I Firewall

Metodi e strumenti per la Sicurezza informatica

Claudio Telmon
claudio@di.unipi.it

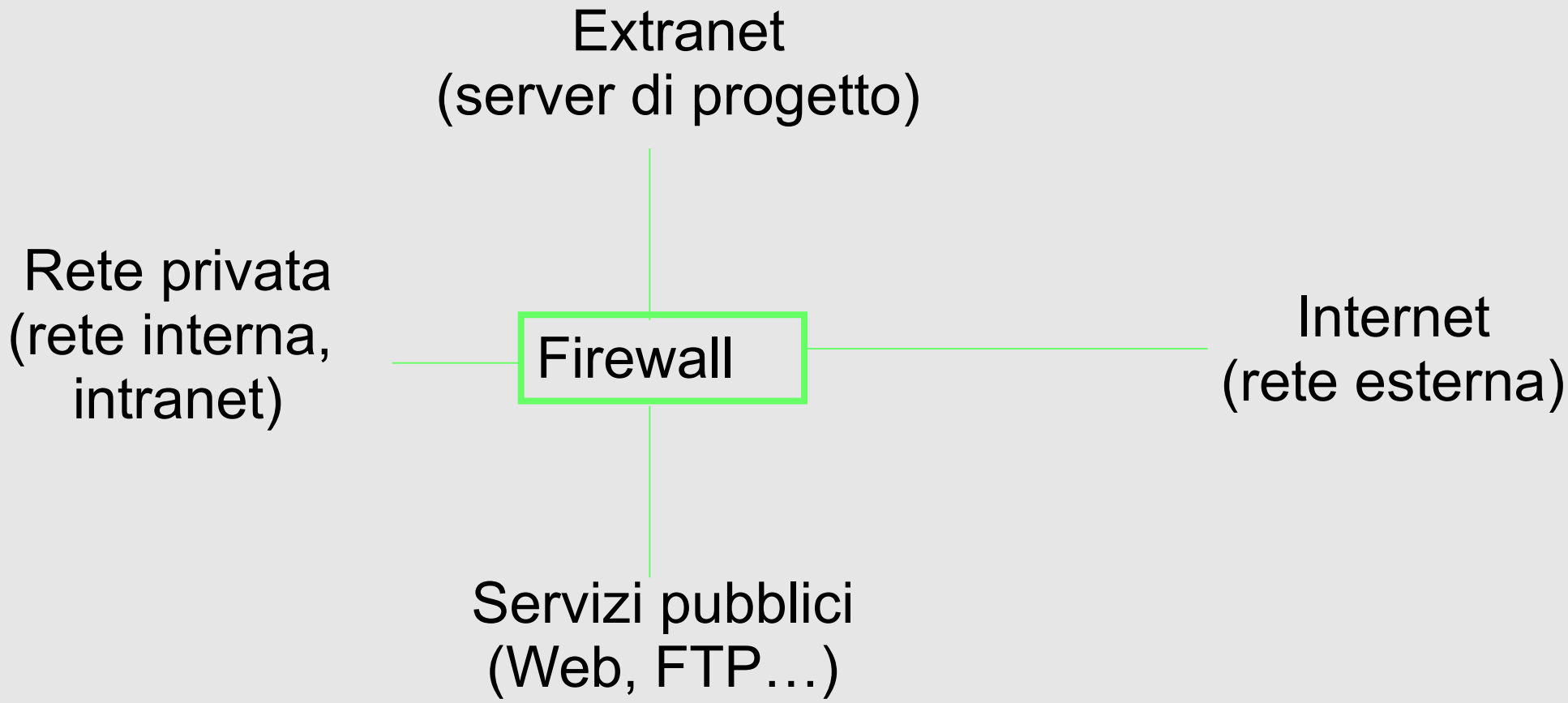
Cos'è un firewall

- Un firewall è un sistema, nel senso più ampio del termine, che ha 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

Perché installare un firewall

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

Esempio


Extranet
(server di progetto)
Rete privata
(rete interna,
intranet)
Firewall
Internet
(rete esterna)
Servizi pubblici
(Web, FTP…)

A quale livello può operare un firewall?

- Livello IP/static filtering
- Livello TCP/UDP/stateful filtering
- Protocollo applicativo
- Content inspection

Vantaggi

- Centralizzazione della politica di sicurezza
- Gestione con personale competente
- Sistema specializzato

Svantaggi

- Difficoltà con protocolli non banali
- Banda disponibile
  - la percezione dell'utente
- Single point of failure (può essere un vantaggio)
- Gestione complessa
  - configurazione
  - verifica
  - analisi dei log
- Senso di fiducia e insicurezza interna
- Costi

Parametri di filtraggio dei pacchetti

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

Azioni possibili

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

Protezione dall'IP spoofing

- Vogliamo evitare che:
  - possano essere "inseriti" pacchetti falsificati nella nostra rete
  - la nostra rete possa essere usata per inviare pacchetti falsificati
- Antispoofing: si associano reti a interfacce
  - si scartano tutti i pacchetti con indirizzo locale non
  - si scartano tutti i pacchetti provenienti dall'interfaccia interna con indirizzo non locale
  - gestire anche il loopback

Cosa un firewall non può fare

"You can't solve social problems with software"

"You're either part of the problem, or part of the solution"

- M. J. Ranum

NAT/PAT

- Non è un meccanismo di sicurezza (anche se c'è qualche vantaggio)
  - NAT statico: 1 IP interno <=> 1IP esterno
  - NAT dinamico: pool di indirizzi esterni
  - Masquerade: 1 indirizzo esterno (richiede il PAT)
- Non c'è controllo del traffico sui singoli indirizzi
- Senza specifiche ACL si accede comunque agli indirizzi interni

Un caso particolare: il masquerade

- È il NAT in cui è disponibile un unico indirizzo pubblico
- Richiede la rimappatura delle porte (PAT)
  - comporta l'utilizzo di moduli specifici per i singoli protocolli, quando possbile
- Con un opportuno filtraggio le macchine interne sono raggiungibili solo se aprono connessioni verso l'esterno
- Non è utilizzabile per i server

Packet filtering

Filtraggio statico

- Permette di bloccare indirizzi, protocolli (tcp, udp, icmp) e porte
- La granularità del controllo è limitata
- Nessuna controllo dei protocolli con gestione dinamica delle porte
- Richiede poche risorse, non mantenendo uno stato delle connessioni
  - adatto per filtraggi grossolani, come la protezione dall'IP spoofing.

