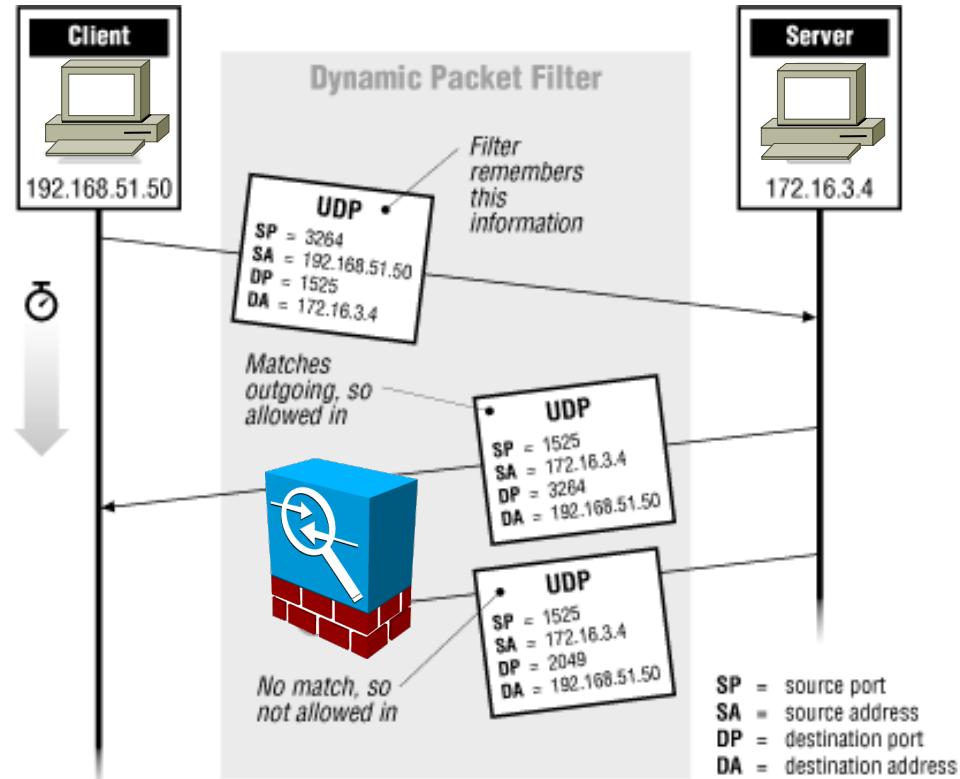
# Firewall hardware e Software

- **Firewall Hardware:**
  - componente passivo che opera una difesa perimetrale basandosi su specifici dispositivi di inspection e filtraggio implementati in ASIC
  - Rigidità operativa/Limitata flessibilità
- **Firewall Software :**
  - software che viene installato direttamente su HW COTS general purpose o in cloud in logica NFV
  - Prestazioni drasticamente inferiori ma funzioni dedicate
  - esegue anche un controllo a livello programma
  - Estremamente flessibile e configurabile



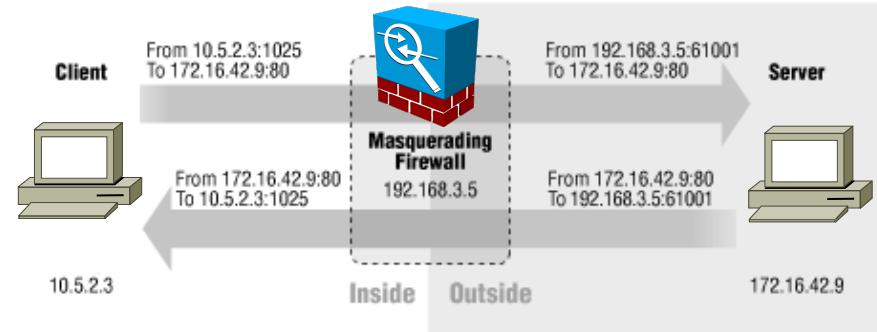
# Firewall: azioni possibili

- Accettare il pacchetto
- Scartare il pacchetto (non avvisa il mittente)
- Rifiutare il pacchetto (avvisa il mittente, es. ICMP port unreachable)
- NAT
- Log (remoto?)
- Filtri dinamici (controllati da IDS)
- Default deny/default permit



# NAT e sicurezza

- Il NAT non è un meccanismo di sicurezza (anche se c'è qualche vantaggio)
  - In presenza di overload effort aggiuntivo per identificare le reali origini
- Non c'è controllo del traffico sui singoli indirizzi
- Senza specifiche ACL si accede comunque agli indirizzi interni
- Il NAT può diventare una vulnerabilità esplorabile da insiders per un DoS:
  - Scansione Massiva per saturare la NAT table
  - Il garbage collection e l'aging delle entries saturano la CPU
  - Il blocco (per filtraggio) dell'indirizzo di NAT in overload blocca l'intera rete

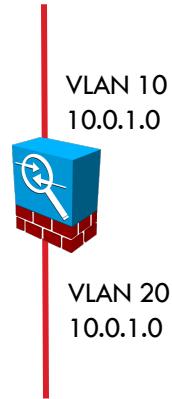


# Firewall: modalità operative

- Un firewall può operare in due modalità:
  - Routed: Opera a livello 3, segmenta reti diverse su base indirizzi IP
  - Trasparente: Opera a livello 2, segmenta su base MAC address
- Un firewall routed si presenta come un dispositivo di livello 3 e ha bisogno di un indirizzo/rete IP su ogni interfaccia
  - Instrada il traffico IP/IPv6 fra le varie interface
  - Supporta i più comuni protocolli di routing



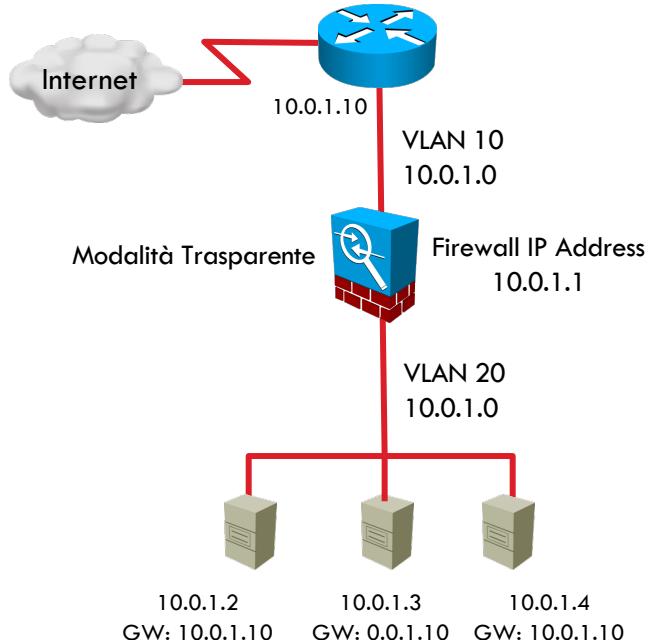
Modalità Routed  
o L3 firewall



Modalità Trasparente  
o L2 firewall

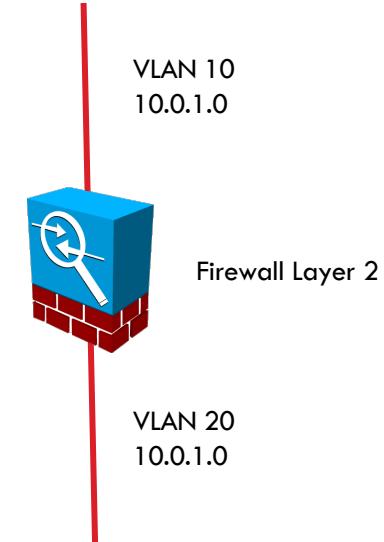
# Modalità Trasparente

- Il traffico Layer 3 deve essere esplicitamente permesso per attraversare il firewall
- I segmenti connessi alle interfacce devono essere sulla stessa subnet di livello 3
- L'IP del firewall non deve essere configurato come default gateway per i dispositivi connessi.
- I dispositivi devono puntare al router che sta avanti al firewall (attraversato trasparentemente)
- Ogni interfaccia individua comunque un segmento/VLAN differente anche se associate alla stessa rete IP



# Benefici di un Firewall Trasparente

- Flessibile, integrato e facile da gestire:
  - Reindirizzamento a livello IP non necessario
  - Nessun NAT da configurare
  - Non si possono verificare problemi di instradamento e undirizzamento
- Totalmente invisibile dall'esterno
- Maggiore robustezza



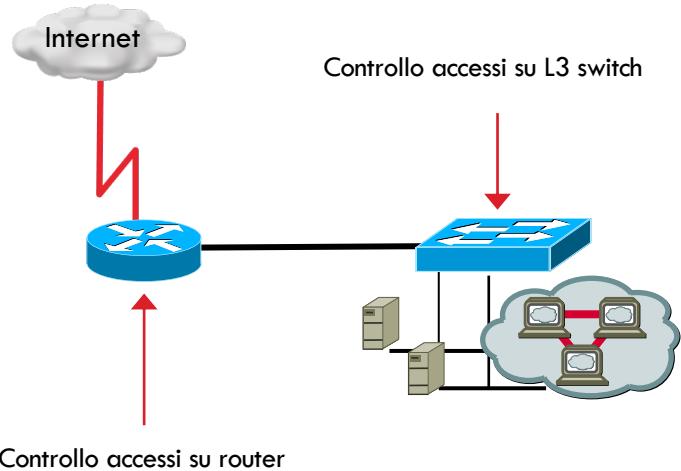
# Funzionalità non supportate

- Le seguenti funzionalità non sono tipicamente supportate da un firewall in modalità trasparente:
  - NAT
  - Protocolli di routing (es. OSPF, RIP, BGP)
  - IP/IPv6
  - DHCP relay
  - QoS
  - Multicast
  - Terminazione VPN



# Filtraggio sul bordo: controlli su border router

- Router e Switch Layer 3 rendono disponibili semplici meccanismi di controllo accessi
  - Stateless
  - IP address e TCP/UDP Port based
- L'uso di controlli complessi con un notevole numero di clausole di filtraggio comporta comunque un certo aggravio prestazionale a carico della CPU nell'attività di forwarding
- Accettabile se Router o Switch L3 gestiscono meccanismi di controllo accessi in Hardware
- Dispositivi già presenti in ogni rete che la partizionano in maniera naturale



# Filtraggio su border router: Pro e Contro

Meccanismo principale di protezione:

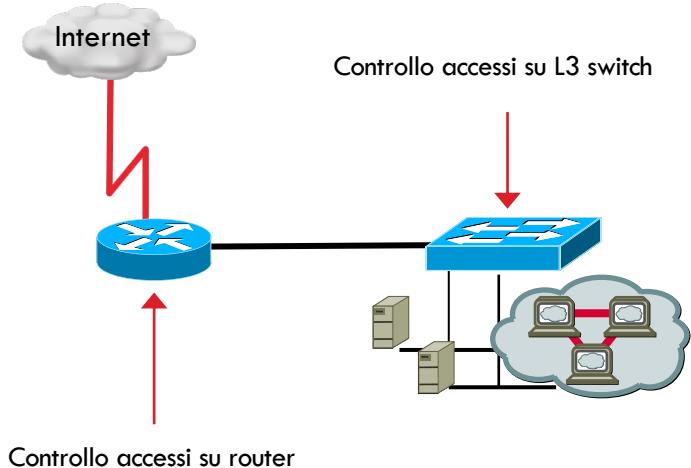
- ACL statiche

## Pro

- Buone prestazioni
- Trasparenza
- Basso costo (eventualmente nullo)

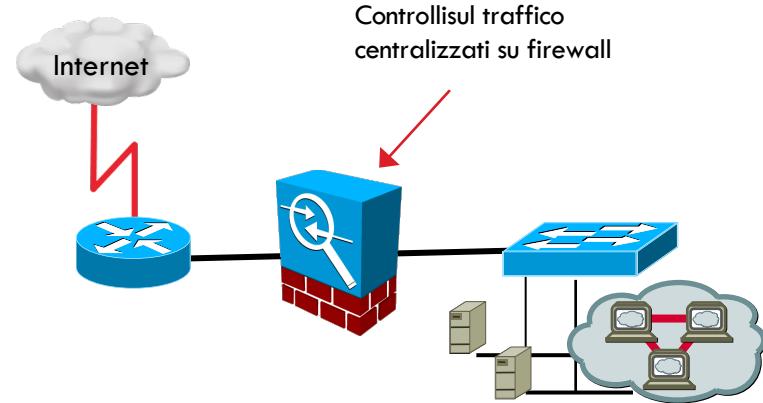
## Contro

- Molti protocolli non sono gestibili
- Nessun controllo sui dati (payload) e lo stato delle sessioni



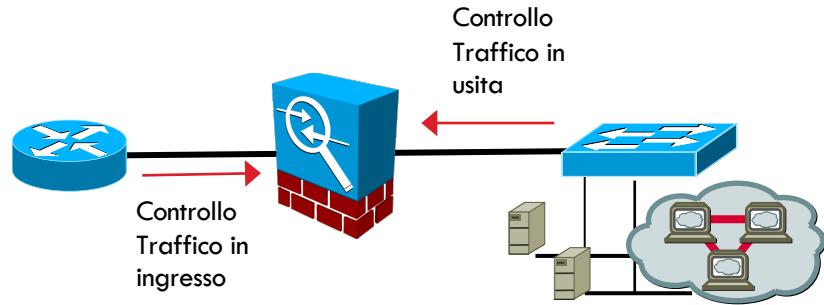
# Filtraggio sul bordo: controlli su firewall

- L'introduzione di un firewall scarica la CPU di router o switch L3 dalla valutazione dei controlli
- La centralizzazione delle politiche di controllo sul firewall costituisce un significativo vantaggio gestionale:
  - riduce la complessità della configurazione
  - centralizza la gestione delle logiche e problematiche di filtraggio
  - Permette di proteggere simultaneamente migliaia di macchine
- Questa politica non scala in presenza di grandi volumi di traffico e diventa un collo di bottiglia prestazionale che può essere sfruttato per creare DoS



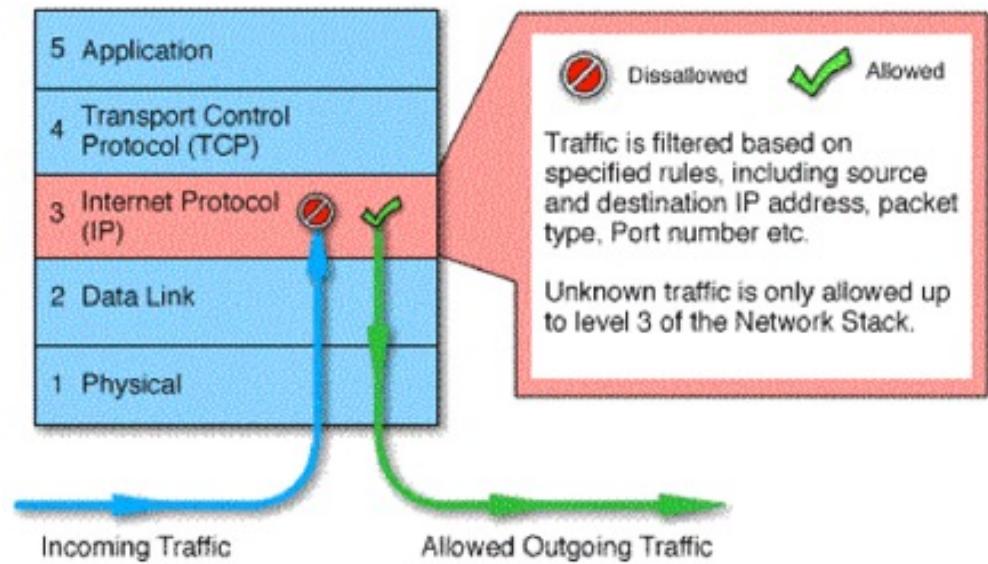
# Dove effettuare il filtraggio

- In ingresso
  - so da quale interfaccia arriva il pacchetto
  - proteggo la rete locale
- In uscita
  - controllo anche il traffico generato localmente
  - blocco ciò che non deve uscire
- I controlli vanno localizzati **più vicino possibile all'origine del traffico**



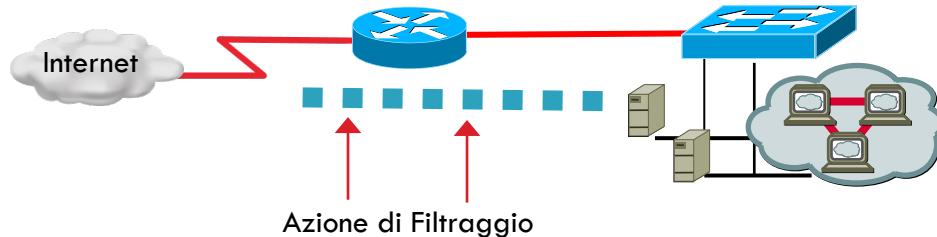
# Parametri di filtraggio

- Header IP
  - mittente
  - destinatario
  - protocollo
  - flags, opzioni (source routing, frammentazione...)
- Header TCP/UDP
  - porta mittente
  - porta destinataria
  - flags TCP (SYN, ACK)



# Filtraggio stateless

- Solo su base IP (sorgente e destinazione), porta TCP/UDP e protocollo
- Controllo effettuato indipendentemente pacchetto per pacchetto
- Non ha alcuna percezione del flusso dei pacchetti che fanno parte di una connessione end to end
- I pacchetti possono provenire anche da interfacce diverse da quella su cui escono (tollerata l'asimmetria)

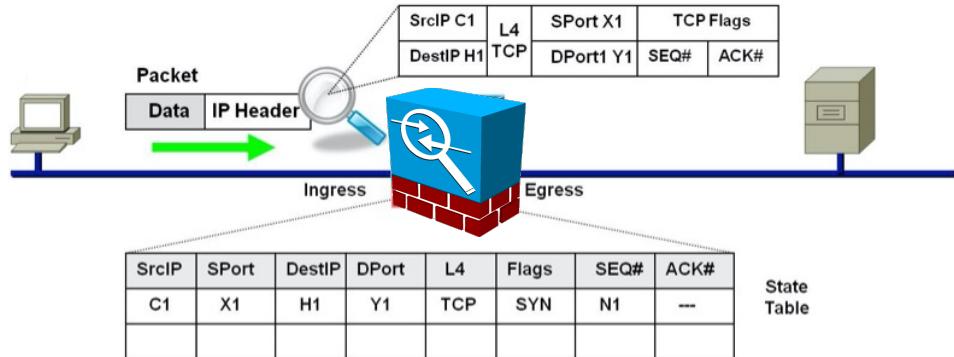


# Limitazioni del Filtraggio Stateless

- Nelle connessioni TCP, le porte con numeri inferiori a 1024 sono assegnate permanentemente a servizi “noti” erogati da specifiche applicazioni server
  - 20, 21 per ftp, 23 per telnet, 25 per smtp, 80 per http...
- I clients che si connettono a tali server usano porte mumerate da 1024 a 16383
  - Tali porte vengono quindi usati quali destinazioni per il recapito del traffici di ritorno
- Cosa dovrebbe fare un firewall se vede, ad esempio, una richiesta in arrivo alla porta 1234 di alcuni client?
  - Deve farlo passare dato che potrebbe essere la risposta di un server in una connessione precedentemente stabilita ...
  - ... Oppure potrebbe essere traffico dannoso ...
- **Non è possibile saperlo** senza mantenere lo stato per ogni connessione

# Filtraggio Stateful

- quando viene stabilita una connessione, se le regole di filtraggio non la bloccano, allora le informazioni relative ad essa diventano entry di una tabella di stato.
  - successivi pacchetti in ingresso saranno valutati in base all'appartenenza ad una delle connessioni presenti nella tabella (session filtering).
  - quando la connessione è conclusa, la entry nella tabella sarà cancellata, per evitare che questa si riempia completamente informazioni riguardanti la connessione che verranno memorizzate :
    - identificatore univoco sessione
    - stato connessione (*handshaking, established, closing*)
    - informazioni sequenzialità pacchetti
    - Indirizzi IP host sorgente e destinazione
    - interfacce di rete utilizzate



# Session Filtering

- Ogni decisione (**permit** o **deny**) viene presa separatamente per ciascun pacchetto, ma nel contesto di una connessione
  - Se è relativo a una nuova connessione, ne va verificata l'ammissibilità sulla base delle politiche di sicurezza configurate
  - Se viceversa è parte di una connessione esistente, la stessa va cercata nella tabella delle connessioni, aggiornandone lo stato, se necessario
    - A questo punto è possibile consentire il traffico in entrata verso una porta con numero elevato solo se è stata stabilita una connessione a quella porta
- I protocolli stateless (UDP) e ICMP sono difficili da filtrare
  - La politica di filtraggio di base è negare tutto ciò che non è consentito
  - Ci vuole attenzione nel filtrare il traffico di protocolli servizio come ICMP
- I filtri possono essere bypassati utilizzando meccanismi di tunneling

# Esempio: tabella di stato delle connessioni

Source Address	Source Port	Destination Address	Destination Port	Connection State
192.168.1.100	1030	210.9.88.29	80	Established
192.168.1.102	1031	216.32.42.123	80	Established
192.168.1.101	1033	173.66.32.122	25	Established
192.168.1.106	1035	177.231.32.12	79	Established
223.43.21.231	1990	192.168.1.6	80	Established
219.22.123.32	2112	192.168.1.6	80	Established
210.99.212.18	3321	192.168.1.6	80	Established
24.102.32.23	1025	192.168.1.6	80	Established
223.212.212	1046	192.168.1.6	80	Established

# Limitazioni del Packet Filtering

- Non è possibile bloccare attacchi application-specific
  - Per esempio, in presenza di un buffer overflow in una routine di URL decoding il firewall non sarà in grado di bloccare uno specific pattern di attacco
- Non sono previsti meccanismi di autenticazione a livello utente
  - ... ad eccezione di una banale autenticazione basata sull'indirizzo IP (spoofabile)
  - manca qualsiasi funzionalità di livello più alto del livello 3 (e in parte del livello 4)
- Il meccanismo è vulnerabile a semplici attacchi layer 3 (ad es. spoofing)
  - L'unica soluzione è l'enforcing di specifiche regole anti-spoofing (vedremo più avanti)
- Una configurazione errata può facilmente portare a compromissione completa della sicurezza di un dominio protetto via packet filtering

# Politiche di filtraggio

- Alla base di ogni politica di filtraggio finalizzata al controllo degli accessi va fatta un' attenta valutazione preliminare:
  - Chi ha bisogno di accedere?
  - Quando e come?
  - Da dove?
  - Con quale decorrenza?
  - Di quali servizi ha bisogno?
  - Che protocolli usa?
  - Che QoS richiede?
- Si parla in questo casi di controlli “**who, what, when, where, why and how**”



# Politiche di filtraggio

Un firewall (o un router) può operare in due modalità diametralmente opposte :

- *Deny All*: Tutto ciò che non è specificatamente permesso è negato. Elevata Sicurezza
- *Allow All*: Tutto ciò che non è specificatamente negato è permesso Facilità di gestione

La seconda policy si usa poco in ambiti di sicurezza, tuttavia è molto utilizzata per troubleshooting e traffic shaping.

# Deny All

*Tutto ciò che non è specificatamente permesso è negato*

- Blocca tutto il traffico e ciascun servizio deve essere implementato caso per caso
- Politica maggiormente conservativa in termini di protezione
- si limita il numero di scelte disponibili all'utente



# Allow all

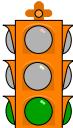
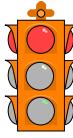
*Tutto ciò che non è specificatamente negato è permesso*

- Inoltra tutto il traffico e ciascun servizio dannoso deve essere chiuso caso per caso
- L'amministratore di rete ha difficoltà sempre maggiore nel garantire la sicurezza man mano che la rete cresce.



