Sicurezza dei sistemi e delle reti[1]

Mattia Monga

Dip. di Informatica
Università degli Studi di Milano, Italia
mattia.monga@unimi.it

a.a. 2014/15

Lezione VIII: I confini di una rete

Sicurezza perimetrale

- Poiché in Internet è una rete di reti (locali) si parla di protezione del perimetro di sottorete.
- Abbiamo già visto che l'assunzione è `locale == trusted`.
- I firewall vengono usati per definire località parzialmente diverse da quelle imposte dai mezzi trasmissivi (LAN).

Sicurezza perimetrale

Firewall

(*parete tagliafuoco*) è un dispositivo che:

- è al confine fra due reti A e B
- tutto il traffico tra A e B (e viceversa) **deve** passare attraverso di esso
- filtra il traffico secondo una precisa **politica d'accesso** (policy)

Compito dei firewall

Il compito dei firewall è stabilire quale traffico ha accesso alla rete (*policy*) e non controllare che il traffico permesso non faccia danni (*control*, intrusion detection).

Cosa sono i firewall

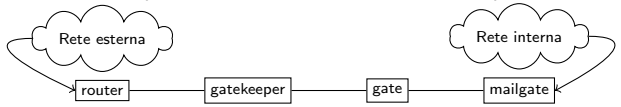
Tipicamente sono realizzati come

- Forwarding gateway
- Filtering router
- Proxy

E stabiliscono politiche (regole) ai vari livelli dello stack TCP/IP

Firewall a vari livelli

I primi firewall (Mogul, 1989 e Ranum, 1992) e



- Gatekeeper proxy applicativo: raccoglie le richieste applicative (Telnet, FTP, SMTP, ...) dall'interno e le manda verso l'esterno
- Gate filtra il traffico

Riassumendo

I firewall

- sono al confine fra due reti
- filtrano il traffico secondo una precisa **politica d'accesso** (policy)
- servono per definire zone di traffico trusted parzialmente diverse da quelle imposte dalle LAN.

Livelli firewall

In generale si possono avere firewall

- a livello applicativo (application gateway, proxy)
- a livello di trasporto (circuit gateway)
- a livello rete (packet filter)

Livelli firewall

- Esistono anche ibridi: dynamic packet filter agiscono a livello rete e trasporto (e talvolta anche applicativo).
- Possono essere realizzati via software o hardware (piú veloci, ma piú costosi e meno flessibili nelle configurazioni).

Stateless filtering

È il metodo piú semplice e piú comune

Stateless filtering

Ogni pacchetto (o comando protocollare, se a livello applicativo) è valutato in isolamento, senza tenere traccia di quelli precedenti

ACL

In pratica si tratta di avere una Access Control List (ACL) che *filtra* i pacchetti o le richieste, uno alla volta

int addr	int port	ext addr	ext port	action
*	*	a.b.c.d	*	block
192.168.2.3	110	*	110	allow

Digressione: ACL

Una ACL fissa la politica d'accesso: espressa in maniera compatta (e comprensibile). Come va interpretato *il silenzio* dell'ACL?

default deny Vietato tutto ciò che non è **esplicitamente** permesso

default permit Permesso tutto ciò che non è **esplicitamente** vietato

Default deny

Normalmente l'ACL è una serie di regole che vengono esaminate dalla prima all'ultima, quindi se l'ultima regola è equivalente a

int addr	int port	ext addr	ext port	action
*	*	*	*	block

si ha *default deny*

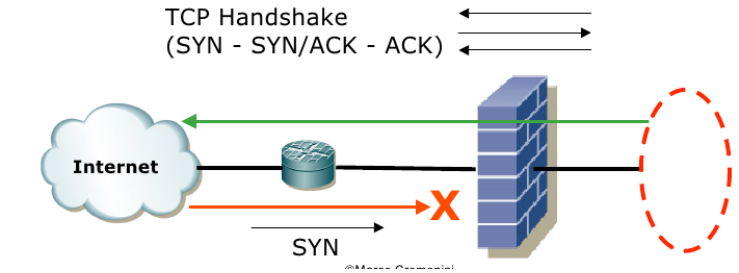
Stateful filtering

Stateful filtering

Si tiene traccia di uno *stato* del sistema e il filtraggio avviene sulla **storia** dei pacchetti o delle richieste.