Implementazione su router

- Buone prestazioni
- Trasparenza
- Basso costo (ev. nullo)
- Attualmente, soluzione da backbone (esistono strumenti più evoluti)

Bastion Host

Dove effettuare il filtraggio

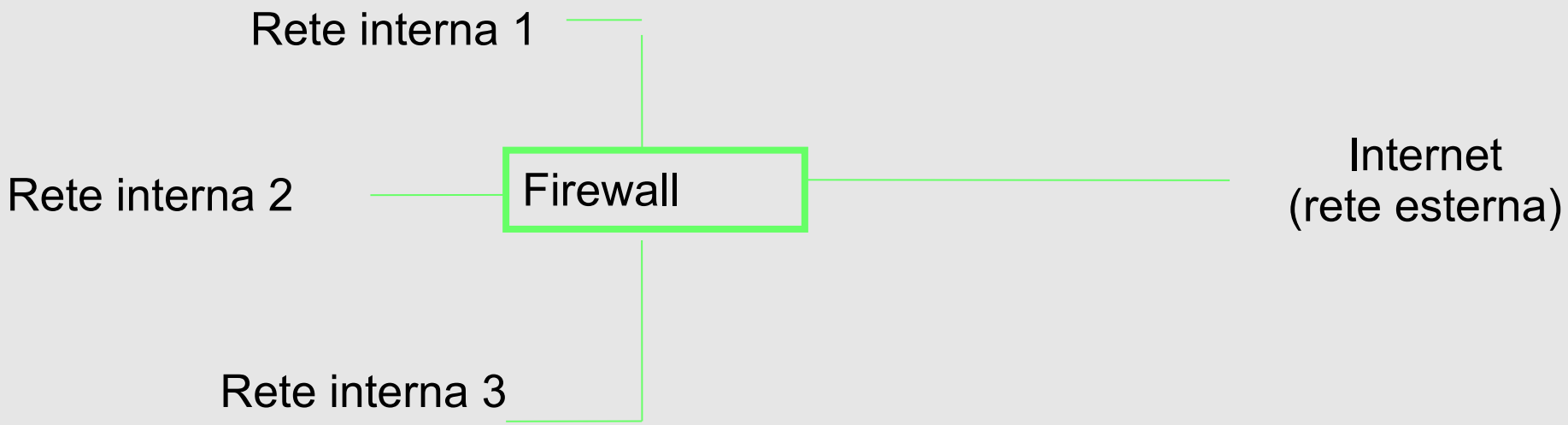
In ingresso

- so da quale interfaccia arriva il pacchetto
- proteggo il sistema locale

In uscita

- gestisco anche il traffico generato localmente

Gestione separata del traffico in transito (es. iptables)



Gestione dei pacchetti ICMP

- Il traffico ICMP non è necessario…
  - tranne per il PATH MTU discovery
  - è comunque utile per evitare rallentamenti
- È opportuno scartare tutti i pacchetti ICMP non necessari?
  - Si eliminano alcuni possibili attacchi
  - Si rallenta la scoperta dei problemi, si rallentano le attività degli utenti; forse è meglio attivare i filtri quando necessario
  - Comunque alcuni pacchetti devono passare
  - È opportuno scartare i pacchetti che forniscono informazioni

Un caso particolare: il servizio Auth/ident

- Servizio di derivazione Unix, si basa sull'affidabilità di root su tutti i sistemi
- Utilizzato dai server per "verificare" i client
- Può fornire informazioni eccessive
- È inutile ed è opportuno disabilitarlo
- Scartare i pacchetti causa rallentamenti, quindi i pacchetti vanno rifiutati
- Alcuni server non "capiscono" il pacchetto ICMP port unreachable al posto del RST

- Morale: mettersi a scartare pacchetti può causare disservizi inaspettati

Regole di filtraggio

```
pkts bytes target     prot  ifname  source          destination   ports
271 56476  ACCEPT      all  lo      127.0.0.1       127.0.0.1     n/a
0      0   DROP        all  *       127.0.0.0/8     0.0.0.0/0     n/a
0      0   DROP        all  *       0.0.0.0/0       127.0.0.0/8   n/a
0      0   ACCEPT      tcp  eth0    0.0.0.0/0       192.168.1.2   *  -> 21
0      0   ACCEPT      tcp  eth0    0.0.0.0/0       192.168.1.2   *  -> 1024:65535
0      0   ACCEPT      udp  eth1    192.168.2.10    192.168.1.2   53 -> *
0      0   DROP        tcp  *       0.0.0.0/0       0.0.0.0/0     *  -> 137:139
0      0   DROP        udp  *       0.0.0.0/0       0.0.0.0/0     *  -> 137:139
0      0   REJECT      tcp  *       0.0.0.0/0       192.168.1.2   *  -> 113
```

Direzione delle connessioni TCP

- Vogliamo permettere che sia possibile connettersi
  - da (IP A.A.A.A, porta a) a (IP B.B.B.B,porta b)
- Ma non
  - da (IP B.B.B.B,porta b) a (IP A.A.A.A, porta a)
- Problema: e i pacchetti di risposta?
- Soluzione: si scartano i pacchetti:
  - da (IP B.B.B.B,porta b) a (IP A.A.A.A, porta a) con il solo flag SYN settato (senza il flag ACK)

IP fragmentation attack (1)

- I filtri statici uno stato relativo ai frammenti/pacchetti che hanno gestito
- Un pacchetto TCP su un pacchetto IP frammentato può avere le informazioni usate per il filtraggio nel primo frammento e il resto del pacchetto nel secondo
- Il secondo frammento viene fatto passare

IP fragmentation attack(2)

- Cosa succede se i due frammenti si sovrappongono?
  - Gli standard dicono che deve essere considerato il secondo frammento
  - Questo permette di sovrascrivere parte del pacchetto, ad esempio il flag SYN
- Soluzioni:
  - mantenere uno stato
  - riassemblare i pacchetti
  - scartare i frammenti con offset piccolo

Stateful Packet Filtering

- Mantiene uno stato delle connessioni:
  - connessioni TCP aperte/semiaperte
  - pacchetti UDP usciti
  - pacchetti ICMP usciti

Stato delle connessioni

- Altri protocolli?
  - ICMP
  - traffico non TCP/UDP
- Content inspection?
  - nessun controllo (simile a proxy generici)
- Es. attacco al server DNS:
  - mi connetto a un servizio autorizzato, o invio un pacchetto non autorizzato
  - i meccanismi di log si connettono al mio server
  - invio query o genero traffico a piacere
- Gestisce FTP in modalità passive

Riconoscimento dei protocolli

- Richiede l'esame del payload
- Spesso può essere effettuato su singoli pacchetti
  - i comandi vengono inviati come singolo pacchetto…
  - … a meno di attacchi, o di casi particolari; che fare in questi casi? Assemblare più pacchetti…
- Richiede la comprensione del protocollo
- Richiede notevoli modifiche allo stack TCP/IP
- Esempi: Checkpoint Firewall-1, Cisco IOS 12.0, Cisco PIX, iptables?

Esempio (iptables)

- 0 0 ACCEPT all -- * * 0.0.0.0/0 0.0.0.0/0 ctstate RELATED,ESTABLISHED
- Permette di accettare i pacchetti facenti parte di una connessione aperta o semiaperta (ESTABILISHED) o di una connessione collegata (RELATED)
- Es. connessione dati di FTP (richiede un modulo)
- Serve un modulo analogo per il NAT

Stateful filtering: vantaggi

- Seleziona correttamente la direzione delle connessioni
- Blocca molte forme di scan
- Evita attacchi come quello di tcp fragmentation
- Permette di gestire protocolli con porte dinamiche (se c'è l'apposito modulo)
  - e se non c'è?