# Filtraggio selettivo

Service	Port	Protocol
echo	7	TCP/UDP
discard	9	TCP/UDP
systat	11	TCP/UDP
daytime	13	TCP/UDP
netstat	15	TCP
quotd	17	TCP/UDP
chargen	19	TCP/UDP
ftp-data	20	TCP
ftp	21	TCP
ssh	22	TCP/UDP
telnet	23	TCP
smtp	25	TCP
time	37	TCP/UDP
rlp	39	TCP/UDP
whois	43	TCP/UDP
tacacs	49	TCP/UDP
domain	53	TCP
whois++	63	TCP/UDP
bootp	67-68	UDP
tftp	69	UDP
gopher	70	TCP
finger	79	TCP
http	80	TCP
link	87	TCP
supdup	95	TCP
pop2	109	TCP
pop3	110	TCP
sunrpc	111	TCP/UDP
auth	113	TCP/UDP
nntp	119	TCP
ntp	123	TCP/UDP
nbios-ns	137	TCP/UDP

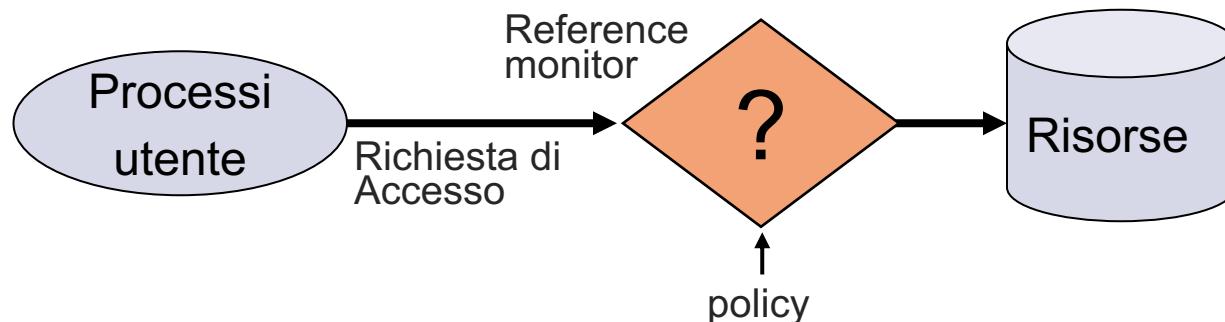


Service	Port	Protocol
nbios-dgm	138	TCP/UDP
nbios-ssn	139	TCP/UDP
imap	143	TCP
NeWS	144	TCP
snmp	161	UDP
snmptrap	162	UDP
xdmcp	177	UDP
irc	194	TCP/UDP
waitsZ39.50	210	TCP
imap3	220	TCP
ldap	389	TCP/UDP
netware-ip	396	TCP/UDP
rmt	411	TCP
https	443	TCP
exec	512	TCP
biff	512	UDP
login	513	TCP
who	513	UDP
shell	514	TCP
syslog	514	UDP
printer	515	TCP/UDP
talk/ntalk	517-518	TCP/UDP
route	520	UDP
timed	525	TCP/UDP
uucp	540-541	TCP
mountd	635	TCP/UDP
wins	1512	TCP/UDP
radius-old	1645-1646	UDP
radius	1812-1813	UDP
openwin	2000	TCP
NFS	2049	TCP/UDP
X11	6000-6063	TCP

- E' consigliabile bloccare oppure filtrare selettivamente i servizi tendenzialmente pericolosi
- Permetti in ingresso solo l'accesso ad un numero estremamente limitato di servizi esposti (e-mail, www, ftp) erogati solo da hosts specifici e eventualmente controllati

# Controllo Accessi a livello utente

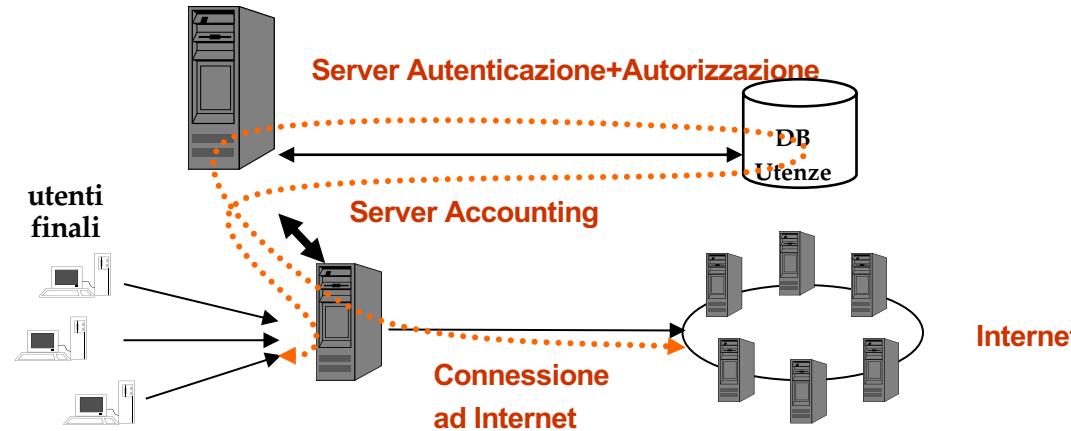
- Assunzioni
  - Il sistema deve conoscere chi è l'utente interessato (identificazione)
    - Autenticazione via username e password, o altre credenziali
  - Le richieste di accesso passano attraverso un “reference monitor” che è deputato a implementare le policy
    - Non è consentito bypassare in alcun modo il reference monitor



# Le Architetture AAA

Autentication, Authorization, Accounting: sono architetture che si basano sulla separazione delle fasi di controllo accessi in:

- Authentication
  - Identificare l'utente in modo certo. (**CHI**)
- Authorization
  - Definire a quali risorse l'utente ha accesso. (**DOVE**)
- Accounting
  - Tenere traccia delle operazioni effettuate. (**QUANDO**)



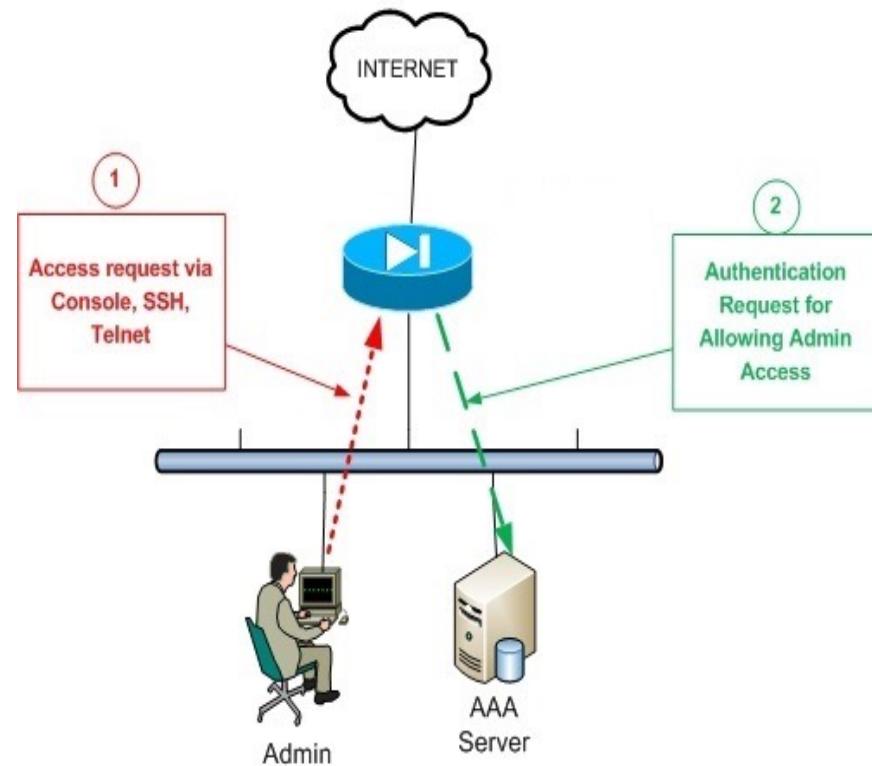
# Autenticazione

- Occorre che l'utente dichiari la propria identità. Esempi:
  - username e password
  - Carta + PIN
- Scoraggiare gli scambi di identità
- Imporre il cambio periodico della password
- Imporre password difficili da indovinare



# Autorizzazione

- Dopo aver identificato l'utente, questi non deve necessariamente aver libertà di movimento in ogni campo; **avrà accesso ad alcune risorse**, ma non ad altre.
- Es. un certo dipendente potrà accedere alla posta elettronica, navigare su Internet, ma non consultare il bilancio o aprire il database paghe.

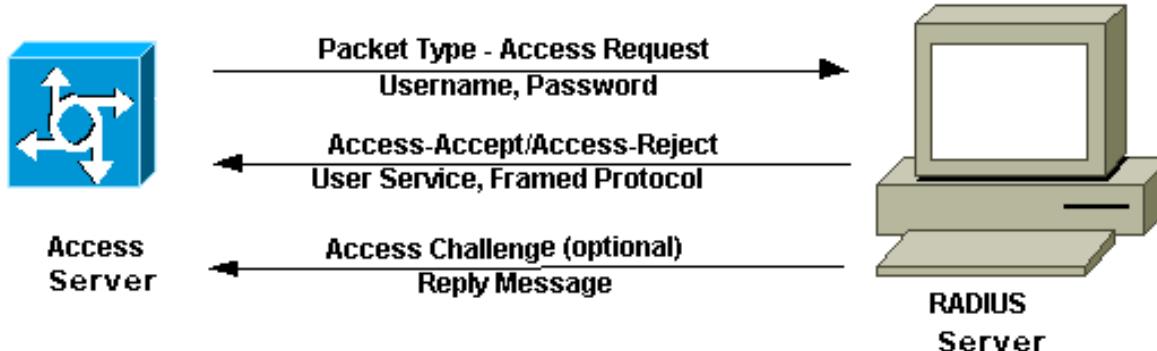


# Accounting

- Conoscere le operazioni fatte da un utente non ha necessariamente scopi punitivi.
- Si raccolgono dati sull'uso delle risorse da parte dei vari settori dell'azienda e può servire a pianificare o rimandare aggiornamenti dei sistemi e aiuta ad indirizzare gli investimenti.
- Logging di specifiche operazioni
- Billing in modelli pay per use

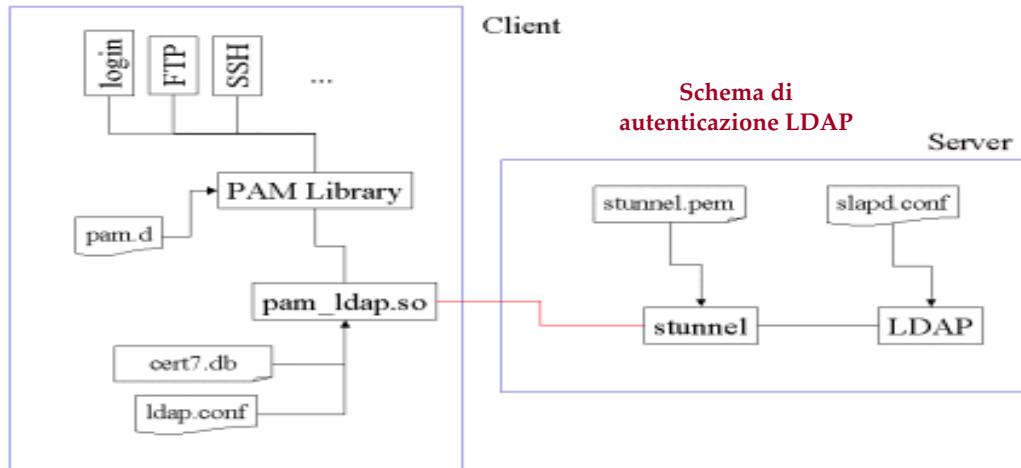
# RADIUS/TACACS+

- Spesso all'estremità di questo processo di convalida è presente un server che supporta **Remote Authentication Dial-In User Service, (RADIUS)** o **TACACS+** che sono protocolli di consolidamento e centralizzazione delle informazioni di convalida, autorizzazione e identificazione degli utenti remoti.
- In breve, RADIUS crea una posizione unica in cui tutti gli utenti remoti vengono analizzati e il server accetta o rifiuta un utente in base a criteri definiti.



# RADIUS & LDAP

Il futuro di RADIUS è in realtà allineato con gli sviluppi sul fronte dei servizi di directory, che si stanno indirizzando verso **Lightweight Directory Access Protocol (LDAP)**, uno standard per la creazione di directory di rete interoperabili.



# Il Modello di Access Control

- Il concetto di protezione e controllo è rappresentato da una tripla:

$$M = (S, O, A)$$

Dove:

- $S$  = insieme di soggetti (entità attive); es. user, indirizzi IP, processi, dispositivi etc...;
- $O$  = insieme di oggetti (entità passive); es. file, interfacce da attraversare, memoria etc...;
- $A$  = insieme di regole che specificano i modi in cui i soggetti possono avere accesso agli oggetti

**Nota: ogni soggetto può anche essere un oggetto**

# Access control matrix [Lampson]

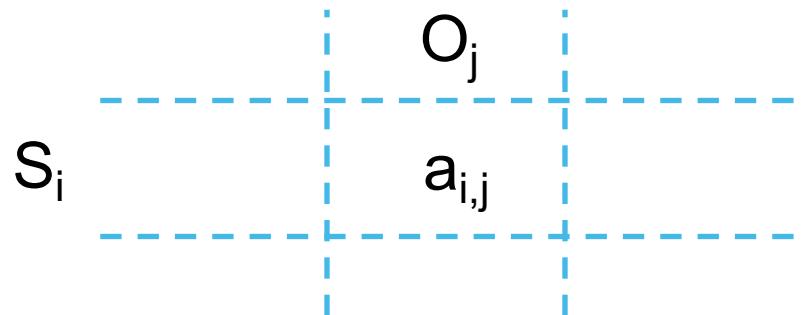
	Oggetti				
	File 1	File 2	File 3	...	File n
User 1	read	write	-	-	read
User 2	write	write	write	-	-
User 3	-	-	-	read	read
...					
User m	read	write	read	write	read

## Il Modello (2)

- Nella matrice sono catturate tutte le relazioni tra le entità:

$$R \rightarrow A[s,o] \quad \text{dove} \quad s \in S, \quad o \in O$$

- $A[s,o] \subseteq R$ ;
- $A[s,o] \rightarrow$  insieme di privilegi/diritti/azioni di  $s$  su  $o$ .



# Il Modello (3)

- Quando viene creato un oggetto:
  - si aggiunge una colonna nella matrice di accesso;
  - il contenuto della nuova colonna è deciso al momento della creazione dell'oggetto;
- E' un **modello astratto**: il significato dei diritti può cambiare in base agli oggetti coinvolti
- Non è molto appropriata per l'implementazione diretta
- La matrice è sparsa e gli elementi possono essere vuoti; in questo caso può essere inefficiente

# Implementazione

- **Access control list (ACL)**
  - Implementano colonne della matrice (associate alle risorse)
- **Capability Lists**
  - L'utente ottiene un “ticket” o **diritto** di accesso per ogni risorsa
  - Due varianti
    - Associano righe della matrice all'utente sotto il controllo dell'OS
    - Implementano ticket non falsificabili nello spazio utente
- Nella maggior parte dei casi reali I controlli vengono realizzati attraverso Access control lists
- Diversi sistemi implementano solo alcuni aspetti delle capability lists

	File 1	File 2	...
User 1	read	write	-
User 2	write	write	-
User 3	-	-	read
...			
User m	Read	write	write

# ACL vs Capability Lists

- Access control list
  - Associano una lista con ogni oggetto
  - Controllo a livello utente/gruppo con la lista stessa
  - Basate sull'autenticazione: bisogna identificare l'utente
- Capabilities
  - Una Capability è un **ticket** (diritto) non falsificabile
    - Devono essere gestite a livello OS
    - Possono essere passate da un utente o processo a un altro
  - Il **Reference monitor** controlla i ticket
    - Non a bisogno di conoscere l'identità di un utente o processo

# ACL e Capabilities

	F1	F2	F3	F4	F5	F6
S1		O, R, W	O, R, W		W	
S2	O, R, W	R			O, R, W	
S3		R	R	O, R, W	R	O, R, W

## CAPABILITY

$S1 = \{(f2, orw); (f3, orw); (f5, w)\}$

$S2 = \{(f1, orw); (f3, r); (f5, orw)\}$

$S3 = \{(f2, r); (f3, r); (f4, orw); (f5, r); (f6, orw)\}$

## ACLs

$F1 = \{(s2, orw)\}$

$F2 = \{(s1, orw); (s2, r); (s3, r)\}$

$F3 = \{(s1, orw); (s3, r)\}$

$F4 = \{(s3, orw)\}$

$F5 = \{(s1, w); (s2, orw); (s3, r)\}$

$F6 = \{(s3, orw)\}$

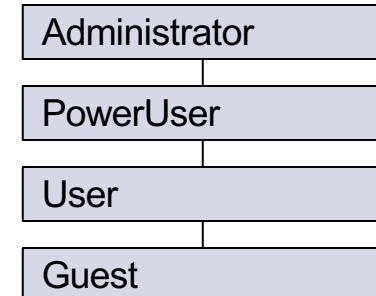
# ACL vs Capabilities

- Delega
  - **Capability**: Un Processo può trasferire una capability al run time
  - **ACL**: Il proprietario di un oggetto può aggiungere permessi alla lista
- Revoca
  - **ACL**: Eliminazione di un utente o gruppo dalla lista
  - **Capability**: Richiesta di riottenere (togliendola) una capability precedentemente concessa
    - Possibile in sistemi che implementano appropriati meccanismi di bookkeeping
    - Indirezione: una capability punta a un puntatore alla risorsa R interessata

If  $C \rightarrow P \rightarrow R$ , allora revoca la capability C ponendo  $P=0$

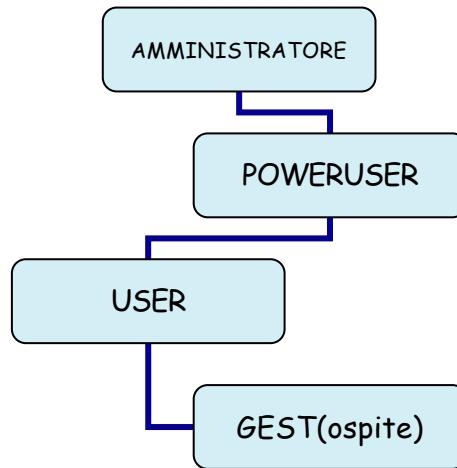
# Ruoli (o Gruppi)

- **Ruolo:** insieme di utenti con specifiche attribuzioni
  - Sono il livello intermedio tra soggetti e oggetti:
  - Administrator, PowerUser, User, Guest
  - Assegnando permessi a specifici ruoli ogni utente ottiene il permesso
- **Strutturati in gerarchia** fra ruoli:
  - Ordinamento fra ruoli annidati
  - Ogni ruolo ha i permessi dei ruoli sottostanti
  - Vanno specificati solo i nuovi permessi associati a ruoli superiori

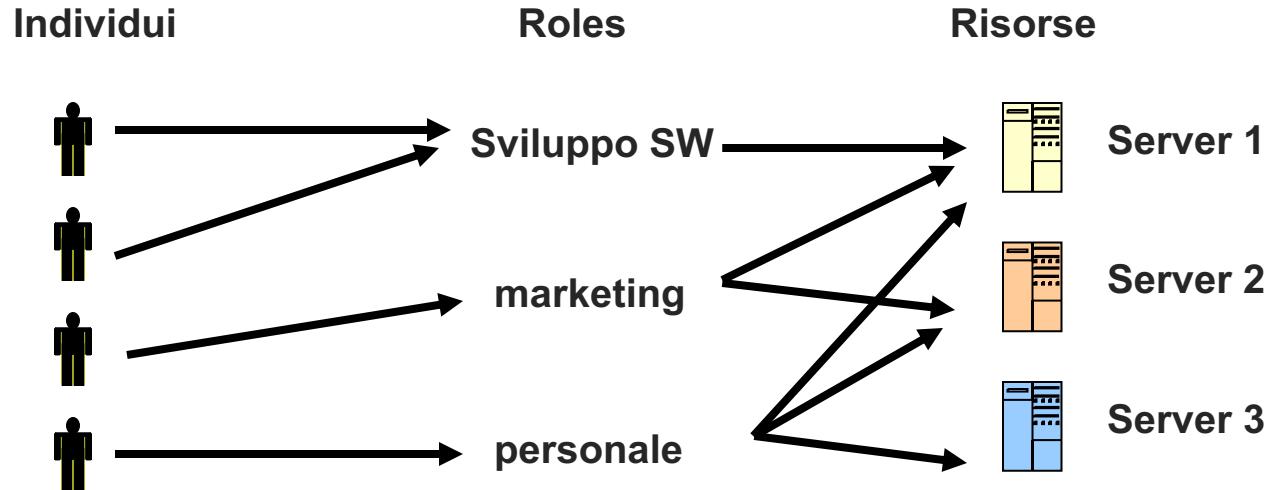


# Gerarchia Di Gruppi

- Anche le attribuzioni di sicurezza procedono in gerarchia
- Ogni gruppo ottiene i permessi del gruppo di livello inferiore;
- Permessi = <diritti, risorse>
- Gerarchia per diritti o risorse:
  - se l'utente ha diritti r, e  $r > s$ , allora l'utente ha diritti s;
  - Se l'utente ha accesso alla directory, l'utente ha accesso a ogni file della directory.



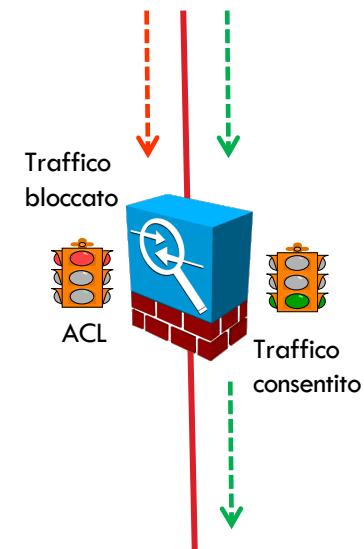
# Role-Based Access Control



**Vantaggio:** gli utenti cambiano più frequentemente dei ruoli

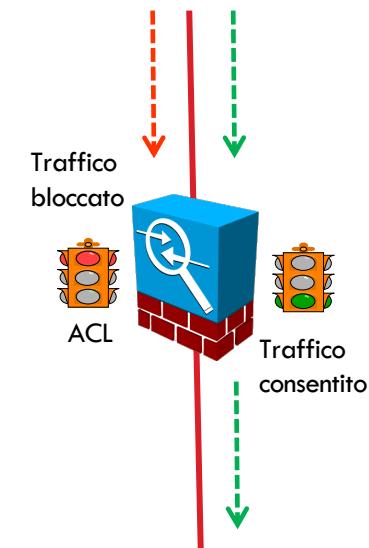
# Meccanismi di filtraggio del traffico: ACL

- Il modo più semplice ed immediato per implementare schemi e politiche di sicurezza è il filtraggio del traffico (**packet filtering**) sui dispositivi di demarcazione (router, L3 switch o firewall)
- Tali dispositivi supportano **liste** di regole **di** filtraggio o **controllo accessi** (ACLs: Access control Lists)
- Ogni pacchetto ricevuto viene confrontato con ciascuna regola, **nell'ordine di apparizione** della stessa nella lista per deciderne **l'inoltro** o lo **scarto**
  - L'applicazione è su **base interfaccia**
  - Le azioni ammissibili sono **permit** (o **allow**) e **deny** (o **drop**)
  - La **direzione di applicazione** dei controlli (**ingresso** o **uscita**) è significativa e definisce la provenienza del traffico interessato



# Meccanismi di filtraggio del traffico: ACL

- Filtraggio applicabile a livello di:
  - **datalink** (su base MAC address)
  - **rete** (su base IP)
  - **trasporto** (su base porta o protocollo)
- Gli elementi che possono essere oggetto di controllo sono:
  - indirizzi sorgente e destinazione (IP/IPv6 o MAC)
  - numeri porta (TCP o UDP)
  - protocollo (IP, TCP, etc.)
  - tipo messaggio ICMP
  - Data e ora

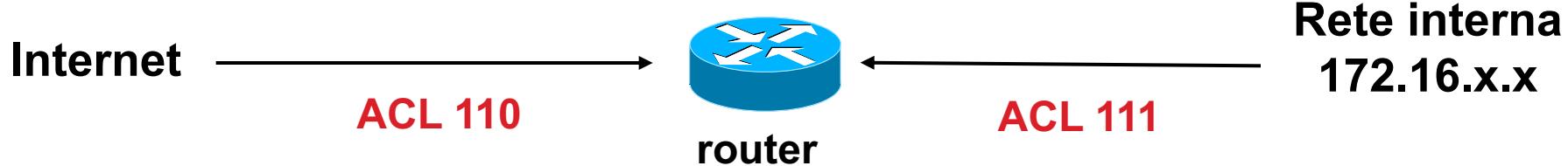


# Sintassi ACL

- Una sola ACL può essere applicata a un'interfaccia in ciascuna specifica direzione (ingresso/uscita):

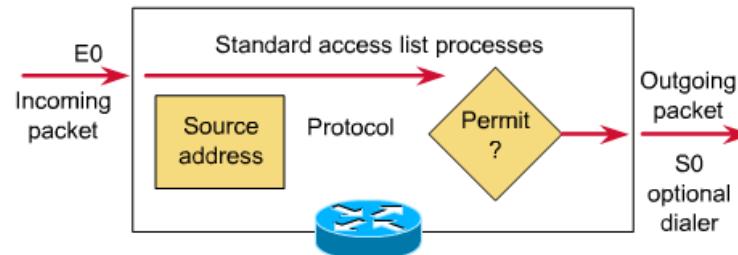
```
interface ethernet 0
  ip access-group 110 in
  ip access-group 111 out
```

- Negli esempi di seguito le ACL 110 e 111 sono applicate rispettivamente in ingresso e uscita sulla border interface che collega un router al mondo esterno

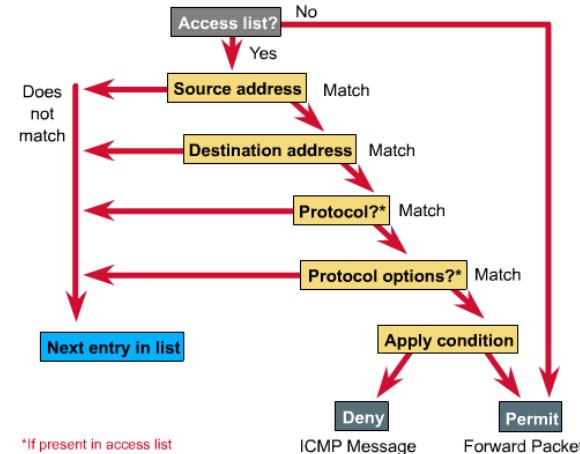
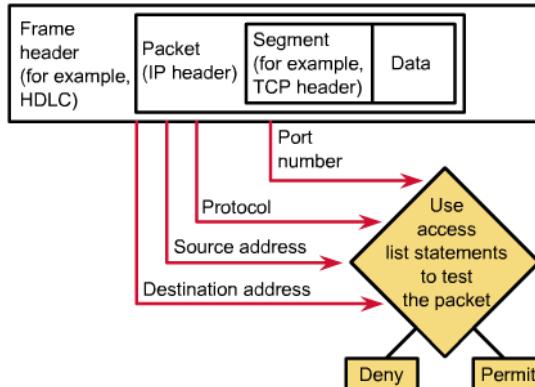


# ACL - Generalità

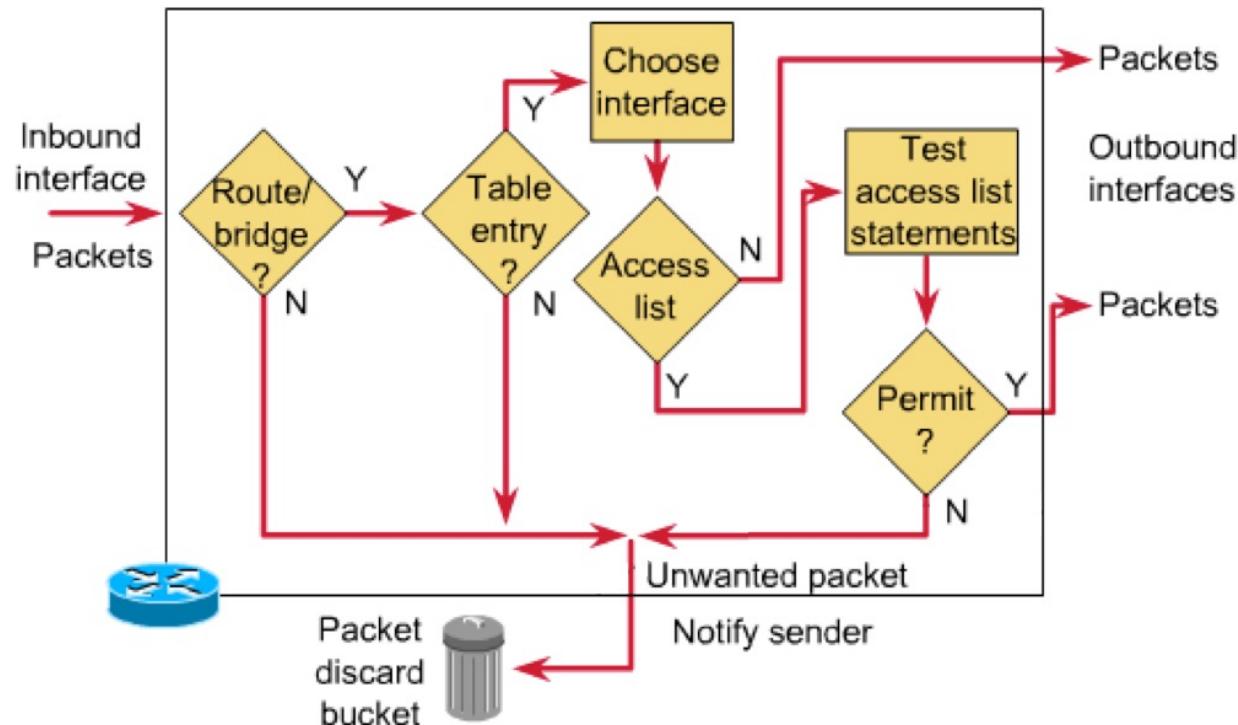
- **ACL Standard:** Controllo accessi su base indirizzo sorgente (host/rete)