Allo scopo occorre mantenere una tabella delle connessioni

Stateful filtering



client addr	client port	ext addr	ext port	state
131.175.12.1	2367	159.132.34.2	22	established

Deep packet inspection

Firewall stateful che operano filtraggio applicativo analizzando il **contenuto** dei pacchetti vengono talvolta detti deep packet filters.

- Analisi del traffico applicativo, la cui liceità va valutata caso per caso
- Generalmente basati su pattern matching di stringhe

Riassumendo

I firewall si differenziano per

- il livello a cui agiscono
- il tipo di regole di filtraggio
  - stateless
  - stateful
  - "deep packet inspection"

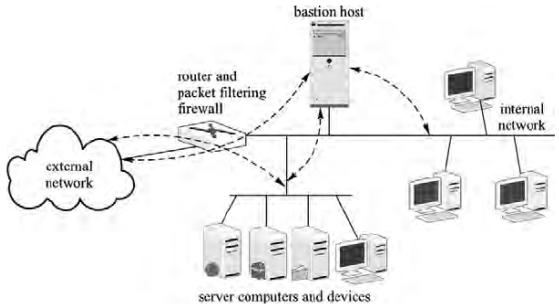
Lezione IX: Pattern ricorrenti

Configurazioni ricorrenti

SHBH *Single-homed bastion host*
DHBH *Double-homed bastion host*
DMZ *Demilitarized zone* (o *screened subnet*)

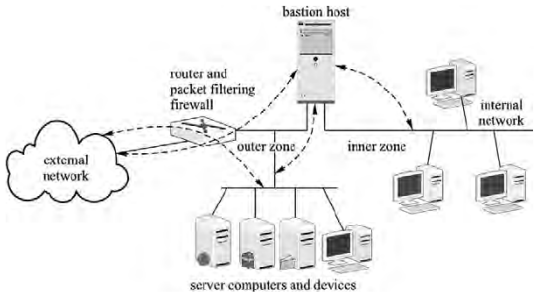
Un *bastion host* è un nodo particolarmente protetto e capace di difesa prolungata che però può essere lasciato al nemico senza danni per la rete interna.

Single-homed bastion host



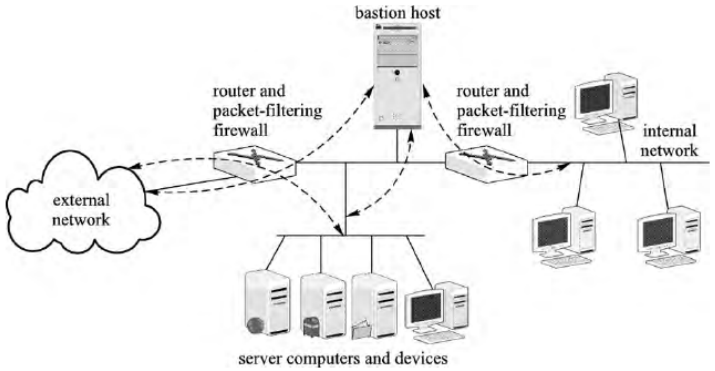
Nel caso il firewall venga compromesso, la rete interna rimane isolata (dal bastion host) dagli attacchi esterni.

Double-homed bastion host



In questo caso si hanno due sottoreti: una "intima" inaccessibile dall'esterno e una piú esterna, ma sempre difesa dal bastion host.