Stateful filtering: limiti

- Non entra nel merito del protocollo applicativo (se non per riconoscere le porte dinamiche)
- Non entra nel merito del payload
- Le regole diventano complesse in presenza di molti sistemi/servizi
- La verifica e aggiornamento delle regole non è facile

ACL su stateful packet filter

- Basate su
  - utenti
  - sistemi
  - servizi
  - orari
- Simili alle ACL dei packet filter
  - vale l'ordine delle regole
  - le azioni fondamentali sono accept, drop, reject
- Funzionalità aggiuntive
  - autenticazione utente
  - analisi del contenuto (per alcuni protocolli)

Esempi

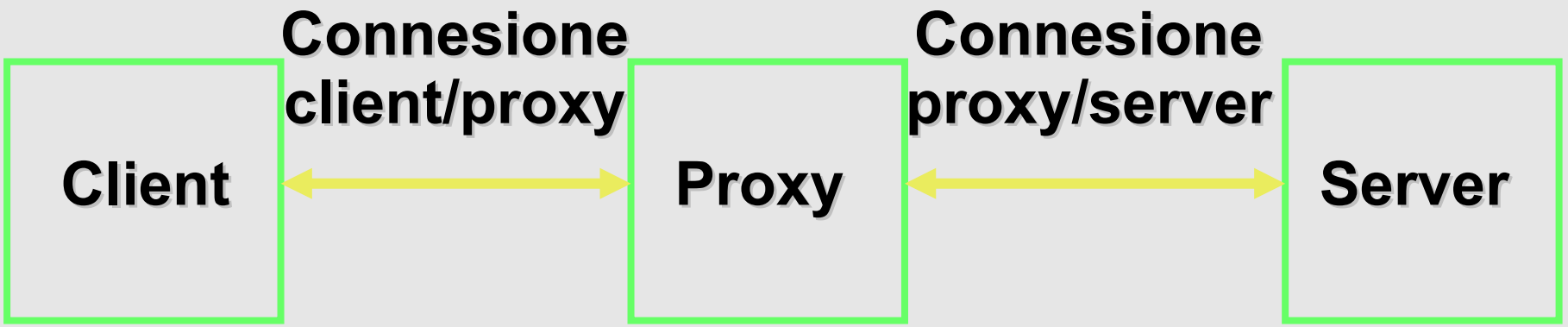
No.	Source	Destination	Service	Action	Track	Install On	Time
1	Local_Net	Any	Any	accept		FireWall	Any
2	Any	Email_Srv	smtp	accept	Long	FireWall	Any
3	Sales@Any	Email_Srv	pop-3	Session Auth	Long	FireWall	Any
4	All_Users@Any	Web_Server	http	User Auth	Long	FireWall	Any
5	Any	Any	Any	drop	Alert	FireWall	Any

Checklist

- Antispoofing
- Frammentazione
- Loopback
- Multicast
- ICMP
- Broadcast
- Servizi

Proxy/Content inspection

I proxy


Connesione
client/proxy
Connesione
proxy/server
Client
Proxy
Server

I proxy

- Ricevono la connessione dal client
- Aprono una nuova connessione verso il server
- Il traffico TCP/IP non passa direttamente da client a server

Vantaggi dei proxy

- Il traffico non passa direttamente fra client e server
  - non sono possibili attacchi basati su TCP/IP
- È immediato realizzare meccanismi per ispezionare i dati (meno banale capire cosa cercare)
- Permettono di aggiungere meccanismi di autenticazione migliori/centralizzati
  - purché compatibili con il protocollo
  - oppure preventivamente per un indirizzo IP

Svantaggi dei proxy

- Interferiscono maggiormente con il traffico
- Difficoltà nella gestione dei livelli bassi dello stack:
  - keepalive
  - ping, traceroute…
- Possono modificare l'indirizzo mittente delle connessioni
- Richiedono un proxy per ogni protocollo…
  - È uno svantaggio?
- Prestazioni

Proxy generici (Circuit Level Gateways)

- Si limitano a passare da un lato all'altro del proxy i dati TCP/UDP
- Permettono di gestire i protocolli che non fanno contrattazione dinamica di porte
- Si limitano a evitare gli attacchi dei livelli bassi dello stack

Proxy specifici (Application Level Gateways)

- Sono realizzati per la gestione di uno specifico protocollo, che quindi comprendono:
  - Possono gestire meccanismi di contrattazione di porte
  - Possono esaminare i dati trasmessi alla ricerca di attacchi o traffico non autorizzato (*content inspection*)
  - in pratica la maggior parte si limita alla correttezza del protocollo
- Permettono un'autenticazione compatibile con il protocollo

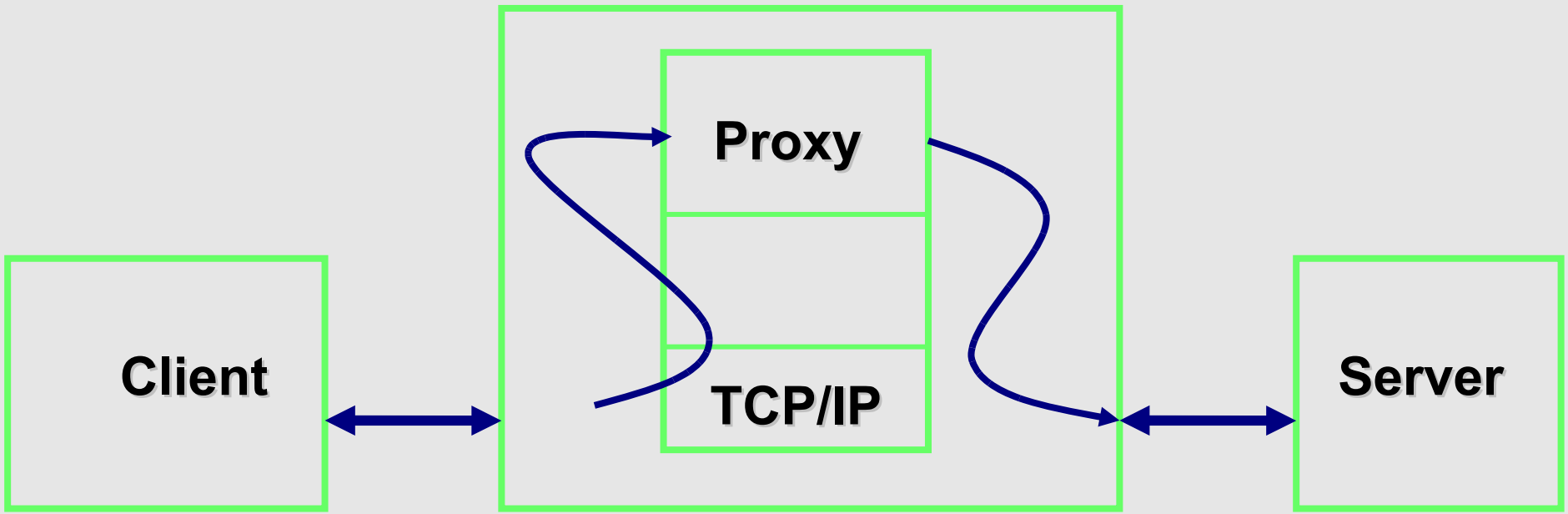
Esempi di proxy specifici

- Http:
  - filtraggio delle URL (client/server)
  - eliminazione del contenuto attivo (server/client)
  - meccanismi antivirus
- FTP:
  - riconoscimento del protocollo
  - limitazione dei comandi supportati
  - meccanismi antivirus
- Posta elettronica
  - filtraggio degli header interni
  - meccanismi antivirus

Proxy trasparenti

- L’uso di proxy richiede che la connessione avvenga tra client e proxy
- Non tutti i protocolli supportano l’uso di proxy
- Come comunicare l’informazione su indirizzo e porta del server?
- Soluzione: il client crede di comunicare con il server, e la comunicazione viene intercettata dal proxy

Proxy trasparenti



Proxy trasparenti

- Richiedono la modifica nello stack TCP/IP
- Le informazioni sul server sono prese direttamente dai pacchetti IP
- Sono invisibili al client
  - possono essere usati con client generici
  - il client non può comunicare informazioni al proxy (porte su cui ascolta)
- Hanno prestazioni analoghe a quelle dei proxy
- Richiedono la risoluzione DNS da parte dei client
- Quale protocollo applicativo selezionare?