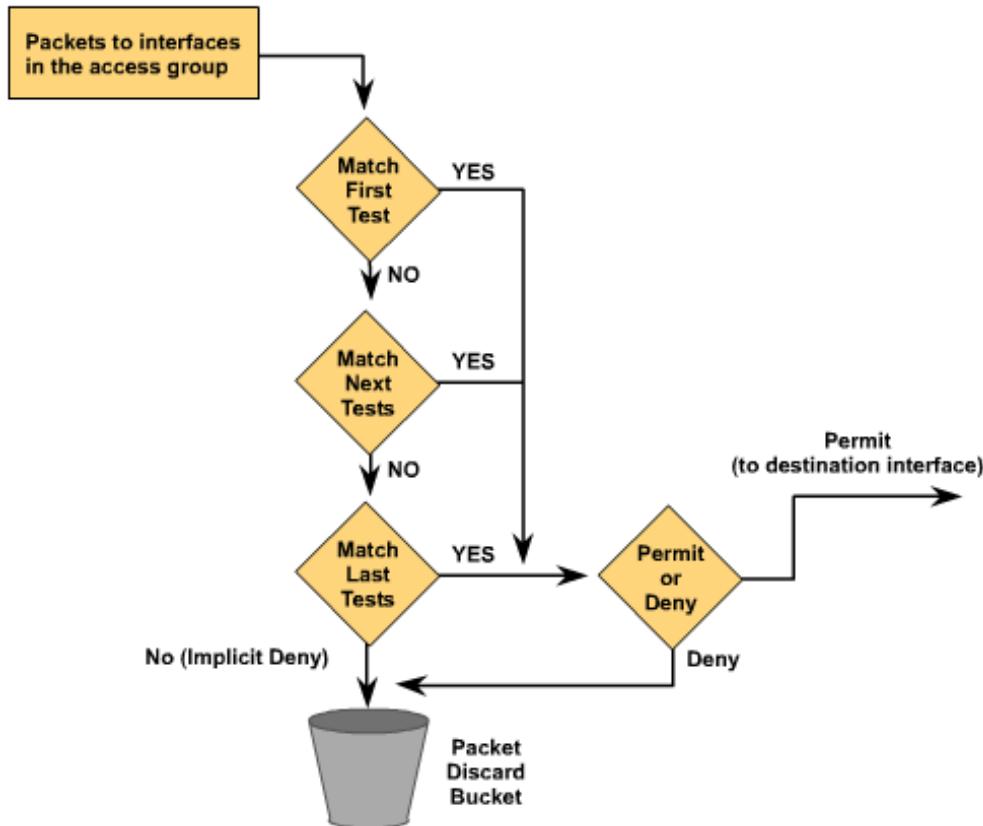
- **ACL Estese:** Controllo accessi su base indirizzo sorgente e destinazione, porta sorgente e destinazione e protocollo



# ACL - Funzionamento

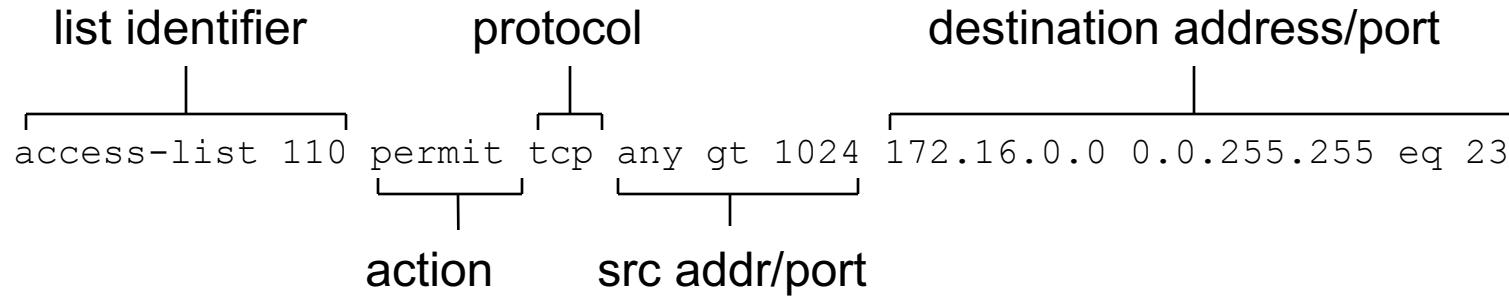


# ACL – Test Regole



# Sintassi ACL

- Una ACL è costituita da regole scandite in sequenza fino al primo match



- Le maschere associate agli indirizzi sono in formato “dotted mask inverso” oppure in formato “/msklen” (es. 0.0.0.255 equivale a /24)

# Wildcard Masks

Access-list 1 permit 172.16.0.0 0.0.255.255				
IP Address	1 0 1 0 1 1 0 0	0 0 0 1 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0
Wildcard mask	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	X X X X X X X X	X X X X X X X X
Match Value	1 0 1 0 1 1 0 0	0 0 0 1 0 0 0 0	X X X X X X X X	X X X X X X X X
Incoming Packet 172.18.4.2				
IP Address	1 0 1 0 1 1 0 0	0 0 0 1 0 0 1 0	0 0 0 0 0 1 0 0	0 0 0 0 0 0 1 0
Wildcard mask	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	X X X X X X X X	X X X X X X X X
Match Value	1 0 1 0 1 1 0 0	0 0 0 1 0 0 1 0	X X X X X X X X	X X X X X X X X
Compares To				
Match Value	1 0 1 0 1 1 0 0	0 0 0 1 0 0 0 0	X X X X X X X X	X X X X X X X X
No Match—Packet Rejected				

# Sintassi ACL

- **Ricerca lunga**: la ricerca viene effettuata finché non c'è un matching (regola con permesso o negazione trovata) o finché la lista non è terminata;
- L'efficienza **dipende dall'ordine** : l'elemento di matching più frequente dovrebbe essere il primo nella lista
- La rimozione di un permesso potrebbe essere senza effetti
- L'opzione **any** sostituisce 0.0.0.0 come IP address e 255.255.255.255 come wildcard mask. Risulta in un matching con qualsiasi indirizzo confrontato.
- L'opzione **host** sostituisce 0.0.0.0 come mask. Questa mask richiede che tutti i bits dell'indirizzo corrispondano. Confronta esattamente un indirizzo.

# Sintassi ACL

- Ogni ACL termina con una regola “deny any any” implicita
- E’ possibile usare operatori relazionali come: eq neq, gt, lt:

```
access-list 110 deny tcp 192.168.1.0 0.0.0.255 any eq www  
access-list 110 deny tcp any eq ftp 192.168.1.25
```

- Si può associare alle ACL un nome logico

```
ip access-list extended allowt permit tcp host 192.132.34.17 any eq 23
```

- E’ possibile definire regole in ACL attivabili su base data/ora ,specificando un “time-range” di validità e uno scope periodico o assoluto

```
time-range no-http periodic weekdays 8:00 to 18:00  
access-list 110 deny tcp any any eq http time-range no-http
```

# Sintassi ACL

- La clausola “established” a fine regola identifica tutte le connessioni TCP che hanno superato la fase di setup (3 way handshake)

```
access-list 110 permit tcp any any established
```

- permette di bloccare tutto il traffico in arrivo dall'esterno, ad eccezione del traffico TCP di ritorno, dovuto ad una sessione TCP iniziata dall'interno.
- verifica, sui pacchetti TCP in arrivo, la presenza dei flag TCP ACK o RST:
  - se sono presenti, il traffico viene permesso,
  - altrimenti si assume che il traffico sia stato generato dall'esterno e verrà bloccato.

# ACLs su Linux: iptables

- ACL semplici possono essere implementati anche su Linux con iptables
- Iptables è usato per impostare, mantenere e ispezionare le regole di filtraggio dei pacchetti IPv4 o IPv6
- È possibile definire diverse tavole
  - filter (default)
  - nat
  - mangle
  - Raw
- Ogni tabella contiene sia catene (chains) integrate che definite dall'utente
  - Ogni chain corrisponde a un elenco di regole che possono essere applicate in maniera opportuna
  - ogni regola specifica i criteri per un pacchetto e un target di azione associato, vale a dire cosa fare con un pacchetto che corrisponde allo schema

# ACLs su Linux: iptables

- **filter:** tabella predefinita che contiene diverse chains integrate
  - **INPUT** (per i pacchetti provenienti dall'esterno destinati all'host)
  - **FORWARD** (per i pacchetti ricevuti dall'esterno e instradati all'esterno)
  - **OUTPUT** (per i pacchetti generati localmente e diretti all'esterno)
- **nat:** gestisce la trasformazione dell'indirizzo (NAT) prima dell'instradamento, affinchè l'indirizzo IP di destinazione sia compatibile con la tabella di routing locale. Contiene tre catene integrate:
  - **PREROUTING** (per alterare i pacchetti non appena arrivano),
  - **OUTPUT** (per alterare i pacchetti generati localmente prima del routing)
  - **POSTROUTING** (per alterare i pacchetti appena stanno per uscire)
- **mangle:** Modifica dell'intestazione TCP o modifica della QoS prima del routing
- **raw:** utilizzato principalmente per la configurazione di eccezioni rispetto al tracciamento delle connessioni

# ACLs su Linux: iptables

- ogni regola specifica i criteri che un pacchetto deve rispettare e un conseguenziale trattamento (target)
  - se il pacchetto non rispetta una specifica regola, viene esaminata la regola successiva nella catena
  - viceversa in presenza di una corrispondenza, il pacchetto viene trattato in base al valore del target
- Possibili target per definire il trattamento di un pacchetto sono:
  - **accept** = consenti il passaggio
  - **drop** = scarta il pacchetto
  - **return** = smette di attraversare questa catena e riprende alla regola successiva nella catena precedente (chiamante)

# Sintassi ACL: iptables

- Possibile creare user-defined chain (analoghe ad ACL) assegnandogli un nome

```
iptables -N acl111
```

- ... e applicarle su specifiche interfacce di ingresso e/o uscita

```
iptables -A FORWARD -i eth1 -o eth0 -j acl110
```

- Una politica di default può essere specificata per ciascuna chain (deny any any)

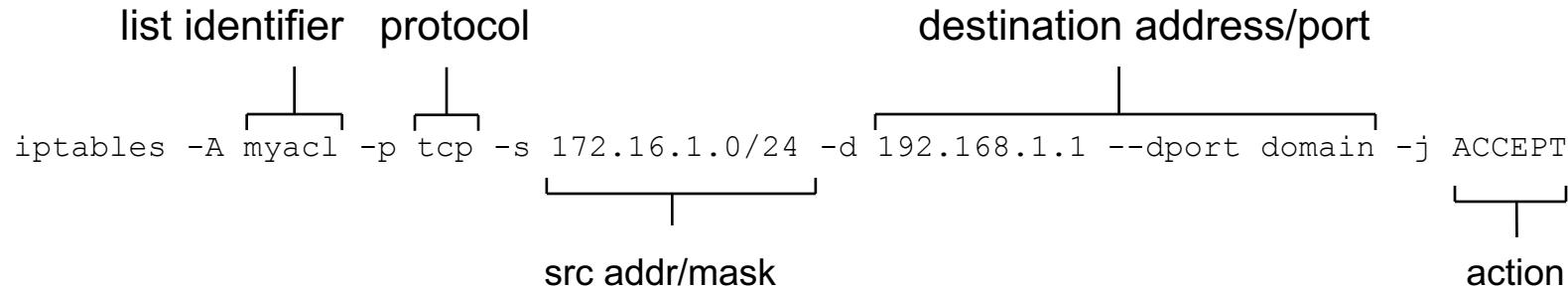
```
iptables -P FORWARD DROP
```

- I parametri più comuni sono:

-p (--protocol) protocol	--sport port
-s (--source) address[/mask]	--dport port
-i (--in-interface) name	-j (--jump) target
-o (--out-interface) name	

# Sintassi ACL: iptables

- La sintassi è intuitiva e basata sulla tradizionale shell command-line

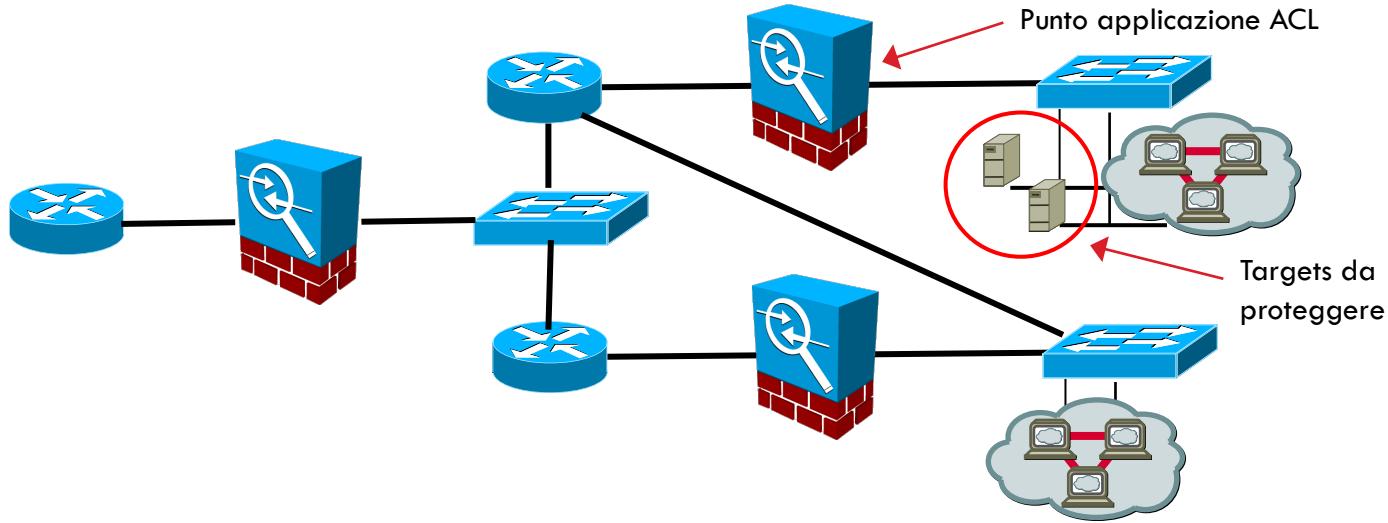


- Come nelle ACLs cisco-like è disponibile la clausula “established”

```
iptables -A acl110 -m state --state ESTABLISHED,RELATED -j ACCEPT
```

# Posizionamento ACL

- Le ACL vanno predisposte quanto più possibile **prossime all'obiettivo** da proteggere
- Ciò permette di **restringere le dimensioni del dominio di sicurezza** in modo da **incrementare l'efficacia** delle politiche di filtraggio implementate e rendere la soluzione più **scalabile**



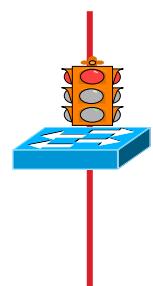
# ACL su switch

- Si può filtrare a livello 2 autorizzando ad es. un solo host a uscire da una porta

```
mac access-list mac-01          interface eth 1
    permit host 00c0.4f00.0407 any      mac port access-group mac-01 in
```

000f.ea91.0408

```
iptables -A FORWARD -i eth1 -m mac --mac-source 00:C0:4F:00:04:07 -j ACCEPT
```



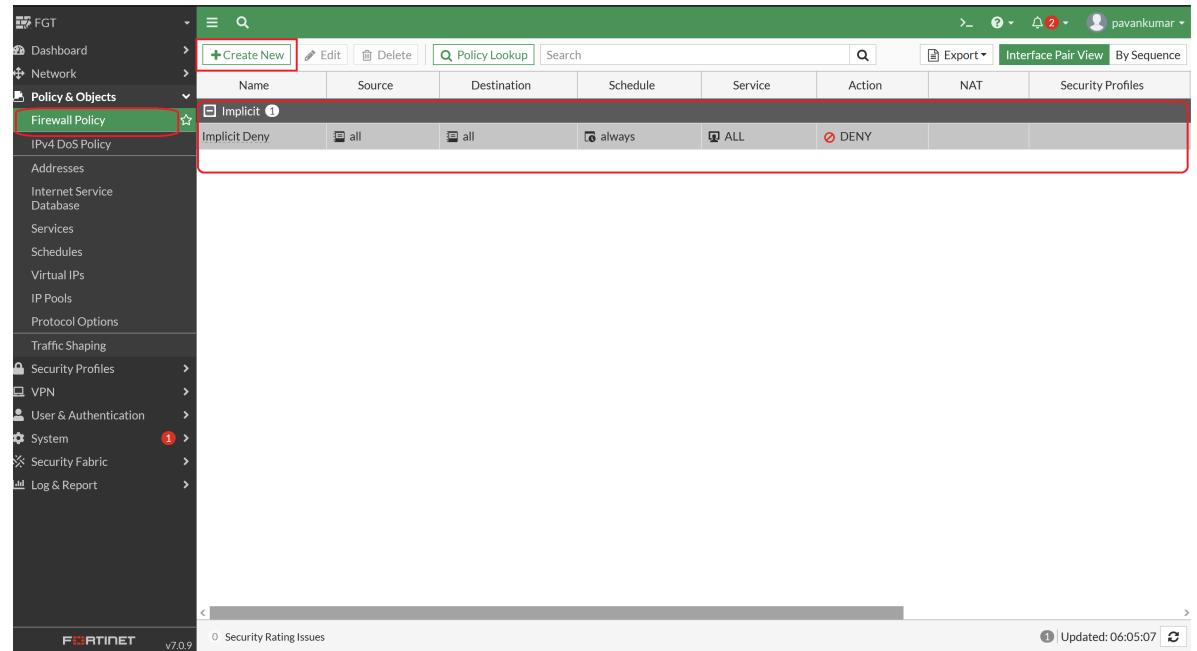
- Può essere utile bloccare completamente il mac di un host compromesso

```
mac-address-table static 000f.ea91.0408 vlan 1 drop
```

```
iptables -A FORWARD -m mac --mac-source 00:0F:EA:91:04:08 -j DROP
```

# ACL su Fortigate NG-Firewall

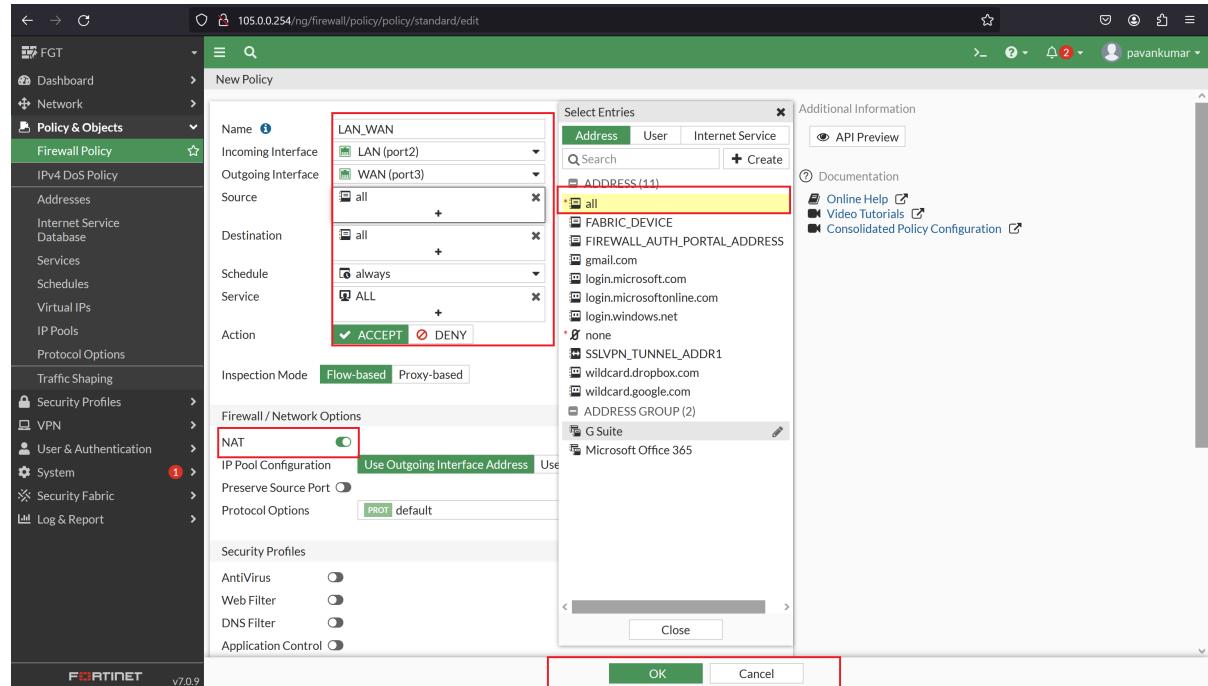
- La logica di controllo accessi su Fortigate Next Generation Firewall si sostanzia nella creazione di specifiche «policies» che consistono in liste di regole di controllo accessi
- Ogni policy definisce:
  - Sorgente e destinazione
  - Servizio (protocollo/porta)
  - Schedule (tempi attivazione)
  - Azione (accept, deny, monitor)
  - Modalità di ispezione
  - Security Profile (vedi in seguito)
- Al termine della lista delle policies il firewall prevede una policy di default «deny all», come nel caso delle ACL in IOS



# ACL su Fortigate NG-Firewall

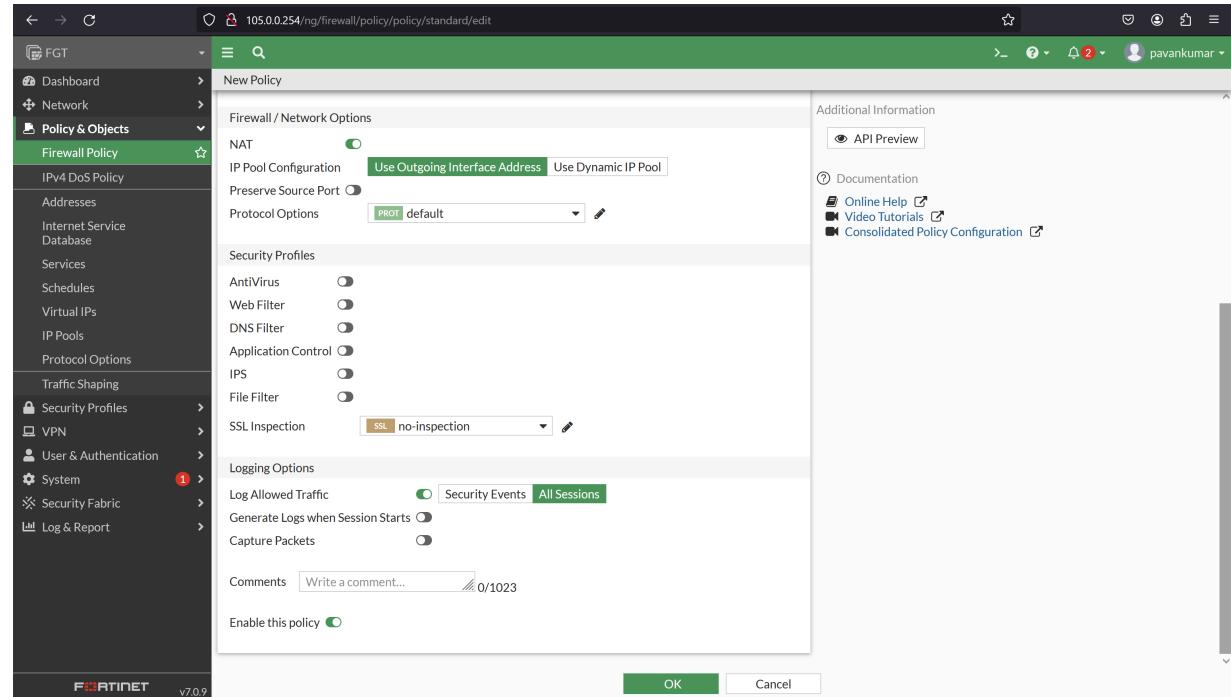
Per controllare il traffico  
nella direzione LAN -> WAN

- Seleziona le interfacce di ingresso (LAN – port2) e uscita (WAN – port3)
- Seleziona gli indirizzi sorgente e destinazione (all = any)
- Scegli il servizio
- Determina l'azione (accept)
- Consenti l'uso del NAT
- Abilita l'ispezione per flusso



# ACL su Fortigate NG-Firewall

- Il security profile definisce particolari trattamenti da effettuare sui flussi di traffico oggetto della policy:
  - Antivirus: effettua scansione antivirus/malware (signatures)
  - Intrusion Prevention System (IPS): individua e previene attacchi
  - Web Filtering: blocca l'accesso a contenuti pericolosi o inappropriati.
  - Application Control: Controlla l'accesso a specifiche applicazioni e protocolli



# ACL su Fortigate NG-Firewall

- Il traffico deve essere autorizzato in entrambe le direzioni, pertanto va creata una policy inversa per autorizzare il traffico da WAN a LAN
- E' possibile creare diverse policy associate che vengono aggiunte l'una dietro l'altra
- Le policy vengono valutate in sequenza e la prima policy corrispondente viene applicata al traffico

The screenshot shows the Fortigate NG-Firewall's web-based management interface. The left sidebar navigation menu includes: Dashboard, Network, Policy & Objects (selected), Firewall Policy (selected), IPv4 DoS Policy, Addresses, Internet Service Database, Services, Schedules, Virtual IPs, IP Pools, Protocol Options, Traffic Shaping, Security Profiles, VPN, User & Authentication, System (with a red notification badge), Security Fabric, and Log & Report.

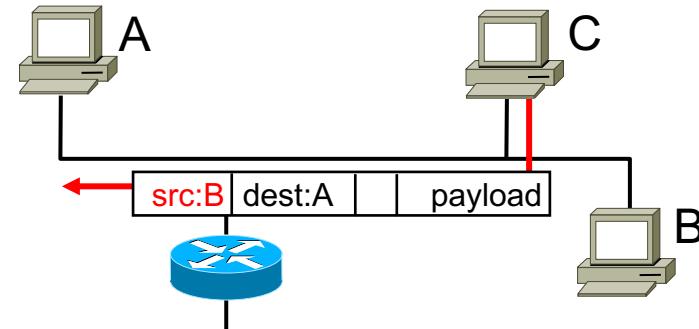
The main content area displays the "Firewall Policy" table. It lists four policies:

Name	Source	Destination	Schedule	Action	NAT	Security Profiles	Log	Bytes	
LAN_WAN	all	all	always	ALL	ACCEPT	Enabled	ssl no-inspection	All	0 B
WAN_LAN	all	all	always	ALL	ACCEPT	Enabled	ssl no-inspection	All	0 B
Implicit	all	all	always	ALL	DENY	Disabled			0 B

Red boxes highlight the "Source" and "Destination" columns for the first two policies (LAN\_WAN and WAN\_LAN) to indicate they are inverse pairs. The "Action" column shows "ACCEPT" for both explicit policies and "DENY" for the implicit policy.