Screened subnet



Si usano **due** firewall per creare una zona di interdizione

Effetti di un firewall

Grazie al firewall:

- separazione in zone aventi diverso grado di sicurezza
- solo i componenti esterni al firewall sono direttamente accessibili
- è possibile regolare la "direzionalità" delle connessioni (i socket rimangono bidirezionali, naturalmente)

Riassumendo

- un firewall realizza una separazione in zone aventi diverso grado di sicurezza
- Alcune delle configurazioni piú comuni prevedono
  - bastion host
  - zone di interdizione

Stateless filtering TCP

ACL per filtraggio:

verso	IP src	IP dst	prot.	port src	port dst	flag	azione

Stateless filtering TCP

verso	IN/OUT o le zone sorgente e destinazione (es. DMZ→Internet), o delle interfacce (es. eth0→eth1)
IP sorgente/destinatario	indirizzi (es. 159.149.10.1, 159.149.10.0/24) o *variabili*
protocollo	TCP, UDP, ICMP, IP
porta sorgente/destinazione	valore o range (es. > 1023)
flag	se è attivo ACK (solo TCP)
azione	permit, deny

Uso di variabili

- scrivere politiche di tipo generale, che possono essere *istanziate* sulla specifica topologia di rete
- modificare indipendentemente politiche e topologia

Esempio

DMZ := 159.149.70.0/24
Internal := 192.168.20.0/24
Private := 10.0.0.0/8
External := not(Internal or DMZ or Private)
WebServer := 159.149.70.11 and 159.149.70.12

Protezione contro lo spoofing IP

ingress ed egress filtering.

verso	IP src	IP dst	prot.	port src	port dst	flag	azione
OUT	Internal	Any	IP	Any	Any	*	Permit
IN	External	Internal	IP	Any	Any	*	Permit
Any	Any	Any	Any	Any	Any	*	Deny

SSH con stateless filtering

La politica da implementare autorizza solo connessioni SSH dall'interno della rete aziendale verso l'esterno.
semplificazione: identifichiamo SSH con i pacchetti TCP con porta destinazione 22(si noti che talvolta si cambia la porta proprio per ragioni di sicurezza!)

verso	IP src	IP dst	prot.	port src	port dst	flag	azione
OUT	Internal	Any	TCP	> 1023	22	*	Permit
IN	Any	Internal	TCP	22	> 1023	*	Permit
Any	Any	Any	Any	Any	Any	*	Deny

SSH con stateless filtering

In realtà però possiamo notare che i pacchetti provenienti dall'esterno della rete dovrebbero essere solo risposte del server: quindi ACK deve essere settato.
Inoltre solo alcuni server ssh potrebbero essere autorizzati.

SSH

sshSrvs := 159.149.70.13 and 159.149.70.42

verso	IP src	IP dst	prot.	port src	port dst	flag	azione
OUT	Internal	sshSrvs	TCP	> 1023	22	1/0	Permit
IN	sshSrvs	Internal	TCP	22	> 1023	1	Permit
Any	Any	Any	Any	Any	Any	*	Deny

Riassumendo

- ingress ed egress filtering
- La scrittura delle regole di filtering impone di adattare la politica di sicurezza al *modello* imposto dal meccanismo di filtraggio
  - `SSH == tcp port 22`
- Occorre una conoscenza approfondita di protocolli e applicazioni

LPP

Principio del Least Privilege (LPP):

> *"ogni attore dispone del minimo dei privilegi necessari per raggiungere gli obiettivi assegnatigli dalle specifiche del sistema"*

È molto difficile da applicare: c'è una costante tensione fra flessibilità e sicurezza.

Protocolli firewall-friendly

Protocolli come Telnet, SSH, rlogin, etc. sono semplici da gestire:

- per loro natura implicano ruoli ben definiti del client e server
- il pattern di scambio di messaggi è un semplice request/reply

In generale invece esistono protocolli molto piú elaborati che richiedono politiche assai piú sofisticate per applicare il LPP.