Socks

- È un protocollo che permette la comunicazione fra client e proxy di informazioni sulla connessione:
  - indirizzo e porta del server
  - autenticazione
  - porte su cui il client si mette in ascolto?
- È un meccanismo molto versatile…
- … purché il client supporti il protocollo
- Non entra nel merito dei dati trasmessi
- Si fida del client?

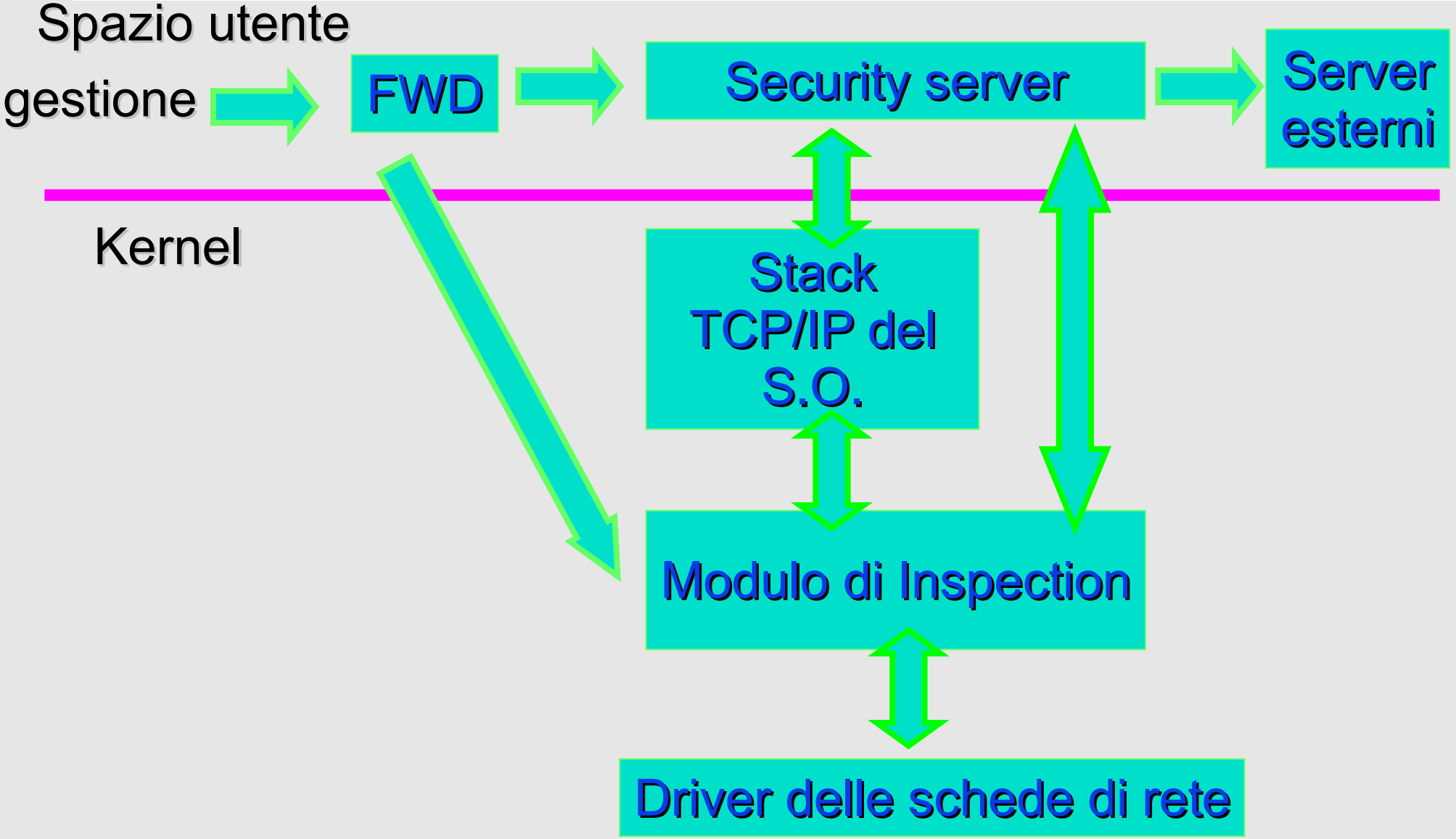
Content inspection

- Ad oggi, lo stateful filtering è “scontato” per qualsiasi prodotto
- La maggior parte dei nuovi attacchi è a livello applicativo
- La content inspection può essere effettuata in modo “semplice” sui proxy
- È più complessa con il packet filtering

SPF e content inspection

- L’esame dei dati di una sessione richiede l’esame di dati contenuti in più pacchetti
- L’esame è complesso
- Che fare dei pacchetti nel frattempo? (Se l’ACK non arriva, dopo un po’la trasmissione di nuovi pacchetti si interrompe)
- In pratica, per esami complessi i dati sono passati ad applicazioni

Esempio: Checkpoint Firewall-1



Esempio: Checkpoint Firewall-1

- Per la content inspection di fatto i dati sono passati a un'applicazione:
  - i vantaggi in termini di prestazioni scompaiono, anche di fronte al costo dell'ispezione; rimane un'ottimizzazione nella gestione dei pacchetti
- Lo stack TCP/IP modificato è ottimizzato ma meno testato
- Server esterni:
  - autenticazione
  - server CVP

Firewall-1 e il content filtering

- L’esame della sessione e i protocolli con contrattazione di porte richiedono di ricostruire la sessione tramite una *virtual machine*
- I dati vengono passati ai *security server*, applicazioni per il content filtering e l’autenticazione
  - auhtentication server
  - CVP server (interfaccia per antivirus)
  - server per traffico http, smtp, ftp

Altre funzionalità

- Protezione dai SYN flood
  - es. invio di un RESET dopo un timeout
- protezione dai fragmentation attack/scan (più corretta dei router)
- antispoofing
- Network Address Translation (NAT)
- Gestione centralizzata di più firewall/politiche

Content Vectoring Protocol

- Problema: passare i dati da ispezionare a server esterni (tipicamente antivirus)
- Soluzione: fornire un'interfaccia standard fra firewall e server esterno
- Quanto è efficace un antivirus nell'esaminare il traffico in rete?
  - Architetture diverse
  - Diversi meccanismi di compressione
  - Prestazioni di un controllo centralizzato vs. distribuito
  - Serve comunque proteggersi dai floppy…

Content Inspection: esiste davvero?

- Generalmente è realizzata solo per i protocolli più comuni, se non in termini di corrispondenza al protocollo
- Sempre più applicazioni usano http come protocollo, “tanto passa dai firewall”
- Che possibilità c’è di capire cosa sta passando?
- Cosa capire di un applet o uno script?