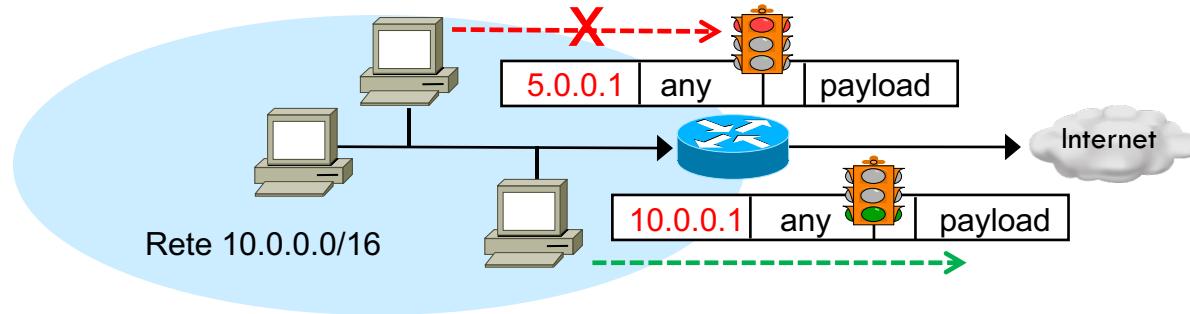
# Spoofing dell'indirizzo IP

- L'indirizzo IP sorgente è attualmente l'**unico** meccanismo **di identificazione della provenienza** disponibile su Internet
- La **falsificazione** di tale dato è alla base di buona parte degli attacchi e delle azioni ostili
- Lo **spoofing** consiste nella falsificazione dell'indirizzo sorgente
  - Qualsiasi utente è in grado di generare pacchetti IP con un valore qualsiasi dei campi previsti dalla struttura protocollore
  - Pertanto è immediato cambiare l'indirizzo sorgente de pacchetti IP per impedire qualsiasi formad identificazione
  - Il risultato è che C nel'attaccare A assume l'identità di B



# Filtraggio in Ingresso

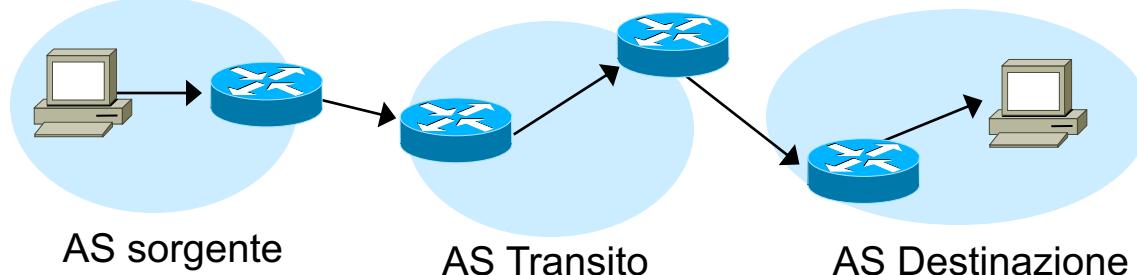
- Soluzione: controllo ed enforcing della correttezza dell'origine dei pacchetti generati



- Politica di filtraggio in ingresso (RFC 2827, 2000): Un router di bordo inoltra esclusivamente i pacchetti con indirizzi sorgente legittimi

# Problemi pratici di realizzazione

- E' necessario che lo facciano tutte le organizzazioni coinvolte e gli ISP di transito
- Il tutto si basa su una logica di collaborazione e fiducia a livello globale
  - Se il 10% degli ISPs non lo implementa  $\Rightarrow$  inefficace
- Altra soluzione: enforcing/validazione IP sorgenti a livello AS peering



- Un pacchetto può transitare solo se l'AS di transito valida la sorgente

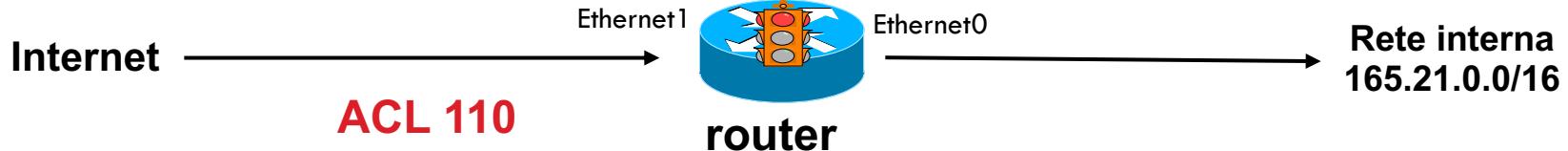
# Anti-Spoofing in ingresso

- Il modo più semplice di proteggersi è quello di scartare tutto il traffico in ingresso con indirizzi sorgente inammissibili rispetto alla provenienza

```
interface ethernet 1  
ip access-group 110 in
```

```
# blocca traffico spoof entrante da eth1  
iptables -A FORWARD -i eth1 ...
```

- Blocca tutto il traffico con indirizzi origine 165.21.0.0/16 se provenienti dall'esterno (sono i miei indirizzi interni!)



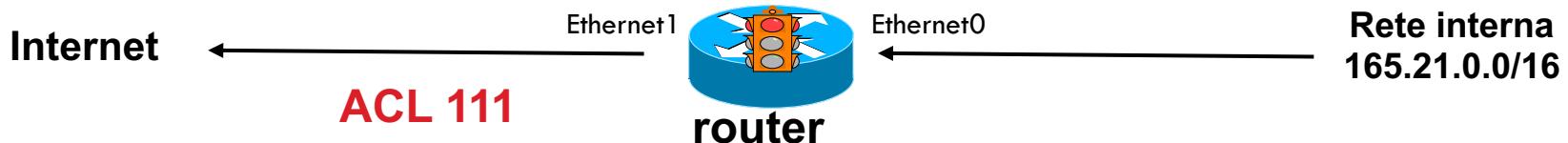
# Anti-Spoofing in uscita

- Per prevenire inoltre spoofing, volontari o involontari, dall'interno della propria rete verso l'esterno, analoghe misure di filtraggio vanno applicate in uscita

```
interface ethernet 1  
ip access-group 111 out
```

```
# non inoltrare il traffico spoof da eth0  
iptables -A FORWARD -i eth0 ...
```

- Blocca qualsiasi pacchetto in uscita con indirizzo di origine che non ricade nella rete 165.21.0.0/16



# Anti-Spoofing ACL (ingresso/uscita)

## Anti spoofing in ingresso

! Blocca i traffico dall' esterno con indirizzi sorgente interni:

```
access-list 110 deny ip 165.21.0.0 0.0.255.255 any log  
access-list 110 permit ip any any
```

```
iptables -A FORWARD -i eth1 -s 165.21.0.0 /16 -j DROP
```

## Anti spoofing in uscita

! Blocca il traffico uscente con IP sorgente estranei:

```
access-list 111 permit ip 165.21.0.0 0.0.255.255 any  
access-list 111 deny ip any any log
```

```
iptables -A FORWARD -i eth0 -s ! 165.21.0.0 /16 -j DROP
```

# Anti-Spoofing: indirizzi riservati

Address Block	Name	RFC
0.0.0.0/8	"This host on this network"	[RFC1122], Section 3.2.1.3
10.0.0.0/8	Private-Use	[RFC1918]
100.64.0.0/10	Shared Address Space	[RFC6598]
127.0.0.0/8	Loopback	[RFC1122], Section 3.2.1.3
169.254.0.0/16	Link Local	[RFC3927]
172.16.0.0/12	Private-Use	[RFC1918]
192.0.0.0/24 [2]	IETF Protocol Assignments	[RFC6890], Section 2.1
192.0.0.0/29	IPv4 Service Continuity Prefix	[RFC7335]
192.0.0.8/32	IPv4 dummy address	[RFC7600]
192.0.0.9/32	Port Control Protocol Anycast	[RFC7723]
192.0.0.10/32	Traversal Using Relays around NAT Anycast	[RFC8155]
192.0.0.170/32, 192.0.0.171/32	NAT64/DNS64 Discovery	[RFC7050], Section 2.2
192.0.2.0/24	Documentation (TEST-NET-1)	[RFC5737]
192.31.196.0/24	AS112-v4	[RFC7535]
192.52.193.0/24	AMT	[RFC7450]
192.88.99.0/24	Deprecated (6to4 Relay Anycast)	[RFC7526]
192.168.0.0/16	Private-Use	[RFC1918]
192.175.48.0/24	Direct Delegation AS112 Service	[RFC7534]
198.18.0.0/15	Benchmarking	[RFC2544]
198.51.100.0/24	Documentation (TEST-NET-2)	[RFC5737]
203.0.113.0/24	Documentation (TEST-NET-3)	[RFC5737]
240.0.0.0/4	Reserved	[RFC1112], Section 4
255.255.255.255/32	Limited Broadcast	[RFC8190] [RFC919], Section 7

# Filtraggio anti-spoofing indirizzi riservati

In aggiunta alle tecniche di anti spoofing di base sopra citate è opportuno filtrare in ingresso (ed eventualmente in uscita) gli indirizzi riservati (RFC 1918) o non correttamente instradabili

```
! Blocca i traffico dall' esterno con indirizzi IP non instradabili:  
access-list 110 deny ip 10.0.0.0      0.255.255.255 any log  
access-list 110 deny ip 172.16.0.0    0.15.255.255 any log  
access-list 110 deny ip 192.168.0.0  0.0.255.255 any log  
access-list 110 deny ip 127.0.0.0    0.255.255.255 any log  
access-list 110 permit ip any any
```

```
interface Serial0/1  
    ip access-group 110 in
```

Oppure rigirare tutto il traffico inammissibile sulla null interface

```
ip route 10.0.0.0    255.0.0.0    null0  
ip route 172.16.0.0  255.240.0.0  null0  
ip route 192.168.0.0 255.255.0.0  null0
```

# Definizione di una semplice politica di controllo accessi

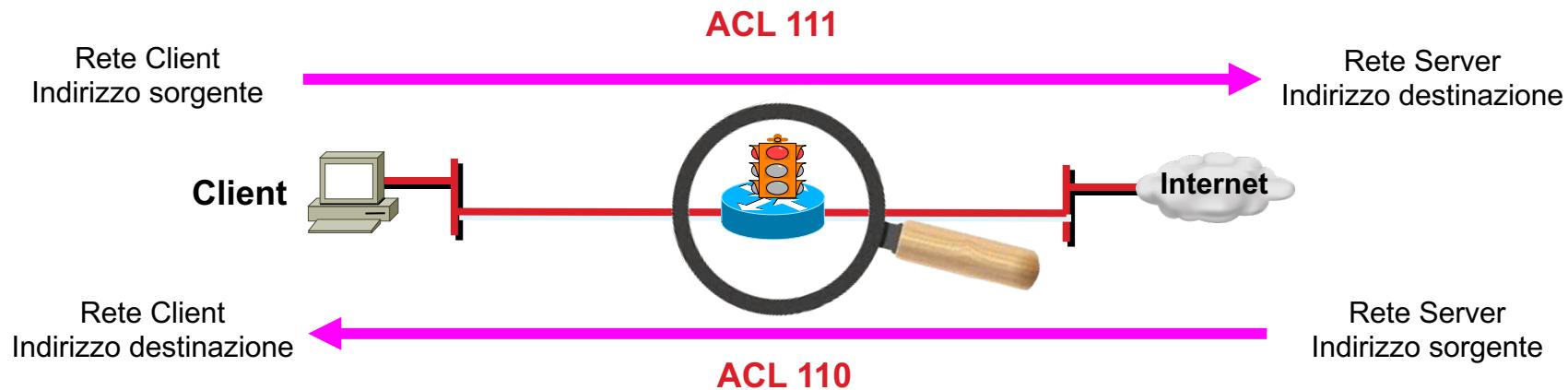
## Esempio (in uno scenario SOHO)

- Consenti in uscita la fruizione di tutti i servizi TCP (www, e-mail, telnet, etc.)  
– Full Internet access
- Permetti in ingresso solo l' accesso ad un numero estremamente limitato di servizi TCP (e-mail, www) erogati solo da hosts specifici e controllati
- Consenti in ingresso il solo traffico relativo alle sessioni aperte dall' interno (attenzione a ftp!)
- Consenti Ping e Traceroute dall' interno e non dall' esterno
- Consenti in maniera controllata i meccanismi DNS



# Dove applicare i controlli?

- Il filtraggio tramite ACL è praticabile su qualsiasi dispositivo di confine
- Nel nostro semplice esempio è efficace agire a livello del “border router”, che separa i due domini di sicurezza distinti (inside, outside) e sui cui è possibile controllare in maniera centralizzata i flussi di traffico che attraversano tali domini.
- Ci bastano 2 ACL (110 e 111) da applicare rispettivamente in ingresso e uscita



# Applicazione dei controlli

```
interface ethernet 1
    ip access-group 110 in
    ip access-group 111 out
```

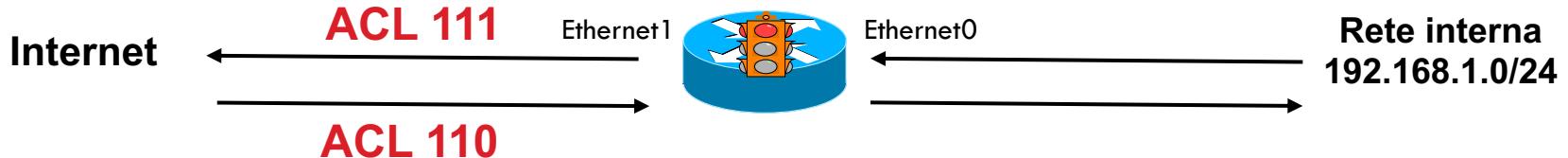
```
iptables -P FORWARD DROP
```

```
iptables -N acl110
```

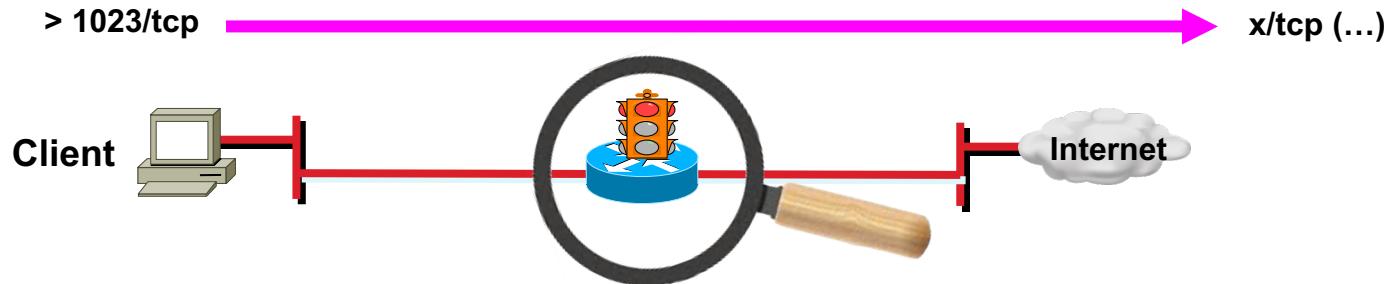
```
iptables -N acl111
```

```
iptables -A FORWARD -i eth1 -o eth0 -j acl110
```

```
iptables -A FORWARD -i eth0 -o eth1 -j acl111
```



# Traffico consentito in uscita

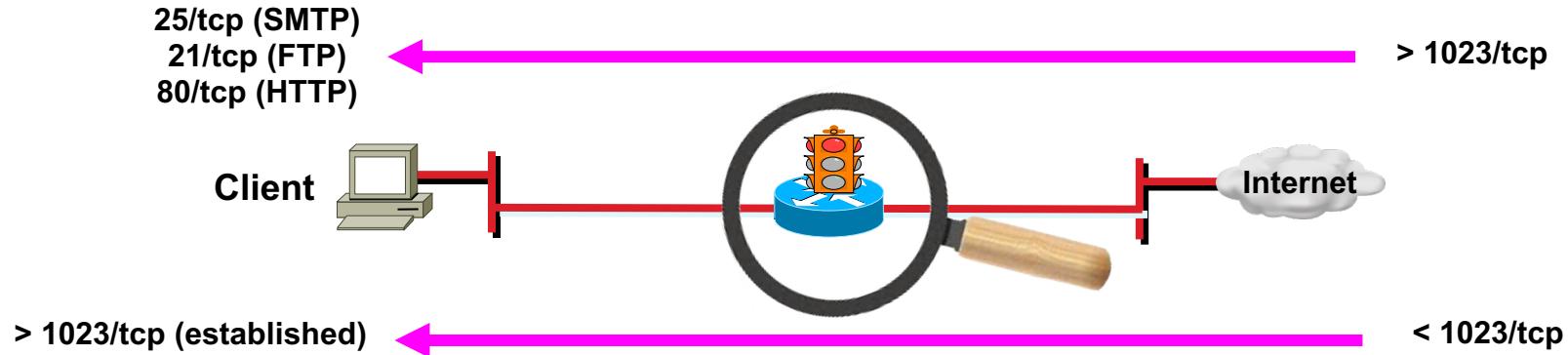


- Qualsiasi connessione TCP in uscita è consentita senza alcuna restrizione
- Le connessioni UDP e raw sono implicitamente bloccate

```
access-list 111 permit tcp 192.168.1.0 0.0.0.255 any
```

```
iptables -A acl111 -p tcp -s 192.168.1.0/24 -j ACCEPT
```

# Traffico consentito in ingresso



- L'accesso ai servizi interni va controllato con la massima attenzione e consentito solo verso gli hosts erogatori di servizi
- Va consentito il traffico in ingresso a ritroso (da outside a inside) solo se relativo a connessioni già aperte dall'interno (established)

# Traffico consentito in ingresso

**! Blocco spoofing in ingresso**

```
access-list 110 deny ip 192.168.1.0 0.0.0.255 any
```

**! Accesso TCP verso i soli servizi ufficiali**

```
access-list 110 permit tcp any host 192.168.1.1 eq 25
```

```
access-list 110 permit tcp any host 192.168.1.1 eq 21
```

```
access-list 110 permit tcp any host 192.168.1.1 eq 80
```

**! Solo traffico TCP di ritorno di sessioni aperte dall'interno**

```
access-list 110 permit tcp any 192.168.1.0 0.0.0.255 established
```

```
iptables -A acl110 -s 192.168.1.0/24 -j DROP
```

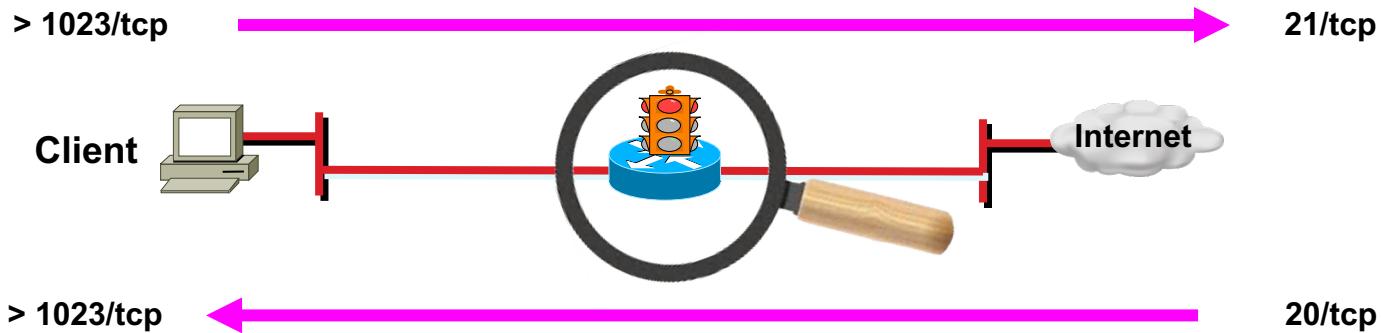
```
iptables -A acl110 -p tcp -d 192.168.1.1 --dport www -j ACCEPT
```

```
iptables -A acl110 -p tcp -d 192.168.1.1 --dport smtp -j ACCEPT
```

```
iptables -A acl110 -p tcp -d 192.168.1.1 --dport ftp -j ACCEPT
```

```
iptables -A acl110 -m state --state ESTABLISHED,RELATED -j ACCEPT
```

# Problemi con FTP

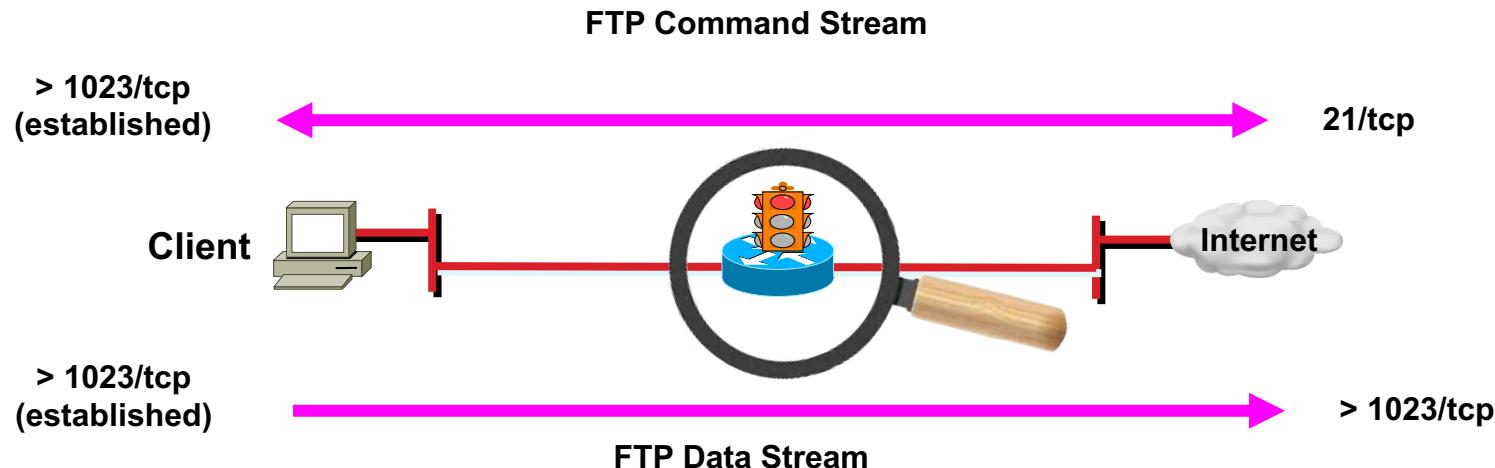


- Dopo aver aperto dall'interno la connessione al canale di controllo è necessario per ciascun trasferimento garantire la possibilità di aprire a ritroso le connessioni dati

```
access-list 110 permit tcp any eq 20 192.168.1.0 0.0.0.255 gt 1023
```

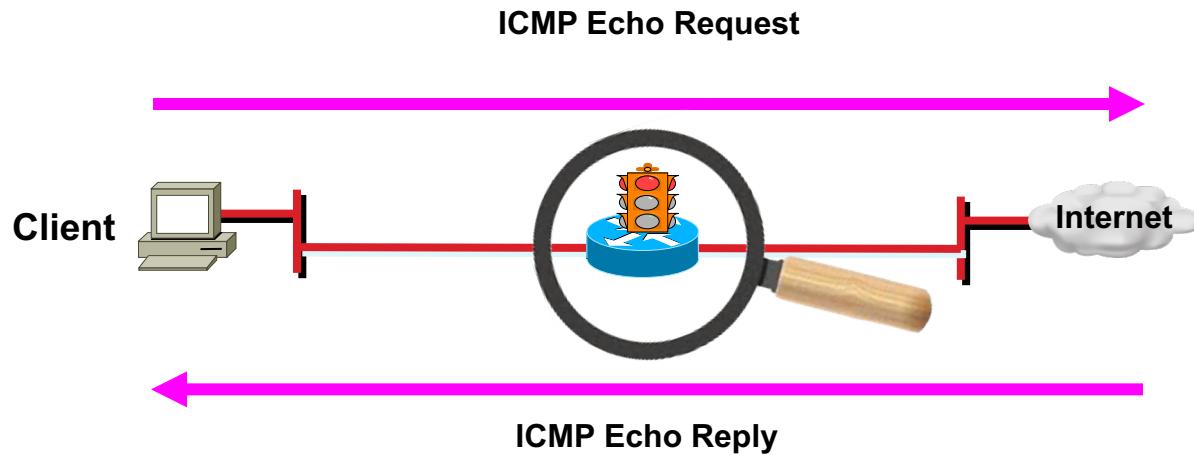
```
iptables -A acl110 -p tcp --match multiport --sport 20 --dports 1024:65535 -j ACCEPT
```

# FTP in modalità passiva



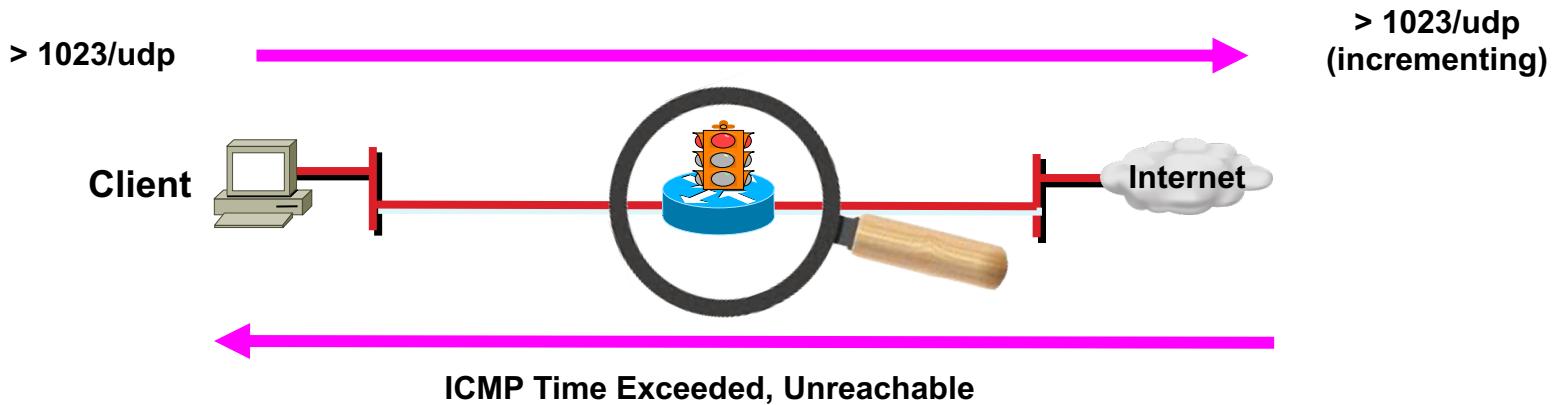
- L'FTP in modalità passiva elimina la necessità di accedere alla porta 20 a ritroso e risolve il problema a monte
- Per prevenire il problema è possibile usare un firewall (fa "stateful" filtering)

# Ping



- Per garantire la funzionalità del ping iniziato dall'interno è necessario consentire a ritroso i messaggi *ICMP echo reply* in risposta a quelli di *echo request*
- Il ping iniziato dall'esterno è inibito

# Traceroute



- Per garantire la funzionalità del traceroute iniziato dall'interno è necessario consentire a ritroso i messaggi *ICMP time exceeded* (step intermedi) e quelli di *ICMP port unreachable* (condizione finale)
- Il traceroute iniziato dall'esterno è inibito