SMTP

Politica: Nella rete aziendale un solo server SMTP è autorizzato a gestire la posta elettronica con l'esterno.

- SMTP: protocollo firewall-friendly
- Client interni alla rete non passano per il firewall

SMTP

Primo tentativo: In analogia con quanto fatto per SSH
smtpSrv := 159.149.70.23
External := not(159.149.70.0/24)

verso	IP src	IP dst	prot.	port src	port dst	flag	azione
IN	External	smtpSrv	TCP	> 1023	25	1/0	Permit
OUT	smtpSrv	External	TCP	25	> 1023	1	Permit
Any	Any	Any	Any	Any	Any	*	Deny

È corretto? No: le connessioni SYN vengono bloccate

SMTP

Le regole devono essere necessariamente piú sofisticate perché vogliamo:

- Scambiare posta: un Mail Server (MS) riceve e invia posta "da" e "verso" altri MS.
- Ricevere posta: MS si connettono al MS aziendale agendo da client.
- Inviare posta: il MS aziendale si connette ad altri MS agendo da client.

Il tipo di connessioni da gestire non è uno solo!

SMTP

Secondo tentativo:

verso	IP src	IP dst	prot.	port src	port dst	flag	azione
IN	External	smtpSrv	TCP	Any	Any	1/0	Permit
OUT	smtpSrv	External	TCP	Any	Any	1/0	Permit
Any	Any	Any	Any	Any	Any	*	Deny

SMTP

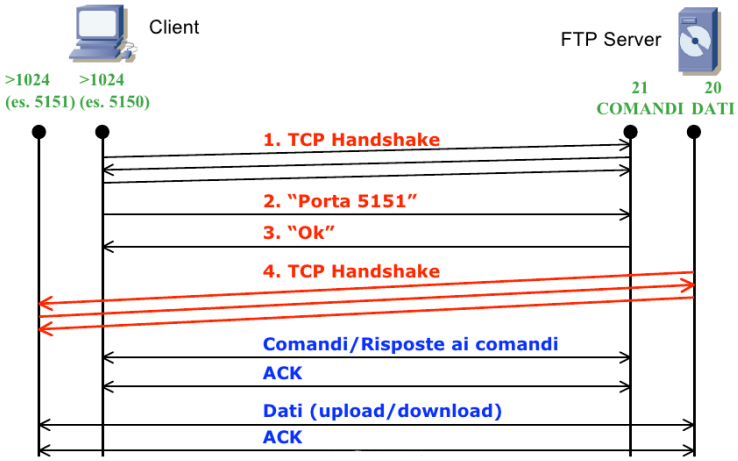
verso	IP src	IP dst	prot.	port src	port dst	flag	azione
IN	External	smtpSrv	TCP	> 1023	25	1/0	Permit
OUT	smtpSrv	External	TCP	25	> 1023	1	Permit
OUT	smtpSrv	External	TCP	> 1023	25	1/0	Permit
IN	External	smtpSrv	TCP	25	> 1023	1	Permit
Any	Any	Any	Any	Any	Any	*	Deny

Riassumendo

- Un principio che dovrebbe ispirare la scrittura delle policy è il Least Privilege
- Non è banale l'applicazione in situazioni realistiche

FTP

FTP non è un protocollo "firewall-friendly"...



FTP

verso	IP src	IP dst	prot.	port src	port dst	flag	azione
OUT	Internal	External	TCP	> 1023	21	1/0	Permit
IN	External	Internal	TCP	21	> 1023	1	Permit
IN	External	Internal	TCP	20	> 1023	**1/0**	Permit
OUT	Internal	External	TCP	> 1023	20	1	Permit
Any	Any	Any	Any	Any	Any	*	Deny