Content inspection: nuove tendenze

- Alcuni produttori stanno finalmente inserendo nei loro prodotti della vera content inspection
- È bene comunque verificare cosa viene realmente fornito
- Funzionalità:
  - verifica della conformità al protocollo
  - rilevazione di attacchi noti
  - input validation su regole e politiche definite

Applicazioni personalizzate

- È difficile che un firewall commerciale protegga le nostre applicazioni personalizzate
  - dovrebbe capire l'input che abbiamo definito noi
  - può comunque riconoscere tipologie di attacco note (es. directory path traversal)
  - se siamo in grado di riconoscere il problema sul firewall, dovremmo riconoscerlo sull'applicativo
  - fornisce comunque ridondanza e centralizzazione
- Avere un firewall davanti a un servizio pubblico non è una garanzia per quel servizio

Complessità del proxy: un rischio

- Più il proxy (o il modulo per SPF) capisce il protocollo, più assomiglia a un client e a un server
- Oltre un certo limite, la complessità lo rende vulnerabile quanto un client o un server
- La conoscenza dell'ambiente è comunque limitata (es. interazione con l'ambiente di Microsoft IIS)

Gestione del DNS

- Gli host interni devono poter vedere gli host esterni
- Si può decidere di nascondere la rete interna
- In tal caso il DNS richiede una configurazione particolare
- Sarà necessario riscrivere, ad esempio, gli header della posta, e istruire gli utenti

Semplicità vs. Corettezza

- Posizione 1: il firewall deve essere essenzialmente semplice: si elimina il più possibile e si irrobustisce il sistema op.
- Posizione 2: un sistema operativo debole, per di più modificato, è soggetto agli errori suoi e delle applicazioni. Serve un sistema operativo intrinsecamente più sicuro

Personal firewall

- Nati soprattutto per linee xDSL
- Controllano:
  - quali applicazioni possono accedere alla rete
  - pacchetti in arrivo dalla rete
- Generano molti falsi positivi
  - ogni pacchetto non previsto è segnalato: su linee dial-up segnalano anche traffico normale
  - l'utenza non sa interpretare i messaggi
  - Es. alcuni provider considerano "a bassa priorità" le segnalazioni dovute a questi strumenti

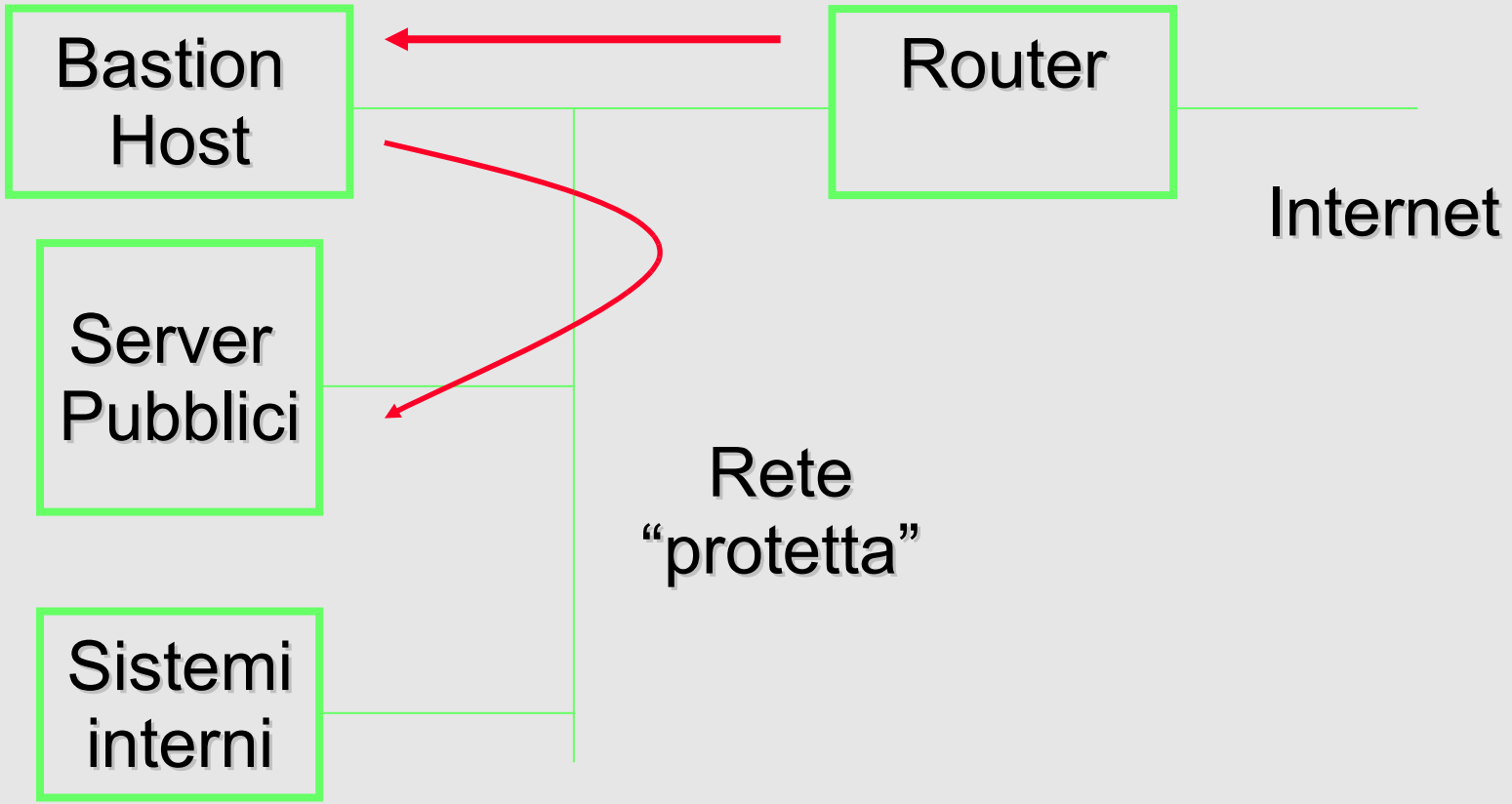
Architetture

Principî guida

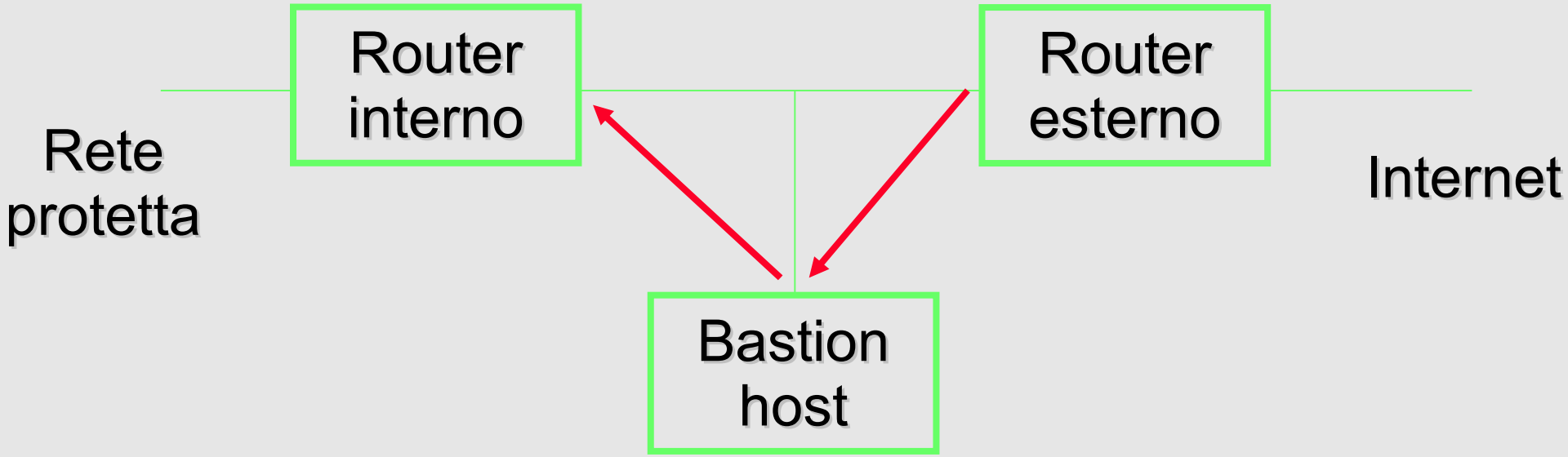
- Ridondanza delle protezioni
  - nessun meccanismo è prefetto
- Separazione fra servizi, sistemi e reti con diversi requisiti di sicurezza
- Flessibilità
- Semplicità
- Uso della nostra rete per attacchi su Internet? (es. proxy “aperti”)
- Verifica: se un meccanismo cede, cosa succede?