# Ping e Traceroute

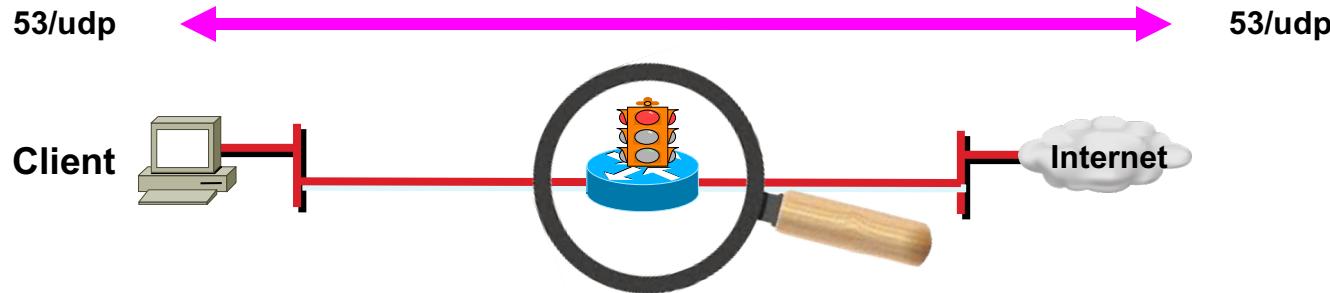
```
access-list 110 permit icmp any 192.168.1.0 0.0.0.255 echo-reply
access-list 110 permit icmp any 192.168.1.0 0.0.0.255 time-exceeded
access-list 110 permit icmp any 192.168.1.0 0.0.0.255 unreachable

access-list 111 permit icmp 192.168.1.0 0.0.0.255 any echo
access-list 111 permit udp 192.168.1.0 0.0.0.255 gt 1023 any gt 1023
```

```
iptables -A acl110 -p icmp --icmp-type echo-reply -s 0/0 -d 192.168.1.0/24 -j ACCEPT
iptables -A acl110 -p icmp --icmp-type destination-unreachable -s 0/0 -d 192.168.1.0/24 -j ACCEPT
iptables -A acl110 -p icmp --icmp-type time-exceeded -s 0/0 -d 192.168.1.0/24 -j ACCEPT

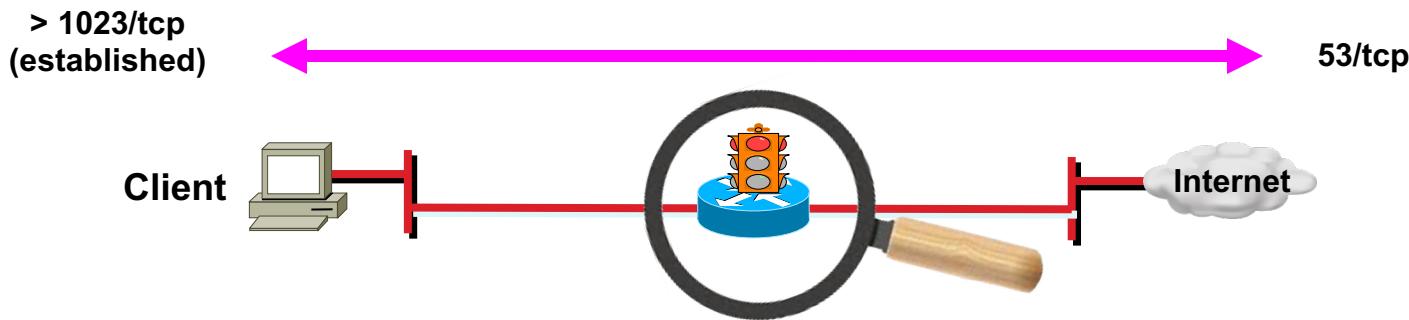
iptables -A acl111 -p icmp --icmp-type echo-request -s 192.168.1.0/24 j ACCEPT
iptables -A acl111 -p udp --match multiport --sports 1024:65535 --dports 1024:65535 -j ACCEPT
```

# DNS: Query



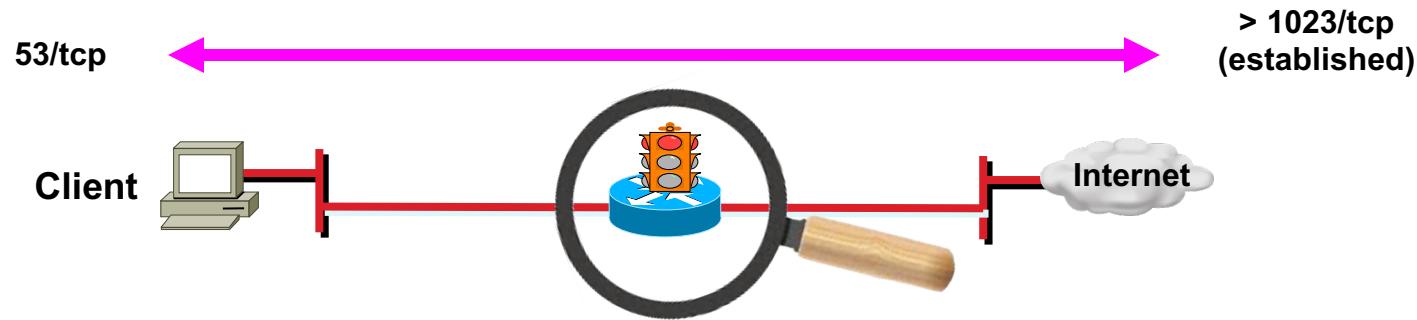
- Vanno garantite richieste e risposte DNS (messaggi UDP nelle due direzioni verso la porta 53)

# DNS: risposte bulk



- ... analogo discorso vale per le risposte bulk su sessioni TCP (solo se iniziati dall'interno)
- In ogni caso le query dall'esterno basate su TCP/53 vanno bloccate
- Il tutto va ammesso solo verso il DNS server

# DNS: Trasferimenti di zona



- ... nonché nella direzione opposta per i trasferimenti di zona su sessioni TCP
- I trasferimenti vanno consentiti solo agli hosts autorizzati (es. 172.16.1.1 secondario esterno)

# DNS: regole di filtraggio

```
! Risposte o richieste DNS al resolver interno
access-list 110 permit udp any host 192.168.1.1 eq 53
! Zone transfers
access-list 110 permit tcp host 172.16.1.1 host 192.168.1.1 eq 53

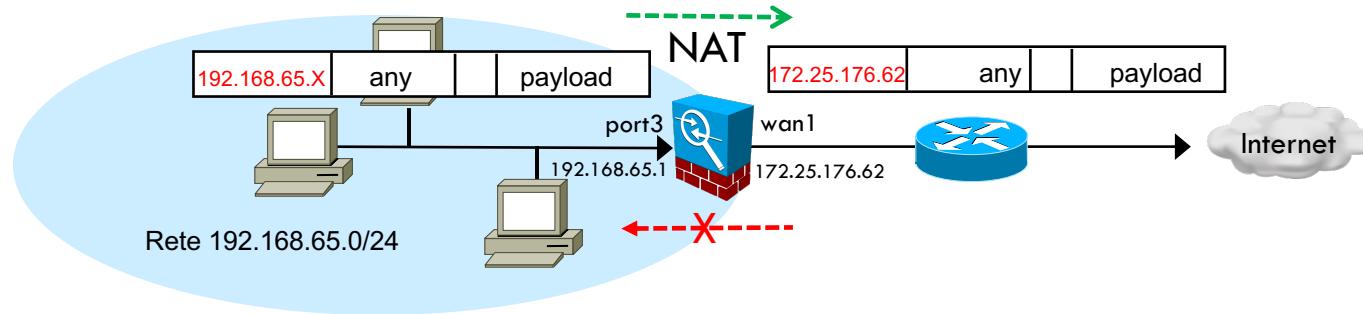
! Server to server queries
access-list 111 permit udp host 192.168.1.1 eq 53 any eq 53
```

```
iptables -A acl110 -p udp -d 192.168.1.1 --dport domain -j ACCEPT
iptables -A acl110 -p tcp -s 172.16.1.1 -d 192.168.1.1 --dport domain -j ACCEPT

iptables -A acl111 -p udp -s 192.168.1.1 --sport 53 --dport 53 -j ACCEPT
```

# Un esempio di configurazione via GUI

- Consideriamo una tipica configurazione di rete



- Il FW fa da DHCP server per la rete interna e blocca tutto il traffico entrante mentre consente l'accesso full internet dalla rete interna verso l'esterno applicando il NAT in modalità overload.

# Configurazione interfaccia WAN

- Network > Interfaces.
  - Imposta la larghezza di banda per l'interfaccia
  - Configura «Role» a «WAN»
  - Assegna l'indirizzo in modalità «Manuale»
  - Abilita l'interfaccia
- Questo si traduce in termini di CLI commands:

```
config system interface
edit port2
set mode static
set ip 172.25.176.62/24
set allowaccess ping
set alias "WAN"
set role wan
end
```

The screenshot shows the FortiGate VM64 management interface. The left sidebar has a 'Network' section with a 'Interfaces' tab highlighted by a red box. The main pane shows a table of interfaces. A second red box highlights the 'wan port (port2)' entry in the table, which is identified as a 'Physical Interface' with the IP address 192.168.2.1/255.255.255.0.

Interface Name: wan1 (90:6C:AC:2A:14:5A)

Alias:

Link Status: Up

Type: Physical Interface

Estimated Bandwidth: 10000 Kbps Upstream | 20000 Kbps Downstream

Tags: Role: WAN [+ Add Tag Category](#)

Address: Addressing mode: Manual DHCP  
IP/Network Mask: 172.25.176.62/255.255.255.0

# Configurazione interfaccia LAN

- Network > Interfaces.
  - Configura «Role» a «LAN»
  - Imposta la modalità di indirizzamento su Manuale e imposta i campi IP/netmask sull'indirizzo IP privato che si desidera utilizzare per l'interfaccia LAN
  - Configura il server DHCP

```
config system interface
edit port3
set mode static
set ip 192.168.65.1/24
set allowaccess ping
set alias »LAN«
set role lan
end
```

```
config system dhcp server
edit 1
set dns-service default
set default-gateway 192.168.65.1
set netmask 255.255.255.0
set interface "port3"
config ip-range
edit 1
set start-ip 192.168.65.2
set end-ip 192.168.65.254
end
end
```

The screenshot shows the configuration interface for a LAN interface named 'lan'. The interface is of type 'Software Switch' and includes members from port 3 to port 10. It is assigned the 'LAN' role. The addressing mode is set to 'Manual' with IP 192.168.65.1 and netmask 255.255.255.0. The interface supports administrative access via various protocols like HTTPS, CAPWAP, and RADIUS Accounting, and is configured as a DHCP server. An address range from 192.168.65.2 to 192.168.65.254 is defined with a netmask of 255.255.255.0. Default gateway and DNS settings are also specified.

Interface Name: lan

Alias:

Type: Software Switch

Interface Members:

- port3 ✘
- port4 ✘
- port5 ✘
- port6 ✘
- port7 ✘
- port8 ✘
- port9 ✘
- port10 ✘

Tags:

Role: LAN

Address:

Addressing mode: Manual

IP/Network Mask: 192.168.65.1/255.255.255.0

Administrative Access:

IPv4:

- HTTPS
- HTTP
- PING
- FMG-Access

- CAPWAP
- SSH
- SNMP
- FTM

- RADIUS Accounting
- FortiTelemetry

DHCP Server

Address Range:

Starting IP	End IP
192.168.65.2	192.168.65.254

Netmask: 255.255.255.0

Default Gateway: Same as Interface IP

DNS Server: Same as System DNS

Advanced...

# Configurazione default route e DNS

- Network > Static Routes.
  - Se esiste un solo link di uscita verso Internet (stub network) imposta «Destination» su «Subnet» e lasciare l'indirizzo impostato su 0.0.0.0/0.0.0.0.
  - Imposta «Gateway» sull'indirizzo IP presente sul next hop rispetto all'Interfaccia WAN
- Network > DNS,
  - Selezione «Specify», e imposta i server Primary e Secondary

```
config router static
edit 0
set gateway 172.25.176.1
set device port3
end

config system dns
set primary 208.91.112.53
set secondary 208.91.112.52
end
```

Destination <span style="color: #0070C0;">i</span>	Subnet	Named Address	Internet Service
	0.0.0.0/0.0.0.0		
Gateway	172.25.176.1		
Interface	wan1		
Administrative Distance <span style="color: #0070C0;">i</span>	10		
Comments	0/255		
Status	<span style="color: green;">Enabled</span>	<span style="color: red;">Disabled</span>	
<span style="color: #0070C0;">+</span> Advanced Options			

DNS Servers	Use FortiGuard Servers	<span style="color: green;">Specify</span>
Primary DNS Server	208.91.112.53	
Secondary DNS Server	208.91.112.52	
Local Domain Name		

# Configurazione Security Policy

- Policy & Objects > IPv4 Policy.
  - Assegnare alla policy un Nome che indichi che la policy riguarderà il traffico verso Internet (nell'esempio, Internet).
  - Imposta l'interfaccia in entrata su lan e l'interfaccia in uscita su wan1. Imposta origine, rete di destinazione, pianificazione temporale (periodo di validità) e servizi (protocolli)
  - Assicurati che l'azione sia impostata su «ACCEPT».
- Attiva il NAT e seleziona «Usa indirizzo interfaccia in uscita» per impostare la modalità «overload»

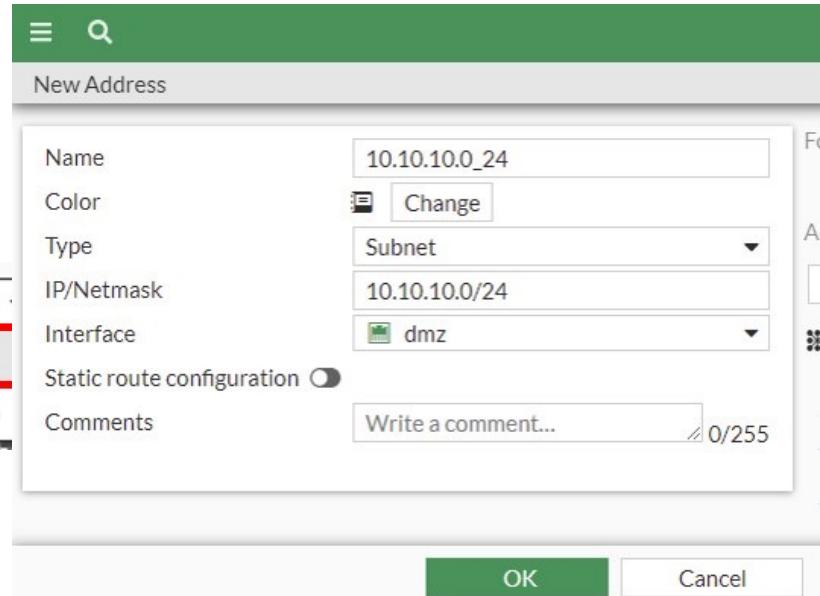
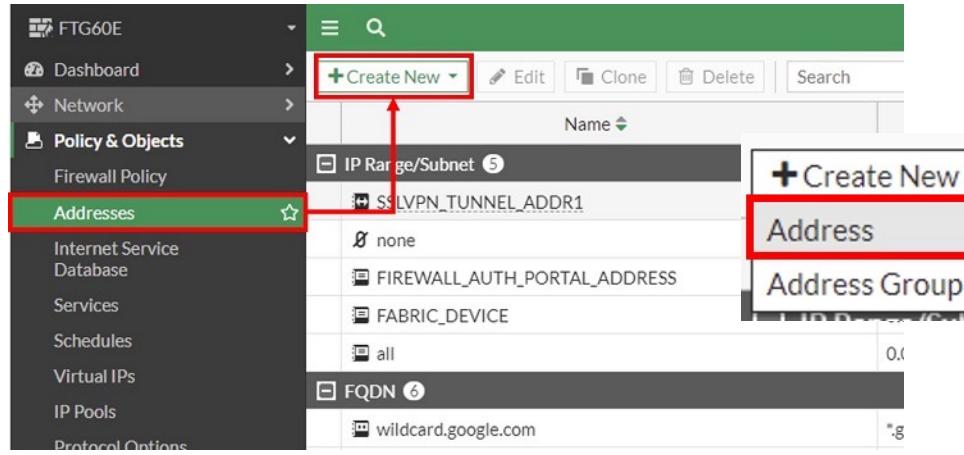
Name	Internet
Incoming Interface	lan
Outgoing Interface	wan1
Source	all
Destination	all
Schedule	always
Service	ALL
Action	<input checked="" type="button"/> ACCEPT <input type="button"/> DENY <input type="button"/> LEARN

Firewall / Network Options

NAT

IP Pool Configuration  Use Outgoing Interface Address  Use Dynamic IP Pool

# Policy Selettive



Associo un nuovo blocco di indirizzi (10.10.10.0/24) all'interfaccia DMZ  
Select [Policy & Objects > Addresses] and click Create New.

# Policy Selettive

The screenshot shows the Fortinet GUI with the following details:

- Left Sidebar:** FTG60E, Dashboard, Network, Policy & Objects (selected), Firewall Policy (selected), Addresses, Internet Service, Database, Services, Schedules, Virtual IPs, IP Pools.
- Header Bar:** + Create New (highlighted with a red box and arrow), Edit, Delete, Policy Lookup, Search.
- Table View:** Name, Source, Destination. One row is selected: Implicit Deny, all, all.

The screenshot shows the configuration dialog for a new firewall policy:

Name	10.10.10.0_24_to_Internet
Incoming Interface	dmz
Outgoing Interface	wan1
Source	10.10.10.0_24
Destination	all
Schedule	always
Service	Web Access
Action	<input checked="" type="checkbox"/> ACCEPT <input type="checkbox"/> DENY
Inspection Mode	Flow-based (selected)   Proxy-based

- Creo una nuova policy che consente alla classe 10.10.10.0/24 il solo accesso verso servizi web.
- Abilito il NAT in overload

The screenshot shows the Firewall / Network Options dialog:

NAT	<input checked="" type="checkbox"/>
IP Pool Configuration	<input type="checkbox"/> Use Outgoing Interface Address <input type="checkbox"/> Use Dynamic IP Pool
Preserve Source Port	<input type="checkbox"/>
Protocol Options	PROT default

# Traffic logging

- Abilitazione selettiva del logging
  - Visualizzazione logs



Source	Source Device	Bytes (Sent/Received)	Sessions	Bandwidth
192.168.65.2	jburkholder-pc	19.92 MB	300	3 Mbps

Summary of 192.168.65.2	
Device	 jburkholder-pc
Applications Detected	3
Bytes (Sent/Received)	27.10 MB 
Bandwidth	1.94 Mbps 
Sessions	287
Time Period	Realtime
FortiGate	FG800D3915800295

Destinations	Applications	Countries	Policies	Domains	Categories	Source Interfaces	Destination Interfaces	Sessions	Bandwidth
Destination						Bytes (Sent/Received)	Sessions	Bandwidth	
HR1 sn-gvbxgn-tvve.googlevideo.com (209.148.198.204)						19.06 MB	1	2 Mbps	
googleadapis.l.google.com (172.217.10.106)						3.93 MB	3	48 bps	
ytimg.l.google.com (172.217.10.238)						1.65 MB	1	256 bps	
fcmatch.youtube.com (172.217.9.238)						943.07 kB	2	40 bps	
gstaticadssl.l.google.com (172.217.9.227)						339.81 kB	2	88 bps	
www.google.ca (216.58.193.67)						317.69 kB	1	48 bps	
pagead2.googlesyndication.com (172.217.11.2)						297.90 kB	1	48 bps	
pagead-googlehosted.l.google.com (172.217.9.225)						152.98 kB	1	48 bps	
208.91.112.53						86.07 kB	222	288 bps	
partnerad.doubleclick.net (172.217.10.98)						83.45 kB	1	48 bps	
redirector.gvt1.com (172.217.10.110)						65.40 kB	2	40 bps	
yt3.ggpht.com (172.217.10.97)						63.22 kB	1	40 bps	
www.google.com (172.217.3.164)						27.01 kB	1	48 bps	
adservice.google.com (172.217.12.194)						21.46 kB	2	112 bps	
cm.g.doubleclick.net (172.217.12.130)						16.69 kB	2	88 bps	
pipeline-edge-prod-25-561439127.us-west-2.elb.amazonaws.com (54.68.157.14)						13.24 kB	1	3 kbps	
208.91.112.52						12.10 kB	41	0 bps	
cs9.wac.phicdn.net (72.21.91.29)						8.34 kB	1	56 bps	
static-doubleclick-net.l.google.com (172.217.9.230)						6.43 kB	1	0 bps	

# Strategie di difesa

- Privilegio minimo (least privilege)
- **Creazione punto di strozzatura (choke point)**
- **Difesa in profondità (defense in depth)**
- Rinforzare l'anello più debole (weakest link)

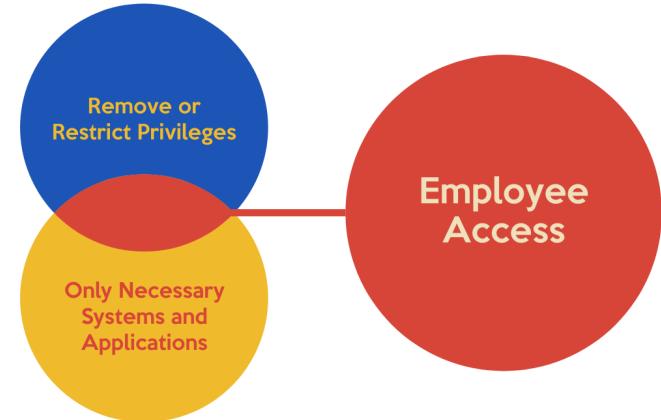


# Privilegio minimo – least privilege

- Strategia basata su principio del “need to know”
- Ogni oggetto (utente, rete remota, amministratore, programma, sistema, ecc...) dovrebbe avere solo i privilegi necessari e sufficienti al suo compito... ma non di più!
- Molti dei problemi relativi alla sicurezza derivano dalla non applicazione di questa regola
- Non è sempre di facile applicazione, specialmente per quanto riguarda gli utenti

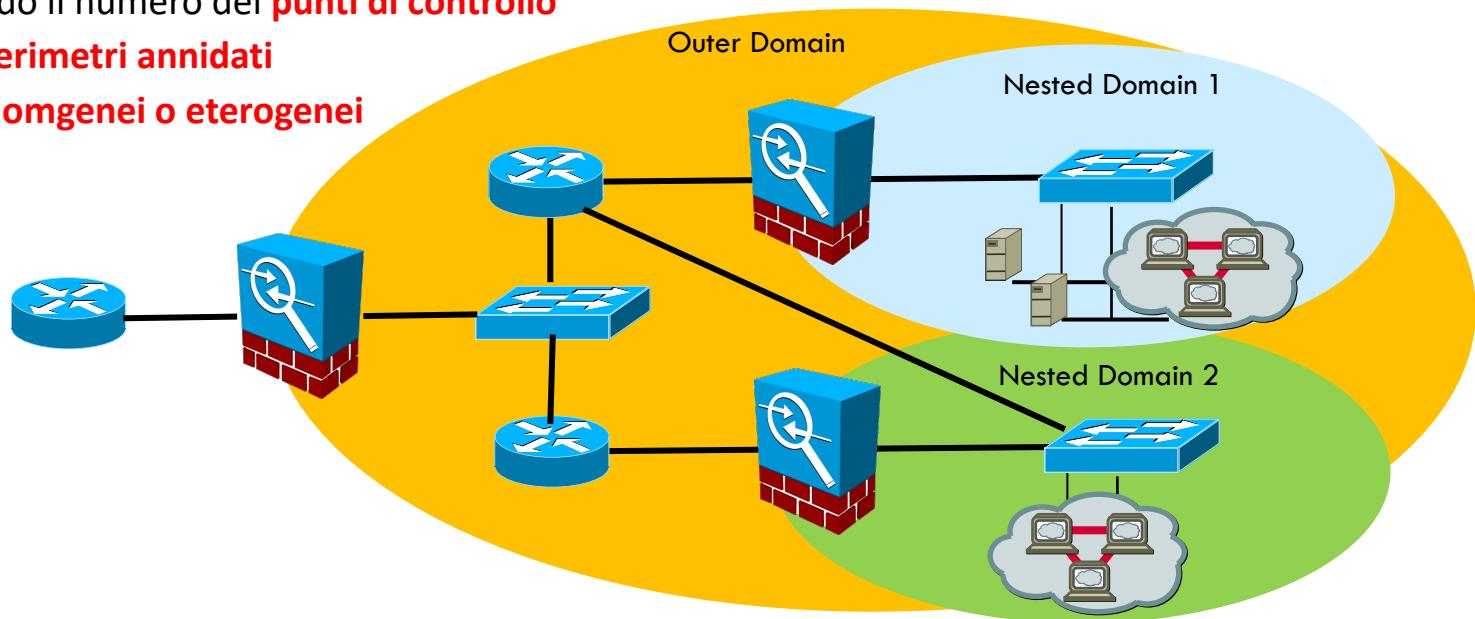
# Privilegio minimo – le basi

- Applicazione massiva di policy di tipo «deny all»
- Information hiding (anche la conoscenza è un privilegio):
  - mascheramento della struttura di sistemi e reti
  - mascheramento delle politiche e dei sistemi di protezione
  - Blocco/riduzione dei meccanismi di esplorazione e di inventory
- Applicabile solo in ambienti «chiusi»



# Difesa in profondità – strategia di base

- Non dipendere da un solo meccanismo di sicurezza, per quanto forte possa sembrare
- Combinare più soluzioni di sicurezza per ridurre la superficie di attacco e proteggere i propri asset critici da multiple angolazioni (**sicurezza multilivello**)
- Introduce il concetto di “**Ridondanza**” nella Security
  - Aumentando il numero dei **punti di controllo**
  - Creando **perimetri annidati**
  - Dispositivi **omogenei o eterogenei**

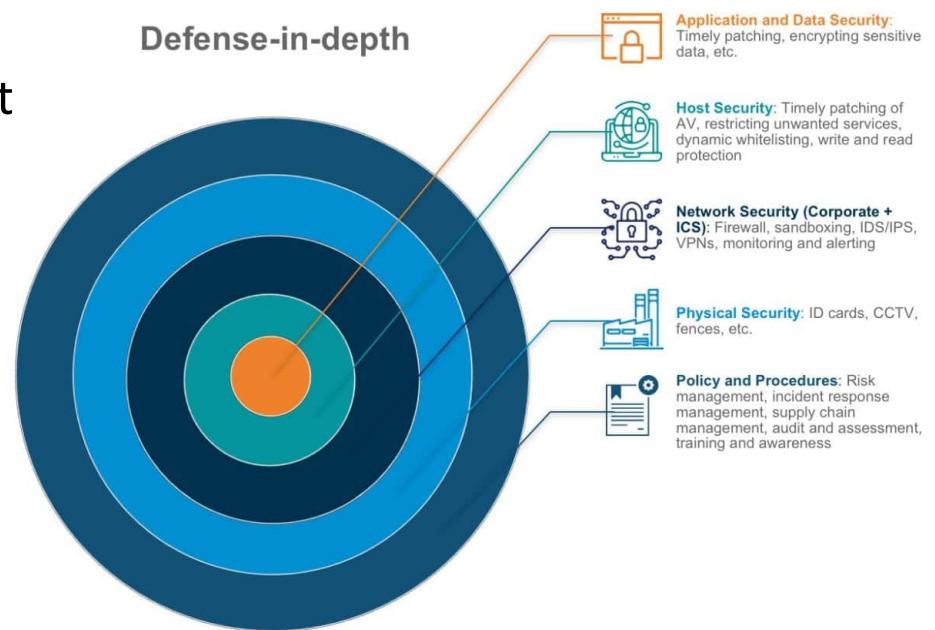


# Difesa in profondità – sicurezza multilivello

- Per assicurarsi sicurezza e disponibilità è necessario creare diversi livelli di protezione :
  - Un approccio a più livelli offre la protezione più completa.
  - Se un livello è compromesso i rimanenti continueranno a garantire la protezione dell'asset
  - La stratificazione crea una catena di punti di difesa che si coordinano per prevenire gli attacchi
    - Tecnologie di difesa a livello applicativo
    - Tecnologie di difesa a livello di sessione
    - Tecnologie di difesa a livello di rete

# Difesa in profondità – defense in depth

- E' possibile combinare elementi omogenei o eterogenei:
  - Firewalls di diversi produttori
  - Strati antivirus/malware
  - Agent di controllo a livello host
  - gateway di autenticazione multifattore
  - sistemi crittografici etc.
  - Misure di sicurezza fisica
  - Procedure di controllo e gestione della sicurezza



# Difesa in profondità – Diversity

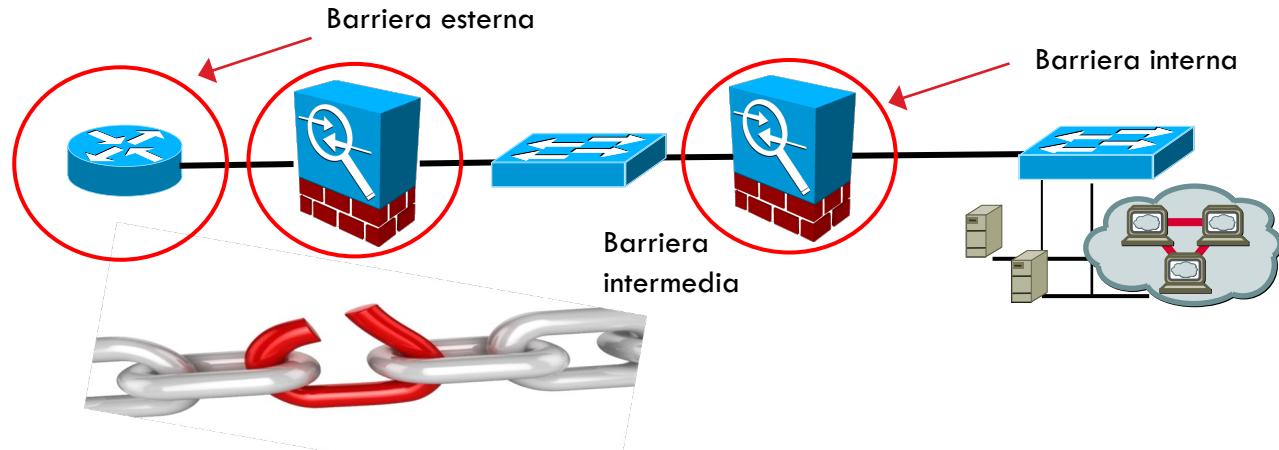
- **Diversity:** variare i controlli e le procedure a diversi livelli.
  - Un'organizzazione può utilizzare diversi algoritmi di crittografia o sistemi di autenticazione per proteggere i dati in diversi stati
  - In una catena di apparati conviene utilizzare soluzioni di vendor deversi
- La complessità non garantisce necessariamente la sicurezza.
  - ... ma riduce drasticamente il rischio
  - Se il processo o la tecnologia usata sono troppo complessi, possono verificarsi errori di configurazione
  - La semplicità può effettivamente migliorare la disponibilità.