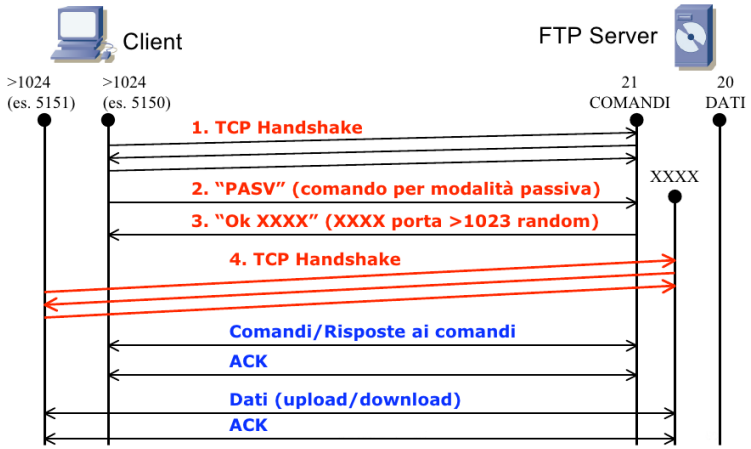
FTP

Il canale dati, dal server verso il client:
ftpserver:20 → ftpclient:XXXX
La politica di gestione "solo connessioni da interno a esterno" non è applicabile al caso in oggetto:

- connessione da esterno a interno
- porta di destinazione della connessione non determinata a priori

FTP in “passive mode”

Una nuova versione del protocollo firewall-friendly. . .



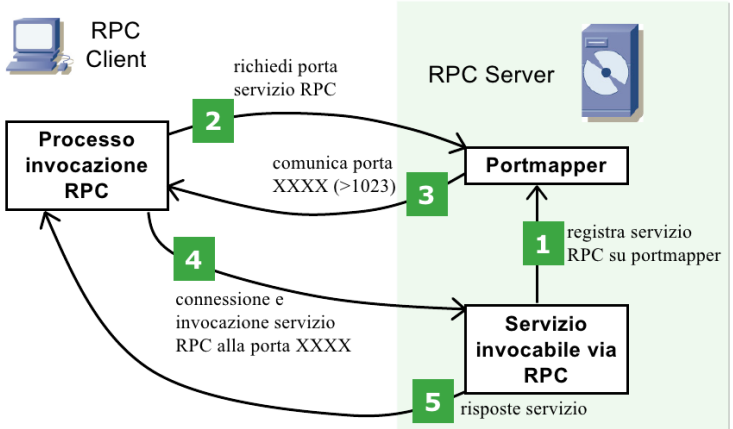
La seconda connessione, relativa al canale dati, viene aperta dal client verso il server:
ftpclient:YYYY → ftpserver:XXXX
La politica di gestione “solo connessioni solo da interno a

FTP in modo passivo

verso	IP src	IP dst	prot.	port src	port dst	flag	azione
OUT	Internal	External	TCP	> 1023	21	1/0	Permit
IN	External	Internal	TCP	21	> 1023	1	Permit
OUT	Internal	External	TCP	> 1023	> 1023	1/0	Permit
IN	External	Internal	TCP	> 1023	> 1023	1	Permit
Any	Any	Any	Any	Any	Any	*	Deny

RPC

Un protocollo complesso



RPC

Il server RPC (attraverso il servizio Portmapper, nel caso UNIX), determina dinamicamente la porta (> 1023) da assegnare al servizio RPC e quindi non si conosce a priori la porta che il server RPC assegnerà al servizio.
(Versione TCP, Unix)

verso	IP src	IP dst	prot.	port src	port dst	flag	azione
IN	External	rpcSrv	TCP	> 1023	111	1/0	Permit
OUT	rpcSrv	External	TCP	111	> 1023	1	Permit
IN	External	rpcSrv	TCP	> 1023	**Any**	**1/0**	Permit
OUT	rpcSrv	External	TCP	Any	> 1023	1	Permit
Any	Any	Any	Any	Any	Any	*	Deny

Riassumendo

Alcuni protocolli risultano piú difficili da gestire

- FTP “attivo”
- RPC