Singolo “dual homed bastion host”

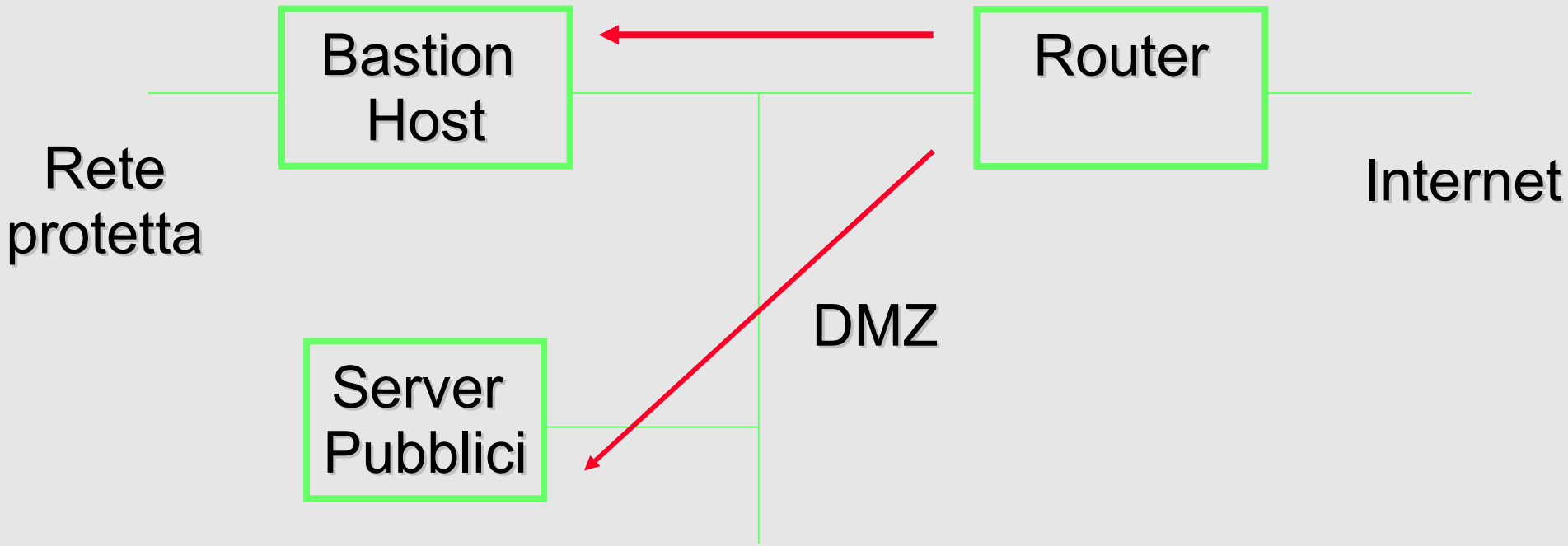
Screening router: rete protetta?



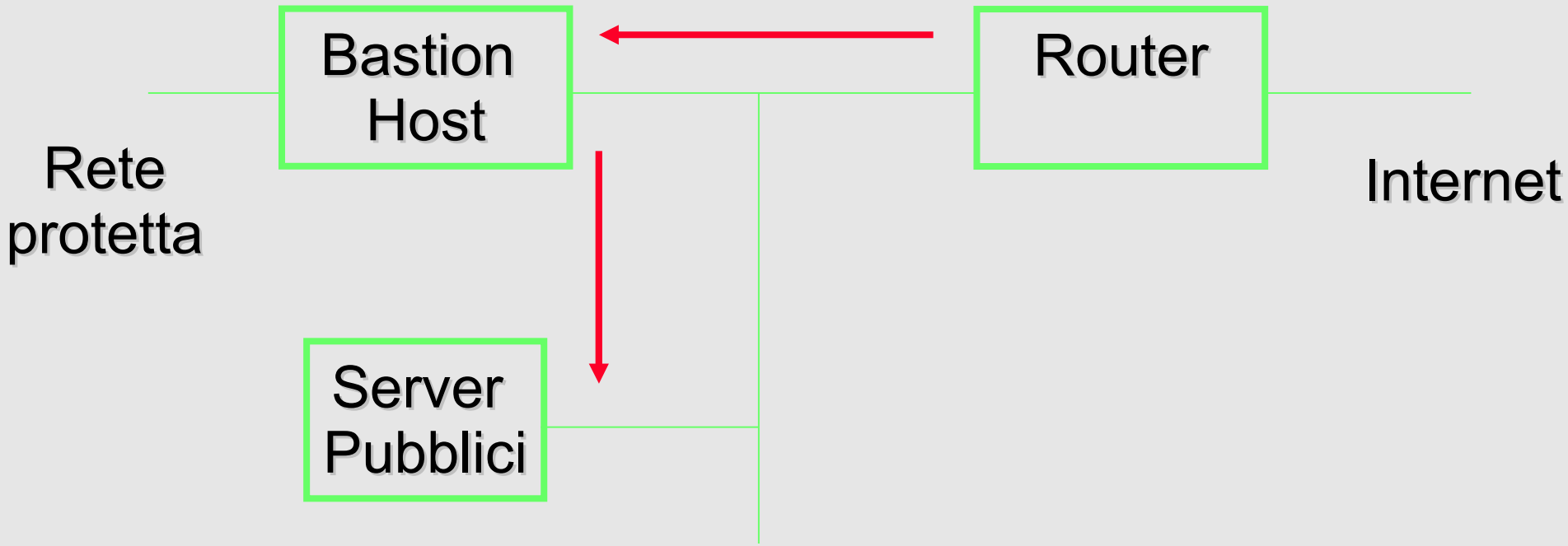
Screened subnet: protezioni ridondanti?