# Punto di strozzatura – choke point

- Un choke point forza gli attaccanti ad utilizzare un “canale” di accesso stretto e facilmente controllabile.
  - Riduce la superficie di attacco minimizzandola teoricamente a un solo punto di controllo
  - Rende più semplice la sorveglianza
  - Ma ciò non è sempre facile da realizzare...

# L'anello più debole – weakest link

- In presenza di multipli punti di security enforcement che lavorano in logica annidata (catena di security enforcement)
  - Ricordarsi sempre che “la catena è forte quanto il suo anello più debole”
  - In altre parole: non sottovalutare nessun componente della “catena di sicurezza”
  - Rinforzare l’anello più debole



# Gestione ridondanza

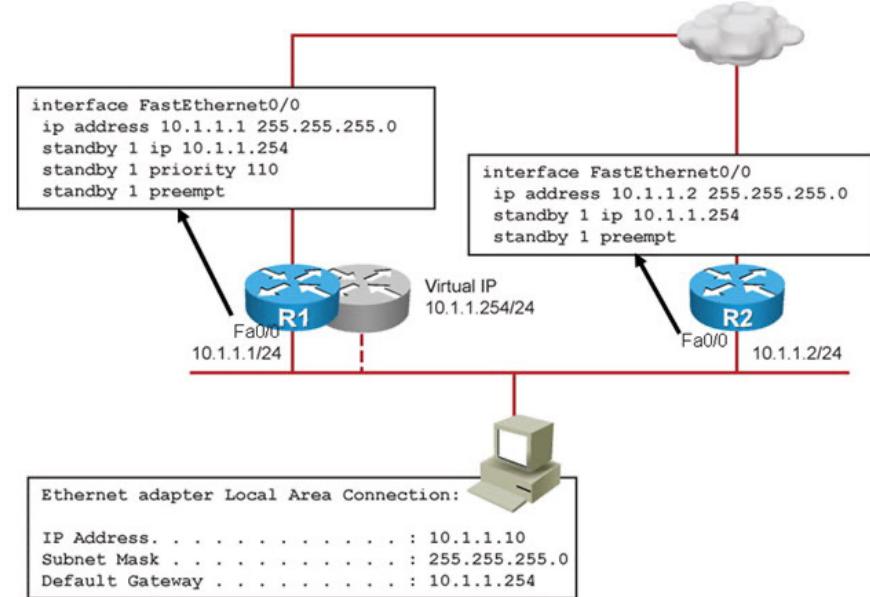
- È necessario identificare tutti i single point of failure (SPOF).
- I SPOF sono gli anelli deboli della catena della sicurezza che possono portare a gravi compromissioni o interruzioni di servizio.
- In presenza di un SPOF lo stesso va ridondato se la sua funzione è critica in modo che la politica di sicurezza non si basi su un singolo elemento.

# First Hop redundancy Protocols

- I protocolli FHRP sono fondamentali nel garantire alta disponibilità ai nostri punti di controllo.
  - I client e i server normalmente puntano a un singolo gateway predefinito e perdono la connettività ad altre sottoreti se il loro gateway non funziona.
  - I FHRP forniscono funzionalità di gateway predefinita ridondante trasparente per gli host finali basati sull'assegnazione di un indirizzo IP virtuale e di un corrispondente indirizzo MAC virtuale.
  - azioni come il failover e il bilanciamento del carico rimangano completamente trasparenti per gli host.
- Esempi di FHRP includono:
  - Hot Standby Router Protocol (HSRP) - Cisco
  - Virtual Router Redundancy Protocol (VRRP) - Standard IETF
  - Gateway Load Balancing Protocol (GLBP) - Cisco

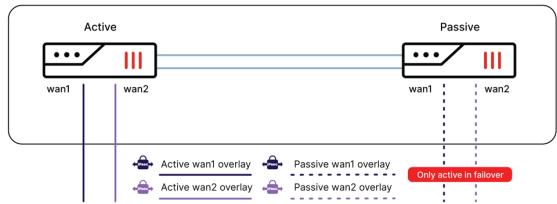
# First Hop redundancy Protocols

- I meccanismi di base sono:
  - Elezione di un singolo router/fw che controlla l'indirizzo IP virtuale
  - Monitoraggio della disponibilità del router/fw attivo (heartbeat)
  - Determinare se il controllo degli indirizzi IP e MAC virtuali deve essere passato a un altro router
  - Gli indirizzi IP e MAC virtuali non sono associati a un dispositivo particolare, ma controllato da un master all'interno di un gruppo di router/fw che partecipano allo schema di ridondanza/bilanciamento



# NG-Firewall HA Cluster

- Esistono protocolli proprietari come FortiGate Clustering Protocol (FGCP) di Firtinet che consentono la negoziazione e creazione di cluster di firewalls in grado di realizzare architetture HA:
  - Un cluster deve contenere almeno 2 firewalls (stesso HW, modello e firmware)
  - Almeno un'interfaccia va dedicata alla sincronizzazione dei firewalls (heartbeat)
  - Un cluster può operare in modalità attivo-attivo oppure attivo-passivo
  - I membri del cluster condividono la medesima configurazione



Master configuration: System> HA> Mode active-active

FortiGate - fortivm1

192.168.30.1/ng/system/ha/cluster

High Availability

Mode: Active-Active

Device priority: 200

Cluster Settings

Group name: HA\_CLUSTER

Password: \*\*\*\*

Session pickup: Enabled

Monitor interfaces: WAN (port1), LAN (port2)

Heartbeat interfaces: HA1 (port3)

Management Interface Reservation

Unicast Heartbeat

OK Cancel

Slave configuration via CLI interface

```
fortivm2 #
fortivm2 # conf sys ha

fortivm2 (ha) # set mode a-a

fortivm2 (ha) # set group-name HA_CLUSTER

fortivm2 (ha) # set password 123456

fortivm2 (ha) # set session-pickup enable

fortivm2 (ha) # set hbdev port3 0

fortivm2 (ha) # set hbdev port4 1

fortivm2 (ha) # end

fortivm2 # slave's external files are not in sync with master, sequence:0. (type CERT_LOCAL)
slave's external files are not in sync with master, sequence:1. (type CERT_LOCAL)
```

# Strategie di difesa: Reattiva e Proattiva

- La difesa **reattiva**, ovvero l'esecuzione di indagini e il controllo dei danni solo a violazione già avvenuta (**controreazione + digital forensics**), è costosa e può comunque comportare perdita di dati, problemi di conformità e danni alla reputazione.
- La difesa **proattiva**, basata sul concetto di prevenzione, cerca di affrontare i potenziali rischi e le vulnerabilità prima che possano essere sfruttati.
- La **segmentazione** è uno dei metodi più comuni attualmente adottati in logica **proattiva** per rispondere a tale esigenza.



# Segmentazione perimeter-based

Il concetto classico di segmentazione come strumento di **difesa proattiva** è fortemente legato a quello di **perimetro**:

- Un dominio viene partizionato in **segmenti interni** ed **esterni** in base all'attendibilità:
  - ciò che è interno è considerato attendibile, ciò che è esterno non lo è.
  - poche restrizioni sulle risorse interne, che comunemente operano su una struttura piatta con una segmentazione interna minima.
- Il filtraggio e la segmentazione avvengono in **punti fissi**
- Per spostarsi tra i segmenti, è necessaria una **policy**

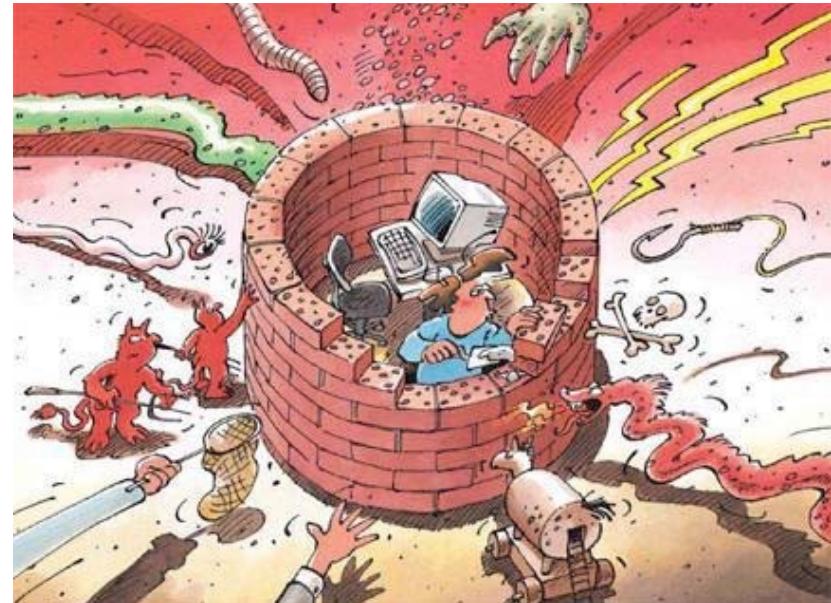
# Segmentazione perimeter-based

Gli strumenti per fare una segmentazione perimeter-based sono:

- I **firewall**: per implementare il controllo del traffico nord-sud (esterno-interno) attraverso le policy
- Le **VLAN** per creare i segmenti
  - Originariamente, le VLAN sono state introdotte per suddividere i domini di broadcast e migliorare le prestazioni della rete.
  - Nel corso del tempo, le VLAN sono state utilizzate sempre di più come strumento di sicurezza, ma non sono mai state concepite per questo.
  - Il problema con le VLAN è che non esiste alcun filtro all'interno della VLAN; il loro livello di accesso è molto ampio.

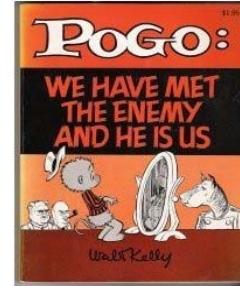
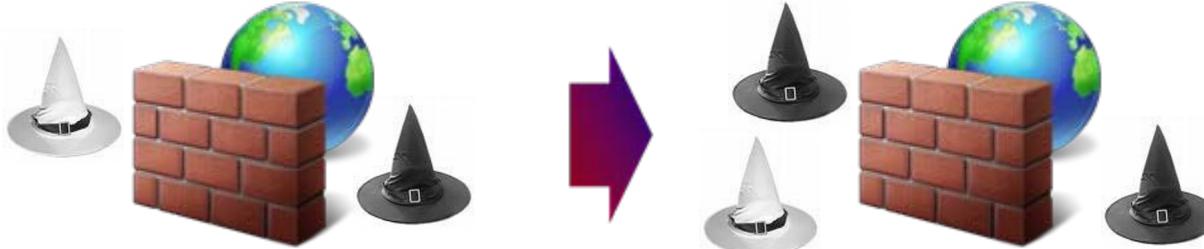
# Rilassamento del perimetro

- Con l'avvento del cloud, del BYOD e della mobilità ubliqua e pervasiva, il **perimetro** diventa **sfumato** e perde **punti di demarcazione** chiari.
- Ciò ha ovvie conseguenze architetturali su tutta la gestione della sicurezza
- E' la fine del **modello fortezza...**



# La borderless architecture

- Diventa impossibile creare barriere perimetrali su cui concentrare gli sforzi e l'attenzione per la security
- Vengono a confondersi ruoli, concetti, direttive di azione e target fondamentali di difesa:
  - Cade completamente il paradigma: tutti “buoni” dentro e tutti cattivi fuori
  - Va riconsiderato il concetto di security domain
  - I meccanismi di segmentazione perimeter-based non bastano più

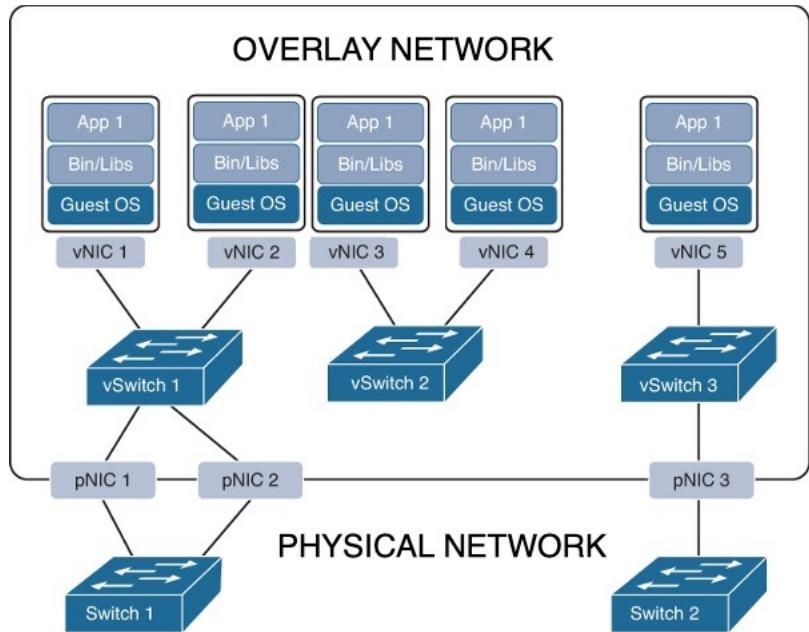


# Evoluzione della segmentazione

- La necessità di controllare il traffico est-ovest e limitare il movimento laterale richiede una segmentazione più **profonda** e **flessibile** della rete
- Entra in gioco la **virtualizzazione della rete**, che porta la segmentazione a un livello superiore
- La segmentazione basata sul perimetro a cui eravamo abituati in passato diventa **virtualizzata** e **distribuita**, insieme a policy di sicurezza flessibili e granulari, che arrivano fino a ogni singolo segmento della rete.

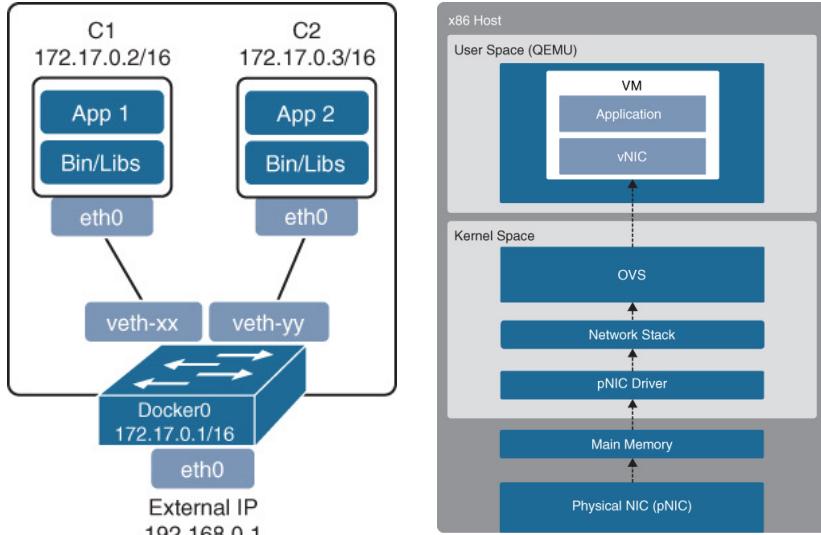
# Virtualizzazione della rete

- La virtualizzazione della rete consente il provisioning di servizi di sicurezza indipendenti dall'infrastruttura fisica
  - Il concetto è quello di creare astrazione dell'HW sotto il controllo del SW.
  - La virtualizzazione trasforma una infrastruttura classica basata su HW in una nuova rete basata su SW.
  - L'infrastruttura fisica sottostante rimane, ma la virtualizzazione della rete consente di utilizzare il software per creare o sovrapporre **reti virtuali** (overlay) sopra quella fisica



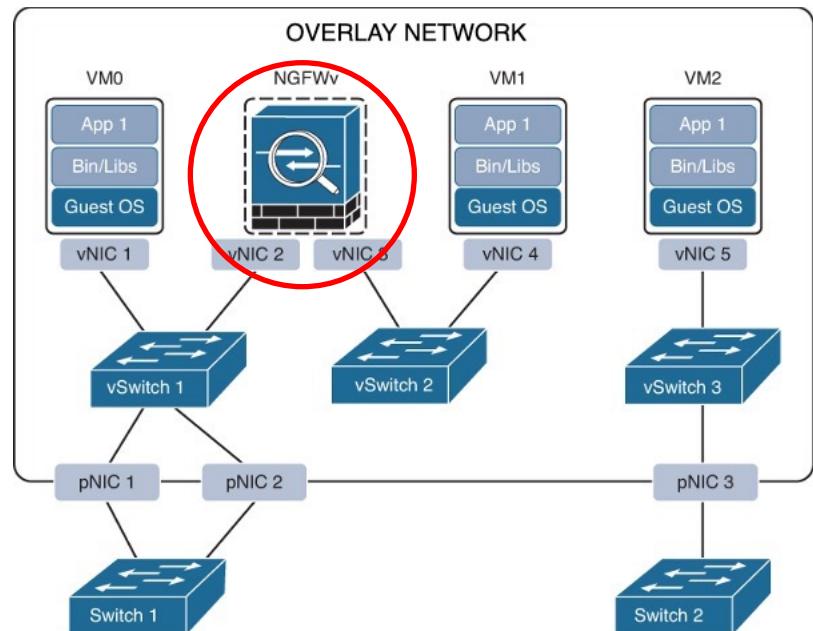
# Virtual Switching

- In un ambiente virtualizzato, possiamo avere interfacce fisiche (pNICs) e virtuali (vNICs) ed un hypervisor che eroga funzioni di switching virtuale fra di loro.
- L'hypervisor ha il compito di trasferire i dati dalle pNIC alle vNIC associate alle VM ed infine alle applicazioni.
- Il modello di riferimento è quello della Open vSwitch (OVS) architecture
- Uno switch virtuale (vSwitch) è uno switch interamente software-based che è in grado di operare come uno switch fisico.
- Consente alle VM di comunicare tra loro all'interno di un server virtualizzato (attraverso vNICs) e con reti fisiche esterne (attraverso pNICs).



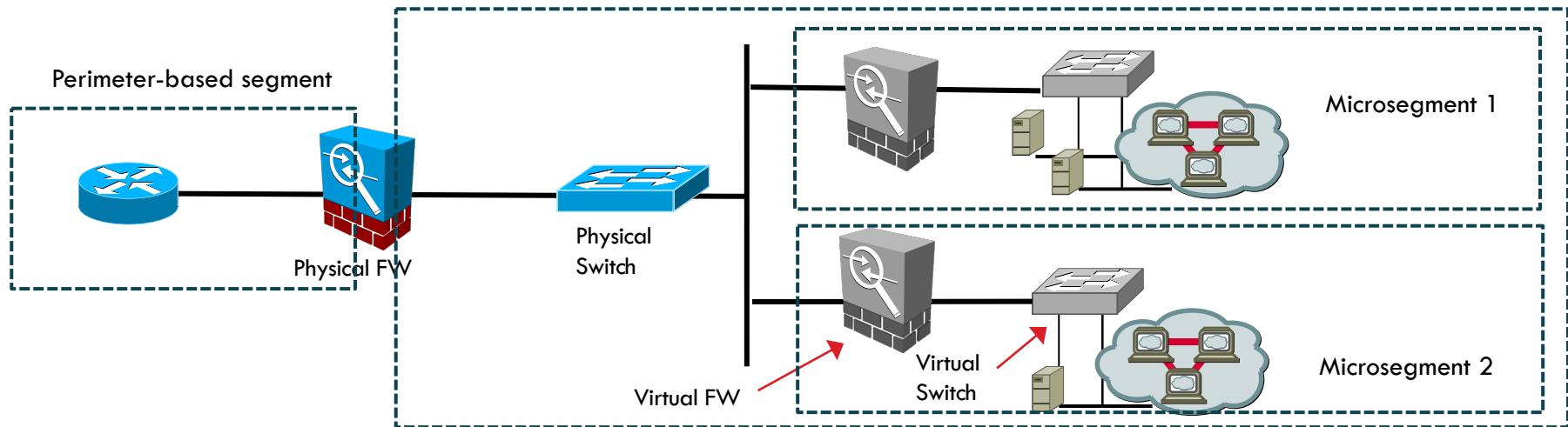
# Network Function Virtualization (NFV)

- La **virtualizzazione delle funzioni di rete (NFV)** definisce gli standard per disaccoppiare le funzioni di rete da hardware proprietario spostandoli su macchine virtuali, container, etc.
- Definisce inoltre come gestire e orchestrare propriamente le funzioni di rete, ciascuna si riferita alla funzione eseguita da un dispositivo fisico, come un firewall o una funzione router
- Diventa possibile “delocalizzare” le funzioni principali di rete:
  - quelle classiche tradizionalmente centralizzate: Routing, Firewall, Proxy, DHCP
  - altre più specifiche: Load Balancer, IDS, IPS, Logging, Monitor, Flow analysis
- Nasce il concetto di Virtual FW



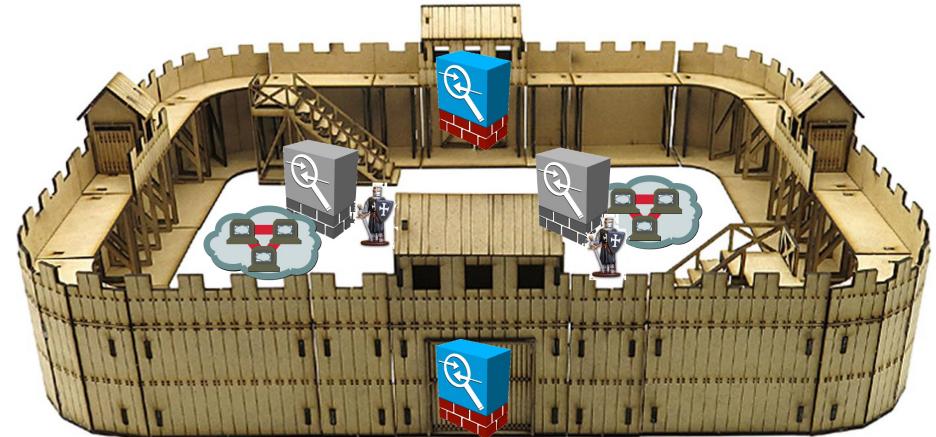
# Microsegmentazione

- A differenza di una architettura tradizionalmente segmentata, la **microsegmentazione** è un approccio software che protegge le singole risorse con **microperimetri di controllo** che bloccano le comunicazioni e il traffico non necessario o legittimo.
- Più veloce e più facile da gestire rispetto ai firewall interni e alle VLAN.
- Separando i controlli di sicurezza dall'infrastruttura sottostante, consente alle organizzazioni di estendere la protezione ovunque.



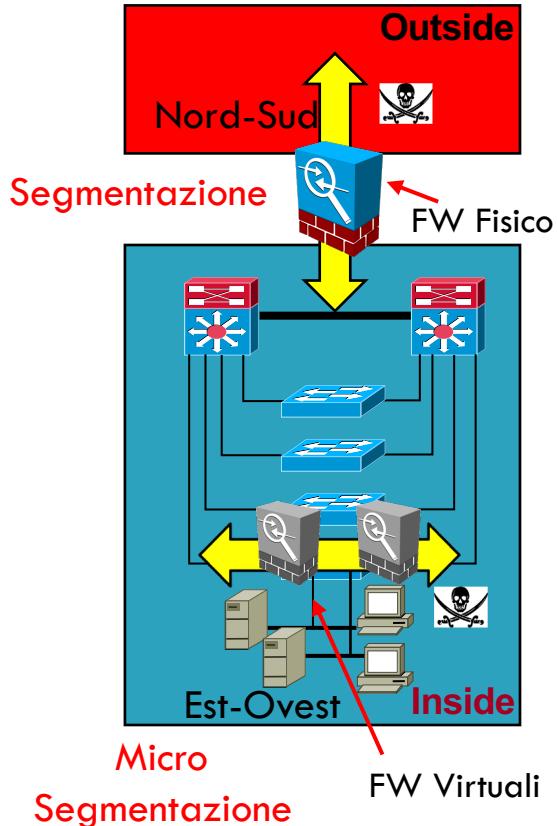
# Microsegmentazione

- La **segmentazione** è più indicata per il traffico **nord-sud**
- La **microsegmentazione** aggiunge un livello di protezione per il traffico **est-ovest** (server-server, app-server, web-server etc.).
- Nell'analogia del modello fortezza la segmentazione della rete si paragona alle mura **esterne**, mentre la microsegmentazione alle **guardie** che controllano le porte delle sale interne.



# Segmentazione e isolamento

- La segmentazione è in grado di bloccare il **l'intrusione a livello di minacce esterne**:
  - in una rete segmentata, una violazione in un dato segmento non rappresenta una minaccia immediata per un altro segmento.
- La microsegmentazione blocca il **movimento laterale delle minacce interne**:
  - segmentare l'accesso in base alle esigenze interne (ad esempio rendendo i dati finanziari inaccessibili alle risorse umane) riduce il rischio di attacchi interni.