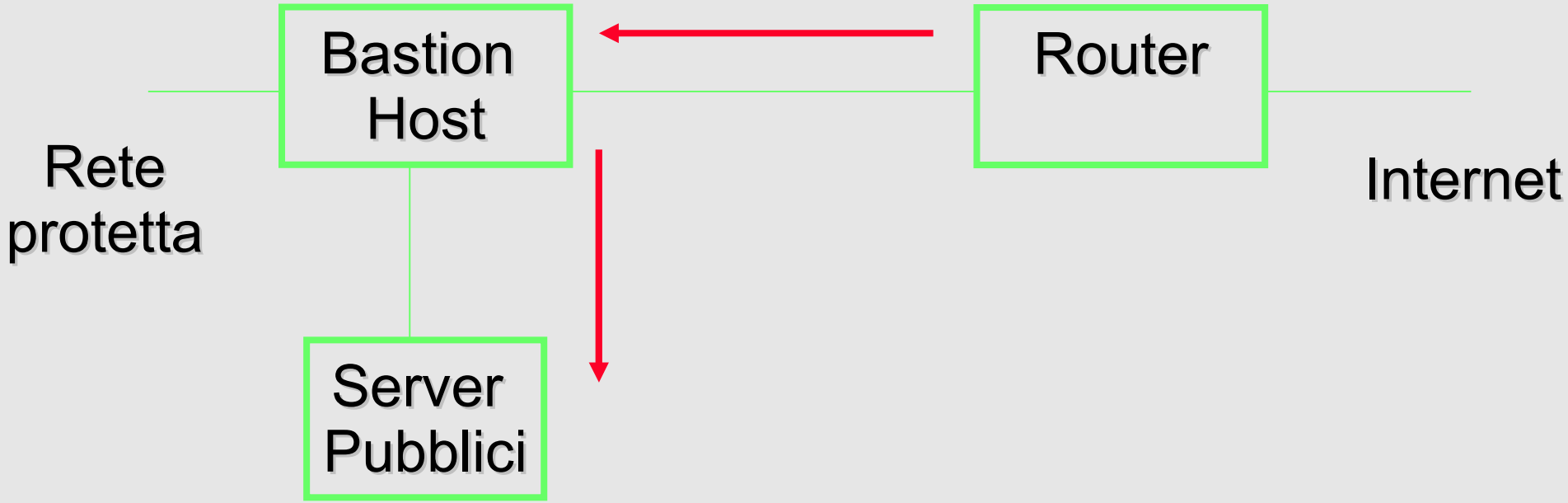
Configurazione base con DMZ



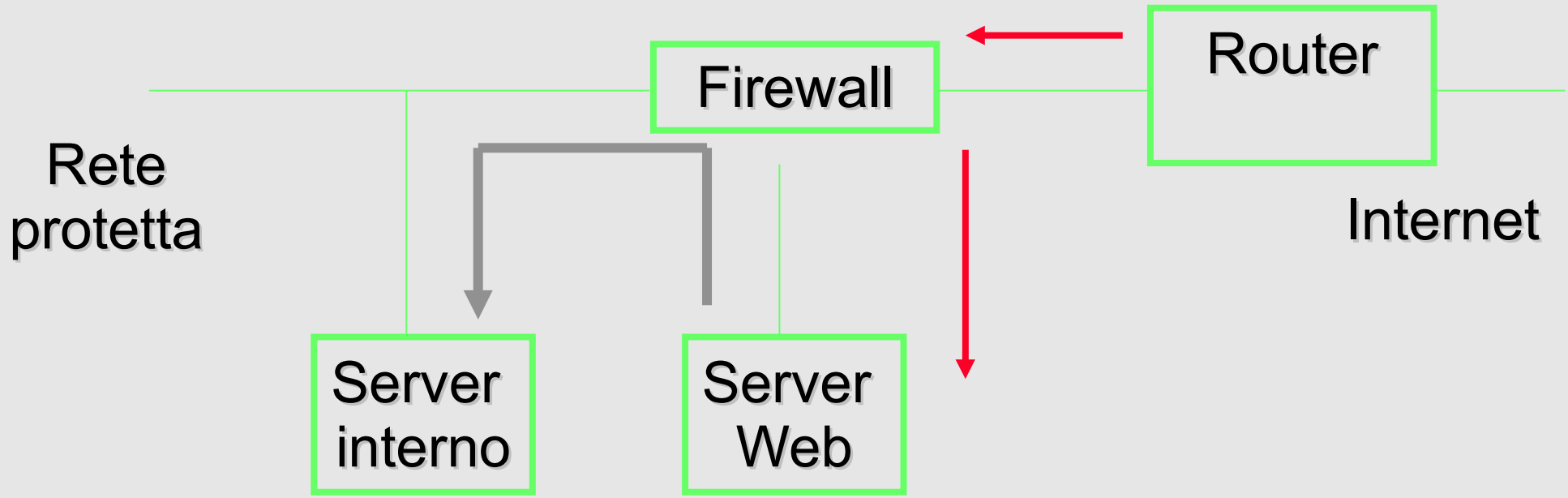
Controllo del traffico da parte del bastion host



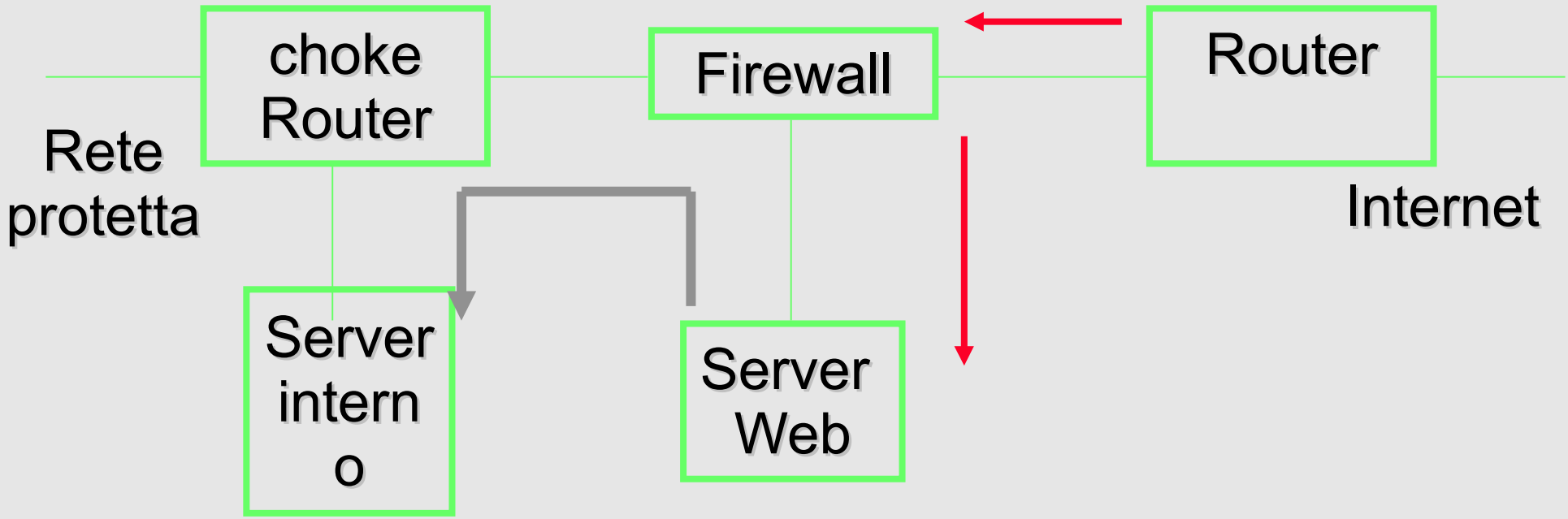
DMZ Protetta



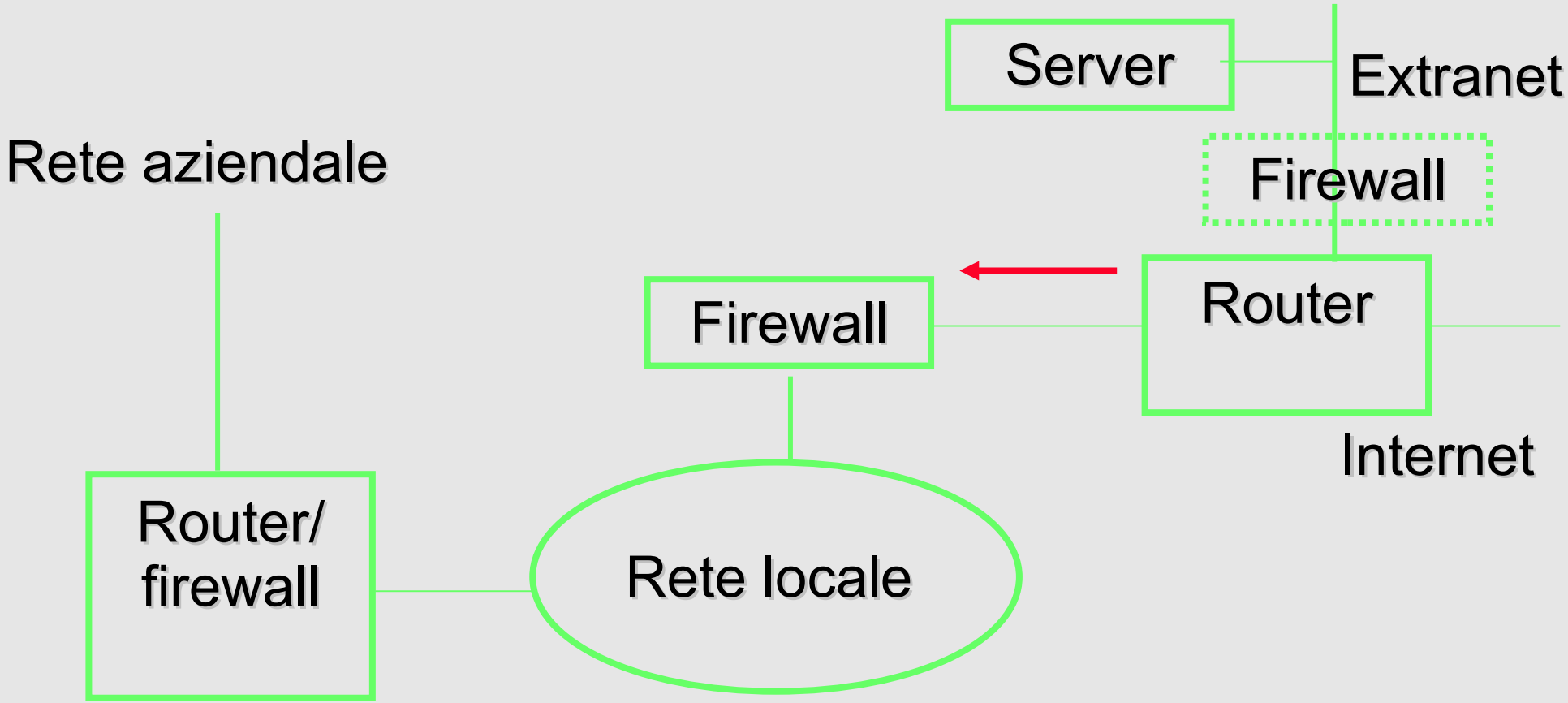
Accesso a sistemi interni



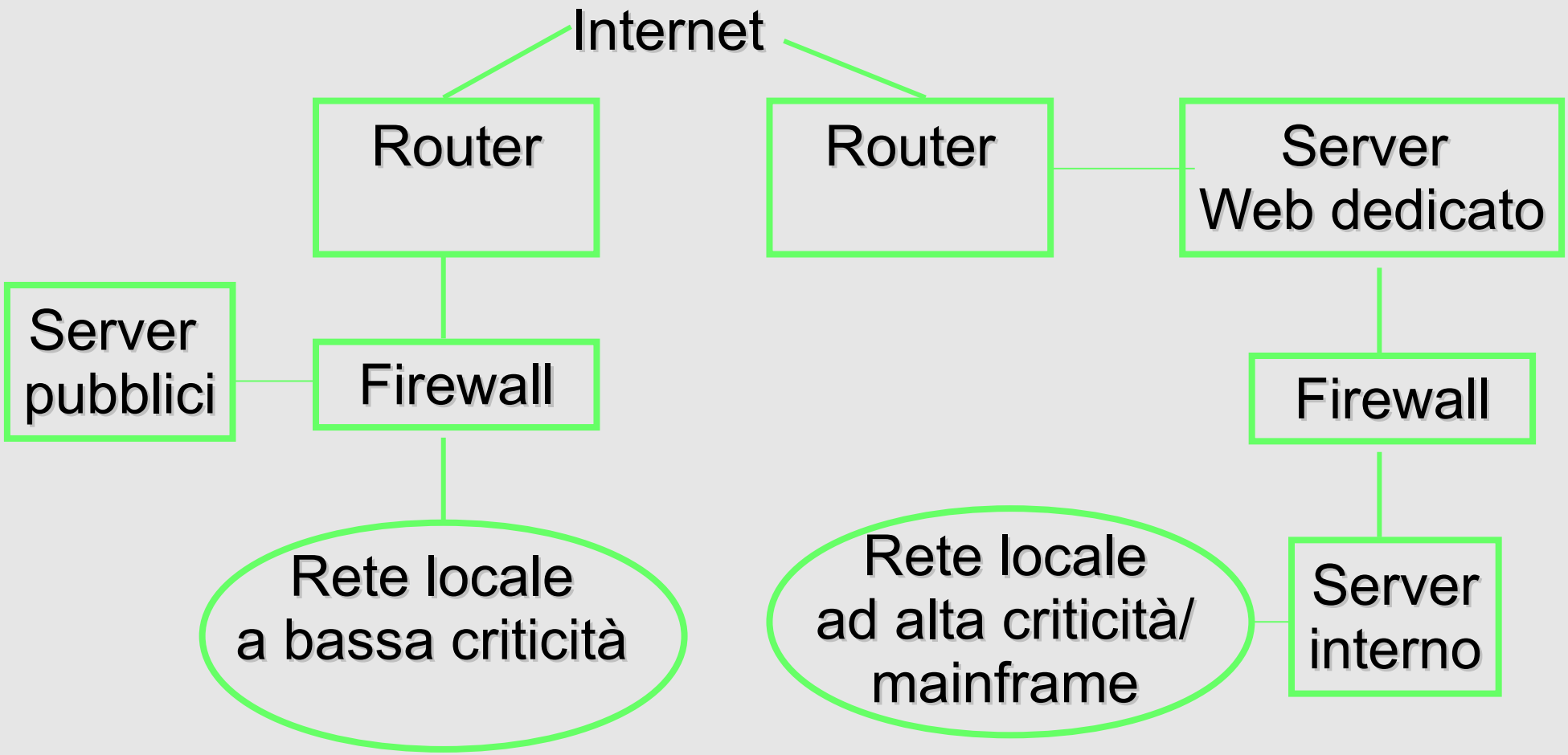
Accesso a sistemi interni: “belt and suspenders”



Sistemi più complessi



Separazione del traffico



Funzionalità aggiuntive

- VPN
  - si integra bene con il firewall
  - evita numerosi problemi (mancato filtraggio, collocazione dell'end-point...)
- Servizi
  - in generale è bene evitare, si aggiunge compessità, punti di attacco e si aumentano i compromessi
- Connettività
  - es. firewall/router; sono funzionalità che si integrano bene, i problemi possono venire dalla gestione

Load balancing

- Viene spesso trattato in modo analogo a quello dei router, ma:
  - c'è uno stato rilevante da tenere
  - la perdita dello stato interrompe (solo) le connessioni esistenti
- Esistono prodotti specifici per allineare lo stato
  - a volte si può trascurare qualche dettaglio, es. accettare gli ack

Bibliografia

"Building Internet Firewalls"
E.D. Zwicky, S. Cooper & D.B. Chapman
ed. O'Reilly, 2000
ISBN 1-56592-871-7