# Vantaggi della segmentazione

- Sicurezza più semplice ed efficace:
  - prevenzione delle violazioni (movimento nord-sud)
  - controllo più rigoroso degli accessi
  - controlli di sicurezza specifici per ogni segmento (più semplice garantire rispetto policy e normative)
- Facilitazione dell'analisi del rischio e della gestione dei danni:
  - quando l'avversario non è in grado di muoversi liberamente è più facile individuare le tecniche di attacco potenziali e identificare i punti deboli nella sicurezza e le vulnerabilità.
- Incremento della sicurezza di endpoint e utenti:
  - se le minacce non possono diffondersi con facilità tra i vari segmenti, gli utenti finali e gli endpoint saranno più sicuri, e i segmenti stessi godranno di protezione maggiore dalle minacce che vengono dagli endpoint.
- Riduzione effetti di congestione e DoS:
  - l'attività eccessiva di un segmento non limita quella degli altri.

# Svantaggi della segmentazione

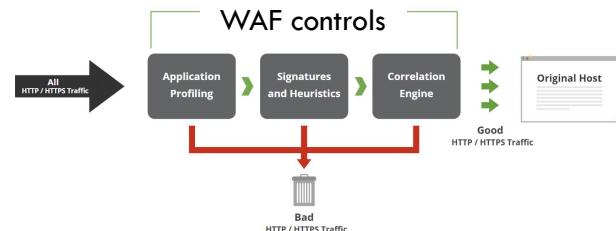
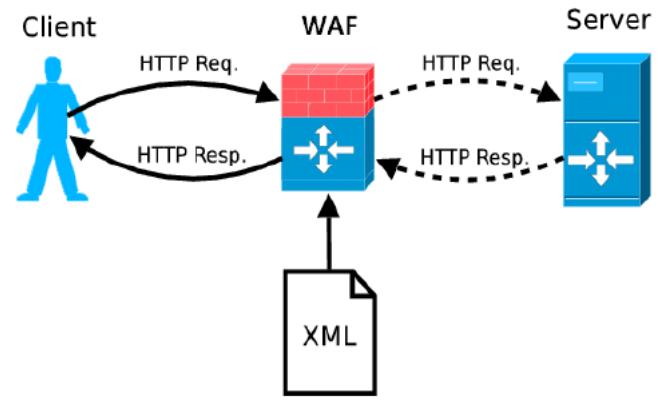
- Eccesso di fiducia:
  - dato che la segmentazione tradizionale basata su firewall è progettata per prevenire gli attacchi esterni, l'infrastruttura resta comunque vulnerabile alle minacce interne.
- Gestione più complessa:
  - l'introduzione di nuovi dispositivi virtuali e non, implica una attività di configurazione complessa, che può riflettersi su qualsiasi modifica.
  - è più complesso definire le policy di segmentazione per i vari servizi e categorie di utenti
- Problemi di scalabilità:
  - per gestire la crescita è necessario creare segmenti più piccoli o aggiornare quelli esistenti, con il conseguente incremento dei costi per la scalabilità e la manutenzione.
- Impatto prestazionale:
  - l'aggiunta di altri dispositivi di rete virtuali e non (ad esempio, firewall e router) può avere un effetto negativo che danneggia le prestazioni complessive

# Segmentazione: buone pratiche

- Non effettuare una segmentazione eccessiva
  - Se da un lato una segmentazione eccessiva può ridurre la visibilità complessiva sulla rete e renderne difficile la gestione, dall'altro una segmentazione insufficiente lascia che la superficie di attacco rimanga estesa e compromette il livello di sicurezza.
- Eseguire verifiche periodiche
  - La segmentazione della rete migliorerà la sicurezza solo se vengono effettuati audit continuativi dei segmenti, per individuare le vulnerabilità
- Applicare il principio dei privilegi minimi
  - Solo applicando il principio dei privilegi minimi a tutti i segmenti è possibile garantire a tutti che l'accesso verrà consentito solo se necessario.
- Limitare l'accesso di terzi
  - Concedere l'accesso a terzi è rischioso, ed è importante farlo solo se necessario, soprattutto se viene concesso per più segmenti.
  - Valutare attentamente le nuove autorizzazioni che vengono concesse risulta fondamentale per preservare un buon livello di sicurezza della rete.

# Web Application Firewall

- Un FW software dedicato a proteggere le applicazioni web
- Fa da filtro tra l'applicazione web e Internet:
  - monitorando e controllando il traffico HTTP/S in e out
  - Effettuando verifiche sull'esecuzione di qualsiasi script (es. Javascript) e verifiche CAPTHCA
  - fornendo un livello aggiuntivo di difesa contro minacce quali injection di SQL, cross-site scripting (XSS), DoS e vulnerabilità di applicazioni web.
  - consente di creare e modificare le regole di filtraggio per personalizzare la protezione in base alle esigenze specifiche dell'applicazione web.
- Il WAF non è esplicitamente un oggetto della ZTA ma la combinazione di ZTA e WAF crea una difesa a più livelli:
  - La ZTA garantisce che solo gli utenti e i dispositivi autorizzati possano accedere alle applicazioni (riduzione superficie di attacco)
  - il WAF protegge le applicazioni da contenuti dannosi all'interno del traffico consentito, filtrando i vettori di attacco noti



# Architetture classiche di controllo accessi

- Tutte le architetture classiche di controllo accessi si basano sulla **segmentazione** perimeter-based e sul modello fortezza
- I più comuni modelli-layout architetturali sono:
  - Schemi single-box, essenzialmente basati sulla strategia choke point
    - Dual Homed Host
    - Screening Router
  - Schemi multibox, basati sulla logica di difesa in profondità:
    - Screened Host Architecture
    - Screened Subnet Architecture

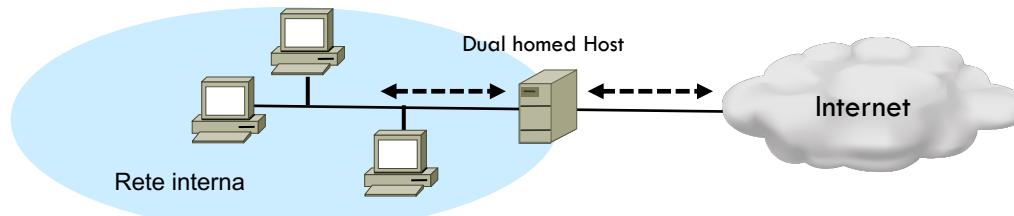
# Architetture Single-box

- Modellano il concetto di “bastione” nello schema “fortezza”
- Vantaggi:
  - facilità di configurazione (ma ovviamente richiedono ugualmente un'attenta pianificazione);
  - economicità
- Svantaggi:
  - tutta la sicurezza dipende da un solo punto di difesa (single point of failure)
  - funzionalità ridotte
  - Limitata scalabilità



# Dual-homed Host

- Host dotato di due schede di rete collegate a segmenti di rete differenti
- IP diverso su ciascuna interfaccia. L'instradamento è disabilitato evitando che i pacchetti IP vengano direttamente instradati da una rete all'altra
  - la comunicazione tra interno ed esterno avviene esclusivamente con la mediazione del dual-homed host.
  - Il livello di controllo è molto alto, a costo, ovviamente, delle prestazioni
- unico punto di contatto tra due domini che non si parlano a livello di rete
  - Utenti posti su domini distinte possono usufruire di una applicazione posta sul dual-homed host stesso
  - Se tale applicazione lo consente, le reti possono anche condividere dati.
  - I due domini possono essere interconnessi in modalità proxy
- Tutto ciò accade senza che vi sia un effettivo interscambio di pacchetti tra rete esterna e interna

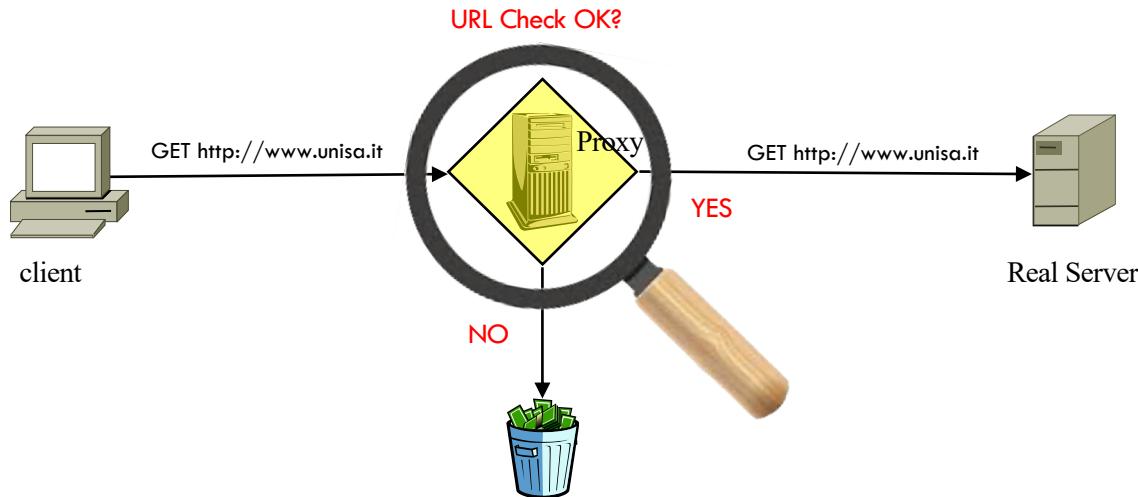


# Svantaggi dual-homed host

- Un dual homed host è molto probabilmente meno sicuro in quanto maggiormente vulnerabile di un router o di un firewall (ad esempio agli attacchi DoS).
- Per fornire i servizi desiderati all'interno è necessario:
  - permettere agli utenti di collegarsi alla macchina
    - assolutamente sconsigliato per motivi di sicurezza)
  - far ricorso al proxying
    - non è sempre possibile
- Inoltre, se si vogliono fornire dei servizi all'esterno:
  - devono girare localmente sul dual homed host
    - altamente sconsigliabile, visto che si tratta dell'unica difesa disponibile.

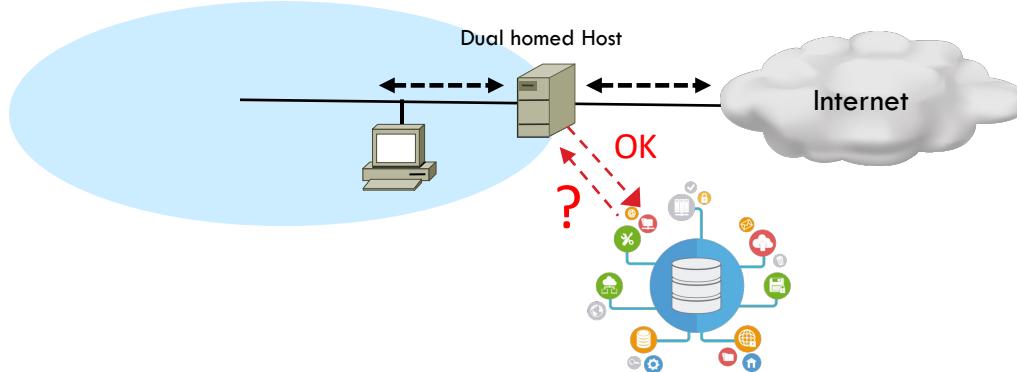
# Proxy

- Ogni richiesta fatta da un client sulla rete interna viene valutata dal proxy e, se rispetta certe regole, viene rigirata al server, altrimenti viene scartata.
- La risposta del server reale viene inviata al proxy server che la rigira al client.
- Un solo host collegato ad Internet ma che faccia da tramite tra gli host interni ed il mondo esterno in modo (quasi) trasparente può essere di grande aiuto sia alla sicurezza che alle prestazioni



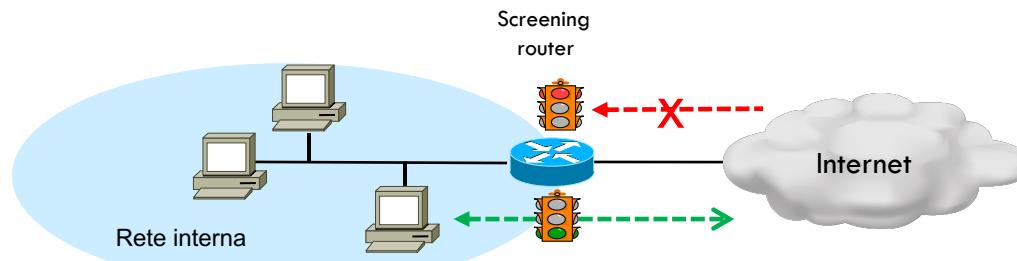
# Vantaggi e svantaggi dei Proxy

- Permettono un ottimo servizio di logging
- Consentono meccanismi di content filtering
- Ritardo tra l'introduzione di nuovi servizi Internet e la disponibilità del servizio proxy relativo
- I servizi proxy possono richiedere diversi server per ogni servizio
- Alcuni servizi Internet non sono gestibili da proxy
- I servizi proxy non proteggono dalle debolezze dei protocolli



# Architettura Screening Router

- Lo screening router consente a host esterni di comunicare con la rete interna in maniera controllata
  - Bloccando o autorizzando il passaggio dei pacchetti in ragione delle specifiche policy di sicurezza implementate
  - E' l'architettura di protezione più comune ed economica in cui la sicurezza del dominio si basa su un solo componente perimetrale
  - le possibilità di filtraggio sono alquanto rigide
  - soluzione adatta se la rete da proteggere ha già un buon grado di sicurezza
- E' un choke point, permettendo di centralizzare i controlli sotto screening router
  - Una volta compromesso lo screening router salta irrimediabilmente la sicurezza dell'intero dominio

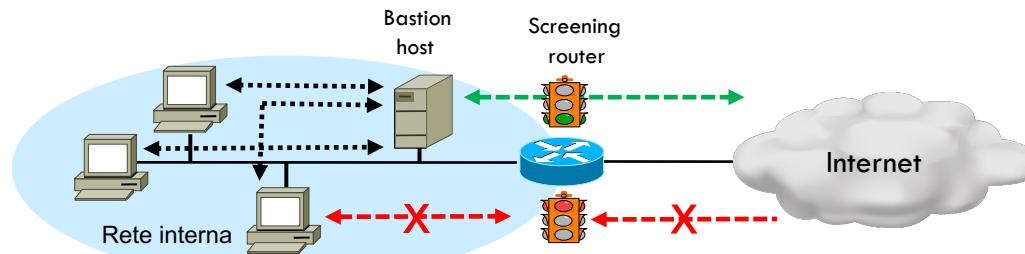


# Architetture Multi-box

- Basate sul concetto di difesa in profondità, prevedono punti di controllo multipli annidati che operano in logica cooperativa
- Vantaggi:
  - Ridondanza nel controllo
  - Possibilità di implementare architetture più sofisticate ed efficaci
- Svantaggi:
  - Aumento della complessità
  - Aumento dei costi

# Architettura Screened Host

- Lo screening router permette agli host esterni ed interni di comunicare solo con uno specifico host della rete reso particolarmente sicuro, in modo tale da poter essere esposto, il **Bastion Host**
- Il Bastion host supporta una singola interfaccia di rete ed è l'unico nodo della rete interna in grado di:
  - Fare traffico verso la rete esterna
  - Ricevere traffico dalla rete esterna
- Gli host interni vedono la rete esterna solo attraverso servizi proxy offerti dal bastion host che è in grado di realizzare meccanismi di caching



# Il ruolo del bastion host

- In questo tipo di architettura la sicurezza viene garantita dal packet filtering (per impedire alle macchine interne di uscire direttamente) gestito a livello di router
  - Il bastion host è l'unica macchina visibile all'esterno (deve essere adeguatamente protetto)
  - Il router limita le possibilità di connessione del bastion host ai casi ritenuti necessari.
  - Il router, può consentire alle macchine interne di collegarsi direttamente ad Internet per alcuni servizi o di bloccarle, obbligandole quindi ad utilizzare i servizi di proxy sul bastion host.
- Dato che è più facile difendere un router che non un dual-homed host, questa architettura è più sicura di quella dual-homed.
  - Ad ogni modo il router è un single point of failure, come pure il bastion host.
- L'uso appropriato di questo tipo di architettura è quando la sicurezza delle macchine interne è buona e le connessioni dall'esterno sono poche
  - **non** quando il bastion host è un web server

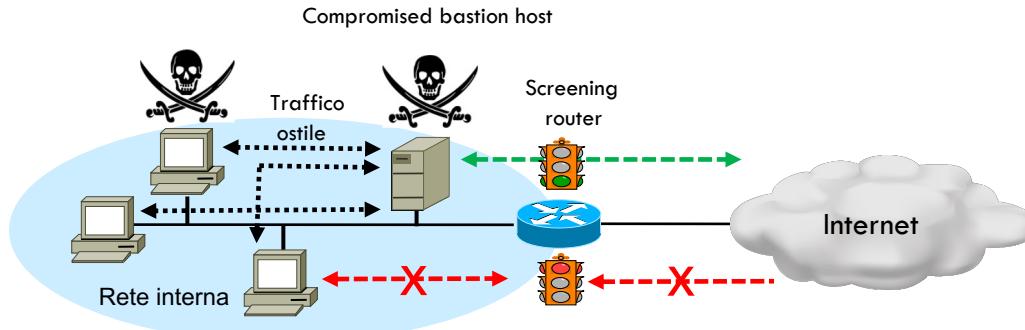
# Servizi forniti dal Bastion Host

Il Bastion Host (BH) deve fornire solo i servizi necessari per accedere ad Internet e quelli che devono essere offerti all'esterno, **e niente altro**

- Servizi proxy
  - Il BH può garantire la fruizione in modalità proxy di servizi che prevedono connettività esterna a macchine della rete interna:
    - PROXY HTTP, FTP, SOCKS et.
    - Nell'erogare tali servizi può realizzare meccanismi di **content filtering**
- Servizi che possono essere resi sicuri
  - offerti dal BH al mondo esterno: *SMTP, HTTP, NNTP, DNS*
- Servizi intrinsecamente insicuri
  - devono essere disabilitati o forniti da un host vittima (honeypot)
- Servizi non utilizzati (almeno non in relazione ad Internet)
  - devono essere disabilitati

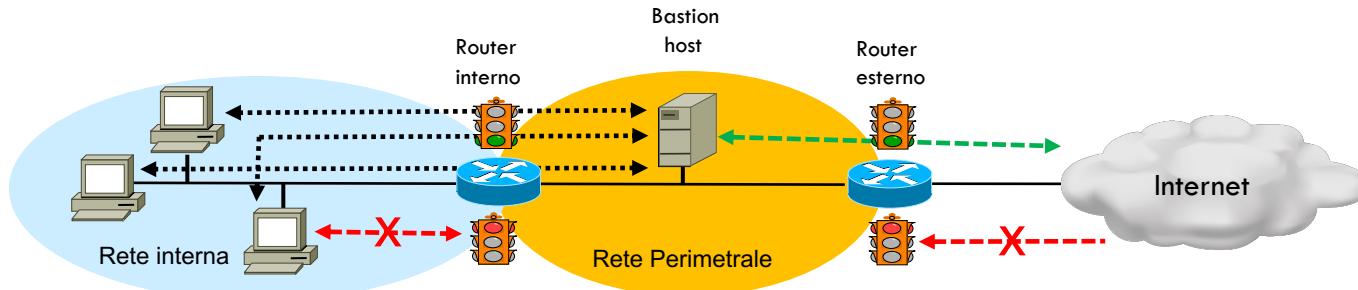
# Vulnerabilità Screened Host

- I bastion host sono le macchine più vulnerabili della LAN, poichè sono le prime a subire l'attacco di un intruso
- In una architettura screened host non ci sono difese tra il bastion host e la rete interna: se si riesce ad attaccare il bastion host si riesce a entrare nella rete interna
- Ciò rende necessario aumentare i punti di controllo e filtraggio in accordo a meccanismi di incremento della profondità dell'architettura di sicurezza



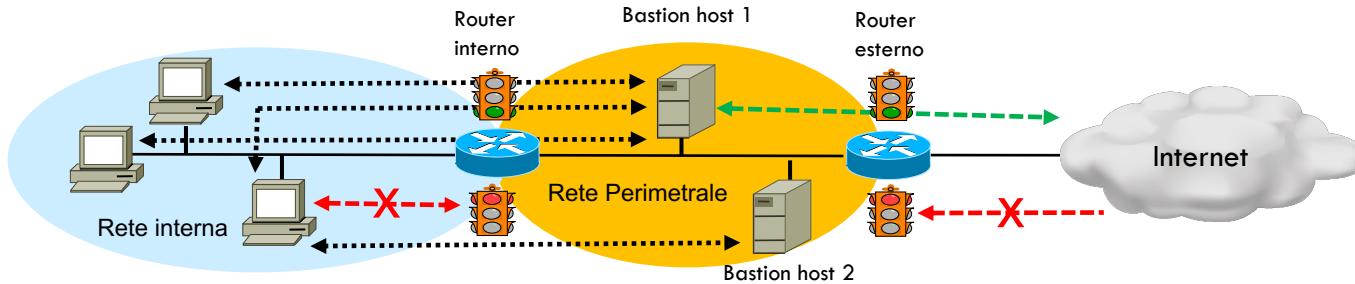
# Screened Subnet

- Aggiunge un livello extra di sicurezza all'architettura screened host (logica in profondità) creando una rete, nota come rete perimetrale, che:
  - isola la rete interna da Internet
  - risolve i problemi associati alla compromissione del router o del BH
- Due screening routers (interno ed esterno) sono presenti:
  - Uno situato tra la rete perimetrale e la rete interna
  - Un altro situato tra la rete perimetrale e la rete esterna
- Abbiamo due punti differenti di enforcing delle politiche di sicurezza
- Per compromettere il dominio vanno compromessi entrambi i router



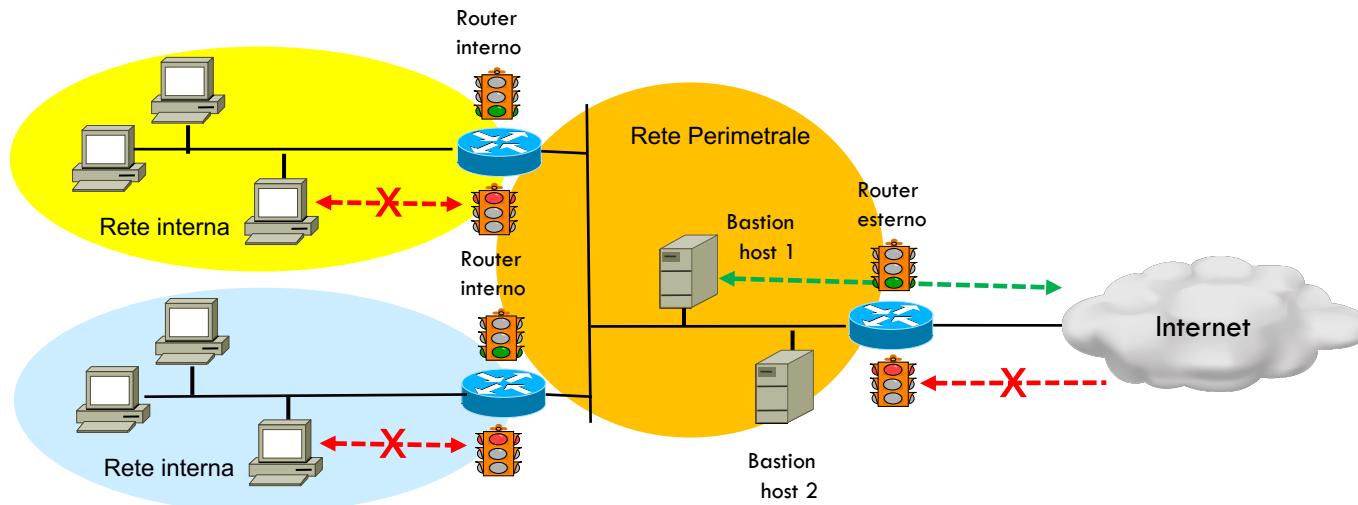
# Screened Subnet

- Anche i Bastion Host possono essere ridondati
- In linea di principio una soluzione ideale sarebbe una macchina per ogni servizio.
- La rete perimetrale spesso contiene al suo interno almeno due application level gateway che svolgono la funzione di bastion host pur conservando l'accesso interattivo ai servizi resi condivisibili tra entrambe le reti
- Risulta molto difficoltoso da parte di un intruso bypassare l'intera sottorete perimetrale, in quanto dovrebbe eludere il filtraggio dei router e la sorveglianza dei bastion host



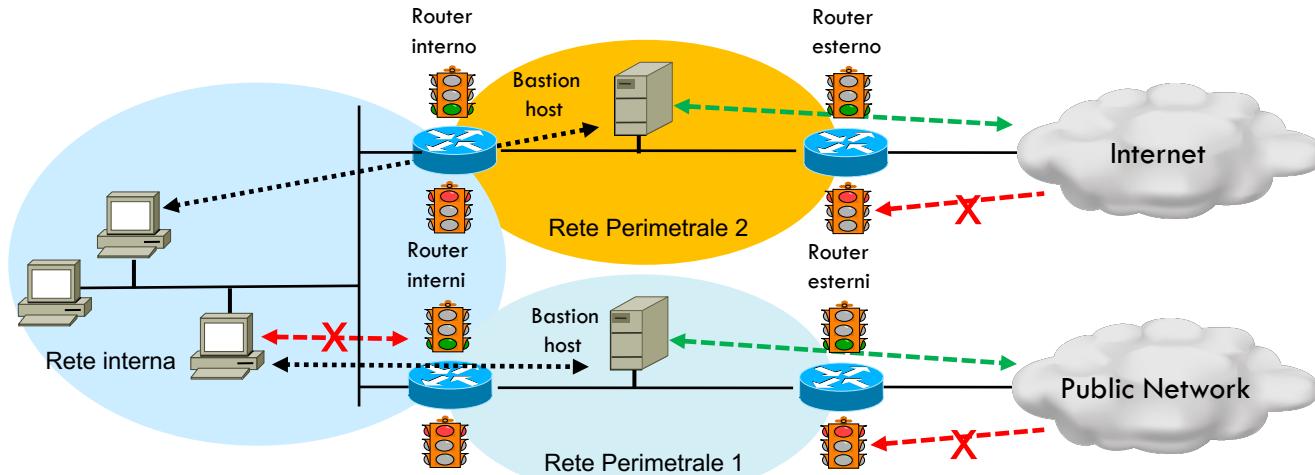
# Screened Subnet

- In presenza di multiple reti interne è possibile usare la rete perimetrale quale dorsale di collegamento fra le differenti reti interne
- Ogni rete ha il suo screening router che la isola dall'esterno



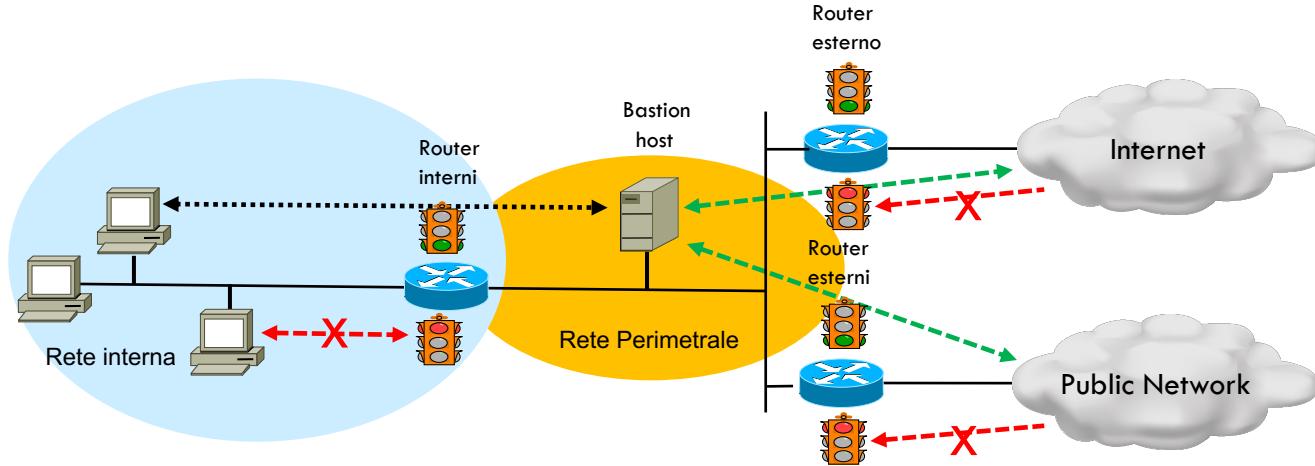
# Screened Subnet

- L'intera architettura può essere replicata creando multiple reti perimetrali ad isolamento del confine con domini di sicurezza differenti
- Ogni rete perimetrалe avrà i suoi screening routers e bastion hosts



# Screened Subnet

- E' anche possibile condividere una singola rete perimetrale con funzioni di dorsale per due reti esterne, ciascuna con il suo screening router esterno
- Lo screening router interno sarà unico e agganciato alla dorsale



# Nuove architetture di controllo accessi

- La fine del modello fortezza e quindi del concetto di «inside trust» richiede la creazione di nuove architetture di controllo accessi
- Il concetto di base è che nessun utente o dispositivo è intrinsecamente attendibile di per sè o in base alla sua posizione
  - Gli approcci basati sul presupposto che:
    - esista un “dentro” e un “fuori” rispetto a un ben definito “perimetro aziendale”,
    - il traffico su cui fare attenzione sia solo quello “nord – sud”, che attraversa il perimetro,
    - le minacce siano “esterne” mentre tutto ciò che è “interno” possa essere considerato “affidabile” sono quindi ormai da considerarsi inefficaci, superati e, per certi versi, anche pericolosi.
- Le politiche di controllo accessi si devono invece basare sul principio dei privilegi minimi indipendentemente da tutto
- Nasce la **Zero Trust Architecture** (ZTA)

# Zero Trust Architecture

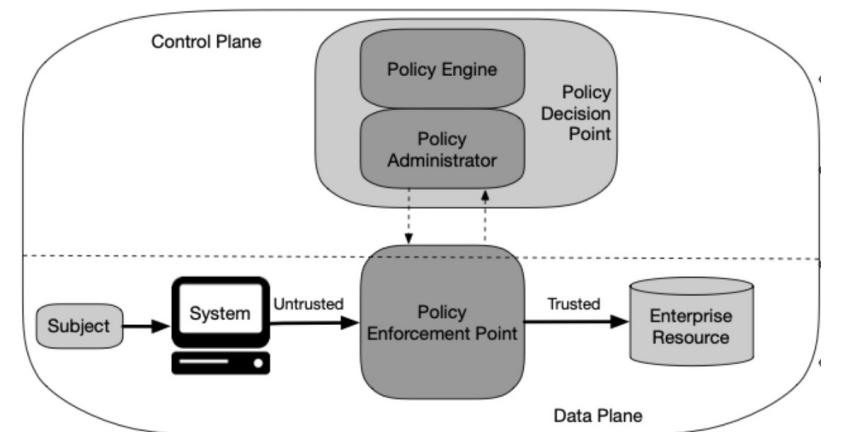
- In un'architettura zero trust, la posizione di rete di una risorsa non è più l'elemento principale del suo profilo di sicurezza
- Un'architettura zero trust applica le policy di accesso in base al contesto:
  - il ruolo e la posizione dell'utente,
  - il dispositivo che utilizza e l'applicazione interessata
  - i dati che richiede (il contenuto delle transazioni)
    - Lo zero trust richiede la visibilità e il controllo completo sugli utenti e sul traffico inviato e ricevuto, compreso quello cifrato
- Il concetto di “fiducia” si sposta verso un nuovo modello basato:
  - sul principio “**never trust, always verify**”
  - su policy che comportino l'applicazione dei controlli “**who, what, when, where, why and how**” a tutto il traffico che interessa il singolo elemento

# Zero Trust Architecture

- Lo schema ZTA collega gli utenti direttamente alle applicazioni su base uno a uno, senza mai legare la loro sicurezza alla loro posizione sulla rete
  - Questo elimina di conseguenza il movimento laterale
  - Inoltre consente di ottenere la segmentazione in un modo radicalmente diverso e più efficace, che risulterebbe impossibile con le VLAN e i firewall tradizionali
  - Invece di una segmentazione della rete rigida, dati, flussi di lavoro, servizi e altro sono protetti da una microsegmentazione definita a livello software, che consente di mantenerli al sicuro ovunque
- il focus della sicurezza si sposta dal “perimetro esterno”, ormai obsoleto, al “**micro-perimetro**” della singola risorsa da proteggere,

# Zero Trust Architecture

- Ogni sorgente di dati, applicazione o servizio è considerata una risorsa
- L' accesso alle risorse è autorizzato da un **policy decision point (PDP)** e garantito attraverso un **policy enforcement point (PEP)**
- Il PDP è strutturato in due diverse componenti:
  - il **policy engine (PE)** il cui compito è prendere e tracciare le decisioni
  - il **policy administrator (PA)** che attua le policy gestendo l'autenticazione e istruendo il PEP a stabilire/negare la comunicazione con la risorsa
- Il **PEP**: ha il ruolo di stabilire, negare, monitorare e terminare le connessioni verso le risorse. E' strutturato in due componenti:
  - Un **agent** che lavora lato client
  - Un **gateway** o **single portal** che opera lato risorsa
- PE, PA e PEP comunicano su una rete logicamente separata e non direttamente accessibile dagli utenti e dalle risorse (**control plane**). Il **data plane** è usato per il traffico delle applicazioni



# Zero Trust Architecture

Esistono tre distinti schemi di implementazione:

- **Device Agent/Gateway-Based**

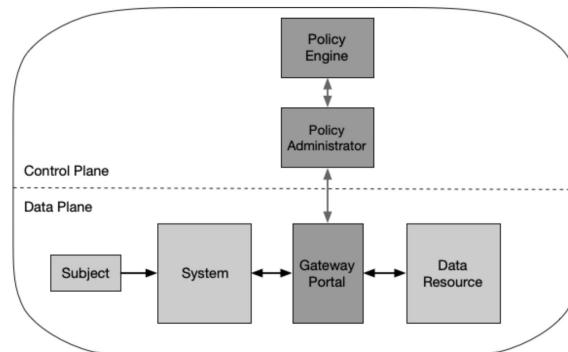
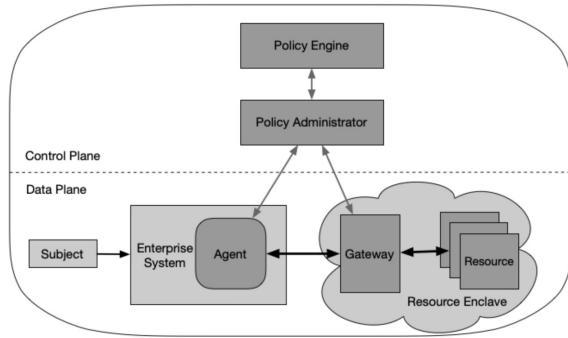
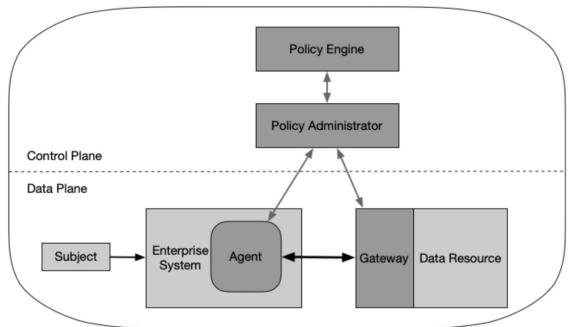
- il PEP è diviso in due componenti che risiedono sulla risorsa o operano come front-end della stessa
  - L'agent è un componente SW che indirizza il traffico al PEP appropriato affinché le richieste possano essere valutate.
  - Il gateway è responsabile della comunicazione con il PA e della conseguente autorizzazione delle sessioni

- **Enclave-Based**

- i componenti agent e gateway non risiedono sopra o davanti alle risorse singole ma al confine di un'insieme di risorse gestite

- **Resource Portal-Based**

- il PEP è un singolo componente che funge da portale/gateway e non esiste un agente locale che gestisca le richieste.
- Può servire una risorsa individuale o un'insieme di risorse
  - non è necessario alcun componente software installato sui client.
  - più flessibile per le politiche BYOD



# Vantaggi della ZTA

Rispetto alla segmentazione tradizionale, l'architettura ZTA:

- Fornisce un accesso adattativo, sensibile all'identità e preciso, svincolato dall'accesso alla rete, ed elimina l'attendibilità implicita, sostituendola con quella esplicita basata sull'identità.
- Le applicazioni interne (e gli indirizzi IP) non vengono mai esposte a Internet, riducendo di conseguenza la superficie di attacco e i rischi.
- Offre una segmentazione tra utente e app tramite policy di accesso granulari applicate su entità virtualizzate direttamente associate agli oggetti da proteggere, anziché richiedere la configurazione di policy di accesso e regole firewall.
- Migliora la flessibilità, l'agilità e la scalabilità, riducendo al contempo la necessità di ricorrere a firewall interni.
- può essere realizzata come servizio in cloud o come software gestito on-premise.
- Consente l'accesso sicuro alle applicazioni per i dispositivi non gestiti e i partner esterni, mantenendo gli utenti fuori dalla rete e riducendo al minimo il rischio di proliferazione dei malware.

# Network Access Control

- Il **Network Access Control** (NAC) è un approccio avanzato alla sicurezza di una rete che mira ad unificare:
  - le tecnologie di sicurezza operanti a livello endpoint (antivirus, host-based intrusion detection, assessment automatico delle vulnerabilità)
  - i meccanismi di autenticazione a livello utente o sistema
  - i meccanismi tradizionali per il controllo della sicurezza della rete
- **Obiettivo:** controllare la sicurezza degli endpoint uniformandone il grado di sicurezza con quella dei dispositivi di rete e in generale dell'intera rete
- **Risultato:** i dispositivi terminali non conformi ai criteri di sicurezza impostati vengono identificati e messi in quarantena

# NAC: Motivazioni

- La principale motivazione è stata la consapevolezza che dopo aver speso miliardi sul perimetro, non si era raggiunto il grado di sicurezza desiderato a causa delle **minacce interne**
- Gli endpoint che non rispettano le politiche di sicurezza stabilite rappresentano una minaccia e possono introdurre un rischio per la sicurezza nella rete.
- Diventa fondamentale impedire agli host vulnerabili e non conformi alle politiche di sicurezza previste quali requisito minimo di ottenere l'accesso alla rete

# NAC: Principali Funzionalità

- Mitigazione di attacchi noti (non zero-day)
  - Per impedire alle stazioni terminali prive di antivirus, patch o software di intrusion detection di accedere alla rete e mettere altri computer a rischio di contaminazione incrociata
- Applicazione di politiche di ammissione in rete
  - Consentire agli amministratori di definire criteri, come i tipi di computer o i ruoli degli utenti autorizzati ad accedere a specifiche aree della rete e garantendone l'enforcement a livello di switch, router e firewall
- Introduzione di meccanismi di controllo accessi identity-based
  - Invece di utilizzare I soli indirizzi IP, il NAC condiziona l'accesso alla rete anche in base alle identità degli utenti opportunamente autenticati

# NAC: Concetti di Base

- Enforcement di controlli **Pre-ammissione** o **Post-ammissione**
  - nel primo casi i dispositivi sono ispezionati prima essere autorizzati ad accedere alla rete
  - In alternativa, NAC post-ammissione prende le decisioni in base alle azioni dell'utente, dopo che gli utenti hanno avuto accesso alla rete
- Raccolta dati **Agent-based** o **Agentless**
  - Un agent SW gira su ogni endpoint
    - L'agent ne analizza e riporta lo stato in termini di sicurezza
  - Alcuni dispositivi non supportano agenti SW (stampanti, scanners etc.)
    - Vengono usate tecniche opportune di scansione e network inventory management (whitelisting, blacklisting, ACLs) per mutuare da remoto le caratteristiche di sicurezza del dispositivo

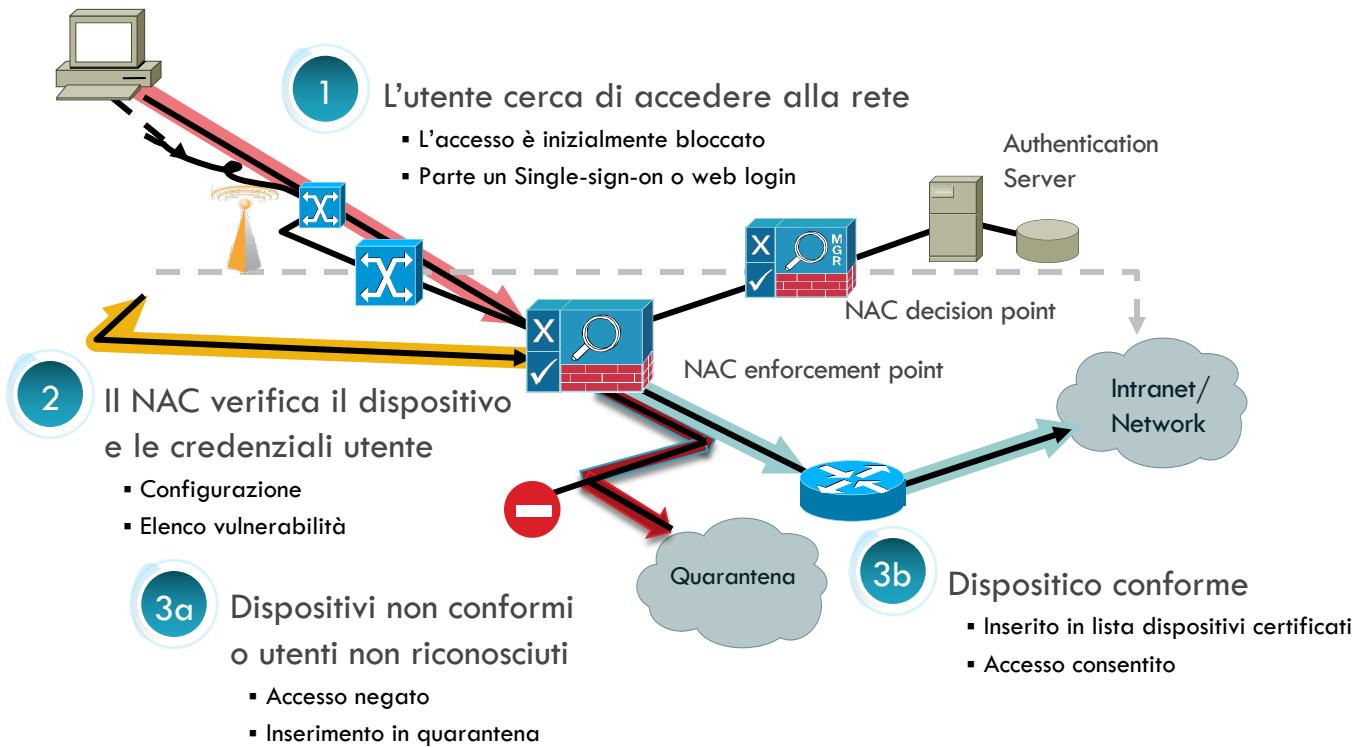
# NAC: Concetti di Base

- **Soluzioni inline o fuori banda**
  - Inline: un singolo dispositivo fa da firewall per l'enforcing di politiche di controllo accessi
  - Out-of-band: Gli agenti sui dispositivi riportano informazioni a una console centrale di management che a sua volta controlla gli switchs pilotando l'enforcement delle politiche
- In ogni caso, il NAC individua ogni device connesso alla rete, lo inserisce in una categoria e ne verifica la compatibilità con le policy di sicurezza
- verifica inoltre in maniera granulare quale tipo e livello di accesso al network è consentito al dispositivo in questione.

# NAC: Concetti di Base

- Remediation basata su **Quarantena** o **captive portals**
  - **Quarantena**: una stazione non conforme alle policy di sicurezza è autorizzata ad accedere con accesso ristretto ai soli server di patching e di aggiornamento.
  - **Captive Portal**: la tecnica del captive portal restring l'utente a una pagina Web di accesso dove lo stesso dovrà procedure all'autenticazione prima di ottenere l'accesso completo.
  - In NAC, un Captive Portal intercetta l'accesso HTTP alle pagine Web, reindirizzando gli utenti a un'applicazione Web che fornisce istruzioni e strumenti per l'aggiornamento dei loro computer.

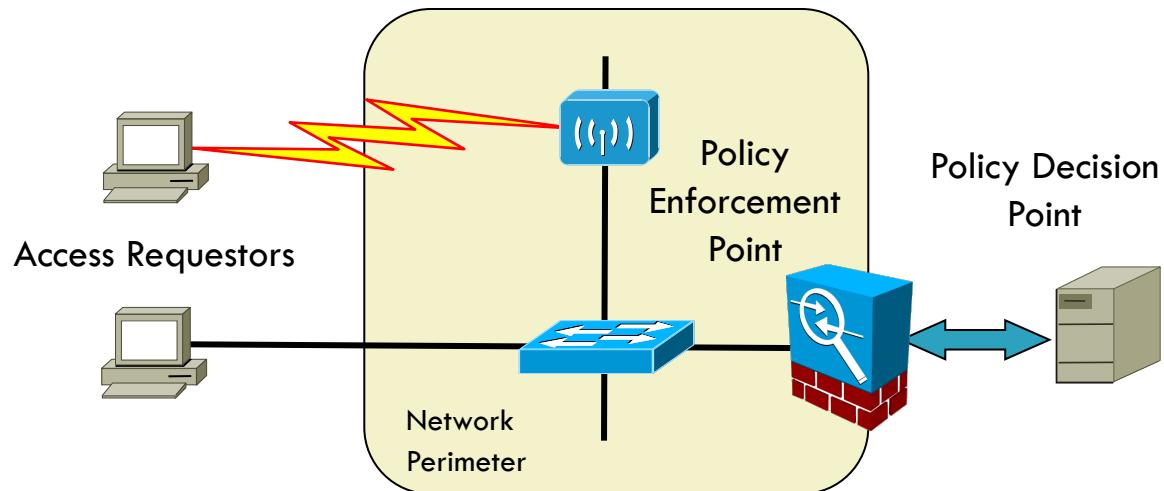
# NAC: Concetti di Base



# Componenti

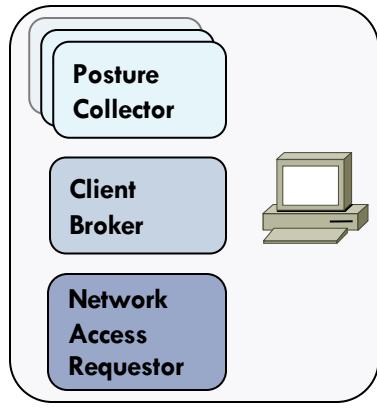
Un'architettura NAC è tipicamente caratterizzata dalle seguenti tre componenti:

- **Access Requestors**
- **Policy enforcement point**
- **Policy Decision point**



# Access Requestors

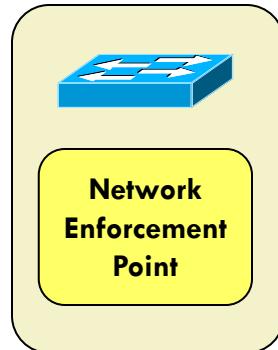
- Esempi:
  - Laptops
  - PDAs
  - VoIP phones
  - Desktops
  - Printers
- Componenti di un Access Requestor/Endpoint (AR):
  - Posture Collectors (PCS)
    - Raccoglie informazioni sullo stato della sicurezza (ad es. Software installato e aggiornato, firewall personale attivato)
  - Client Broker (CB)
    - Raccoglie le informazioni da uno o più posture collectors
    - Le consolida per passarle al Network Access Requestor
  - Network Access Requestor (NAR)
    - Connnette i clients alla rete (e.g. 802.1X)
    - Gestisce l'autenticazione a livello utente
    - Invia i dati utili ricevuti relativamente alla sicurezza delle stazioni utente (posture data) ai Posture Validators



# Posture Agent

- Un **posture agent** (PA) funge da unico punto di contatto sull'host per aggregare le credenziali da tutti i plug-in di controllo e comunicare con la rete.
- Questo modulo garantisce anche una relazione di fiducia con la rete allo scopo di scambiare credenziali e informazioni necessarie al NAC
- Funziona come un componente middleware che acquisisce le informazioni sulle politiche dell'host e le comunica in modo sicuro al server delle politiche NAC
- Interagisce direttamente con le applicazioni "abilitate per NAC" in esecuzione sull'host senza l'intervento dell'utente

# Policy Enforcement Points

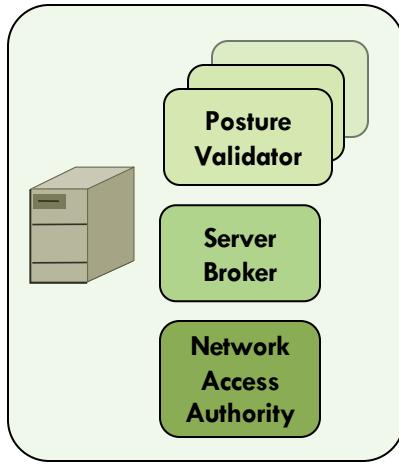


**Policy Enforcement Point**

- Componenti di un Policy Enforcement Point (PEP):
  - Network Enforcement Point
    - Garantisce e controlla l'accesso alla rete
- Esempi di Network Enforcement Points
  - Switches
  - Wireless Access Points
  - Routers
  - VPN Devices
  - Firewalls

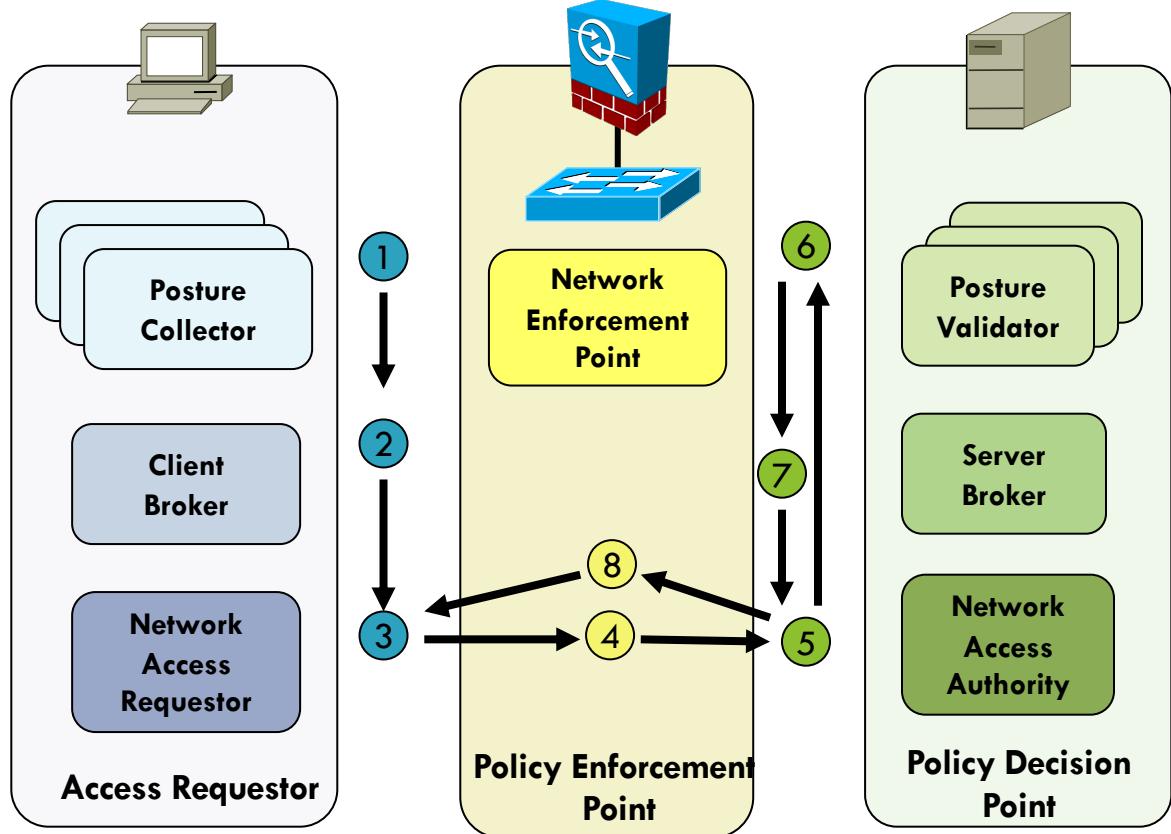
# Policy Decision Point

- Componenti di un Policy Decision Point (PDP)
  - Posture Validators (PVS)
    - Ricevono le informazioni dai corrispondenti posture collector
    - Le validano rispetto alle policy da applicare
    - Ritornano i risultati al Server Broker
  - Server Broker (SB)
    - Raccoglie/consolida le informazioni ricevute dai Posture Validator(s)
    - Determina le decisioni di accesso
    - Passa tali decisioni alla Network Access Authority
  - Network Access Authority (NAA)
    - Valida informazioni di autenticazione and posture
    - Passa il risultato di tali decisioni a un Policy Enforcement Point



# Esempio di transazione NAC

1. I PCS raccolgono informazioni sullo stato della sicurezza dell'endpoint
2. Il CB consolida i dati dell'endpoint ricevuti dai PCS e li passa al NAR
3. Il NAR ottiene tali dati di valutazione della sicurezza li invia al PEP
4. Il PEP riceve tali dati e li invia al PDP.
5. La NAA riceve i dati dal AR tramite il PEP. Se è coinvolta l'autenticazione, la NAA verifica su un DB di autenticazione se l'utente ha credenziali valide.
6. Ogni PVS controlla le informazioni dal corrispondente PCS e trasmette il verdetto al SB
7. Il SB consolida l'input del PVS in un'unica policy response
8. La NAA invia la risposta e le istruzioni corrispondenti al PEP.



# NAC: pro e contro

- Implementato correttamente, il NAC è un sistema di sicurezza che dà all'organizzazione la sensazione di avere controllo completo sulla sicurezza, anche in uno scenario in evoluzione e diversificazione
- Non è comunque una panacea. Per questo motivo il NAC dovrebbe essere utilizzato in sinergia con altri sistemi.
- Va evidenziata l'importanza di un'adeguata attività di monitoraggio in grado di rilevare se una specifica politica NAC è o meno in grado di soddisfare le esigenze di sicurezza di un'organizzazione, senza diventare eccessivamente intrusiva