La Sicurezza in rete: nuovi trends, nuove minacce, nuove tecniche di difesa

Francesco Palmieri

Università Federico II di Napoli

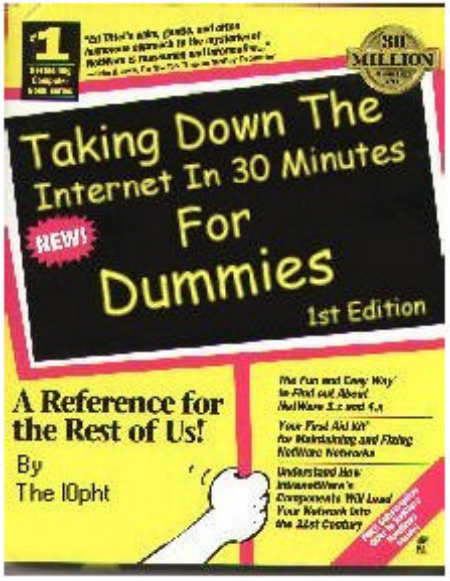
fpalmier@unina.it

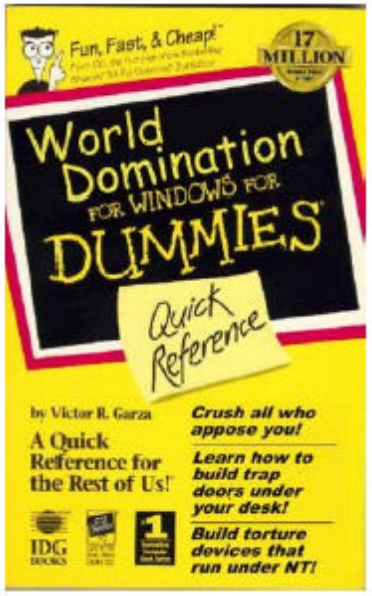
Le premesse

- Internet è un infrastruttura critica ma il suo modello di sicurezza di base è cambiato molto poco a partire dalle origini (anni 70).
- Negli ultimi anni la rete è diventata elemento strategico per il business e strumento essenziale in tutte le attività lavorative e sociali
- Si parla ormai estesamente di sfruttamento commerciale della rete
- In poche parole, la connettività in rete ha acquisito un valore *irrinunciabile* per tutte le istituzioni pubbliche o private

Le nuove minacce

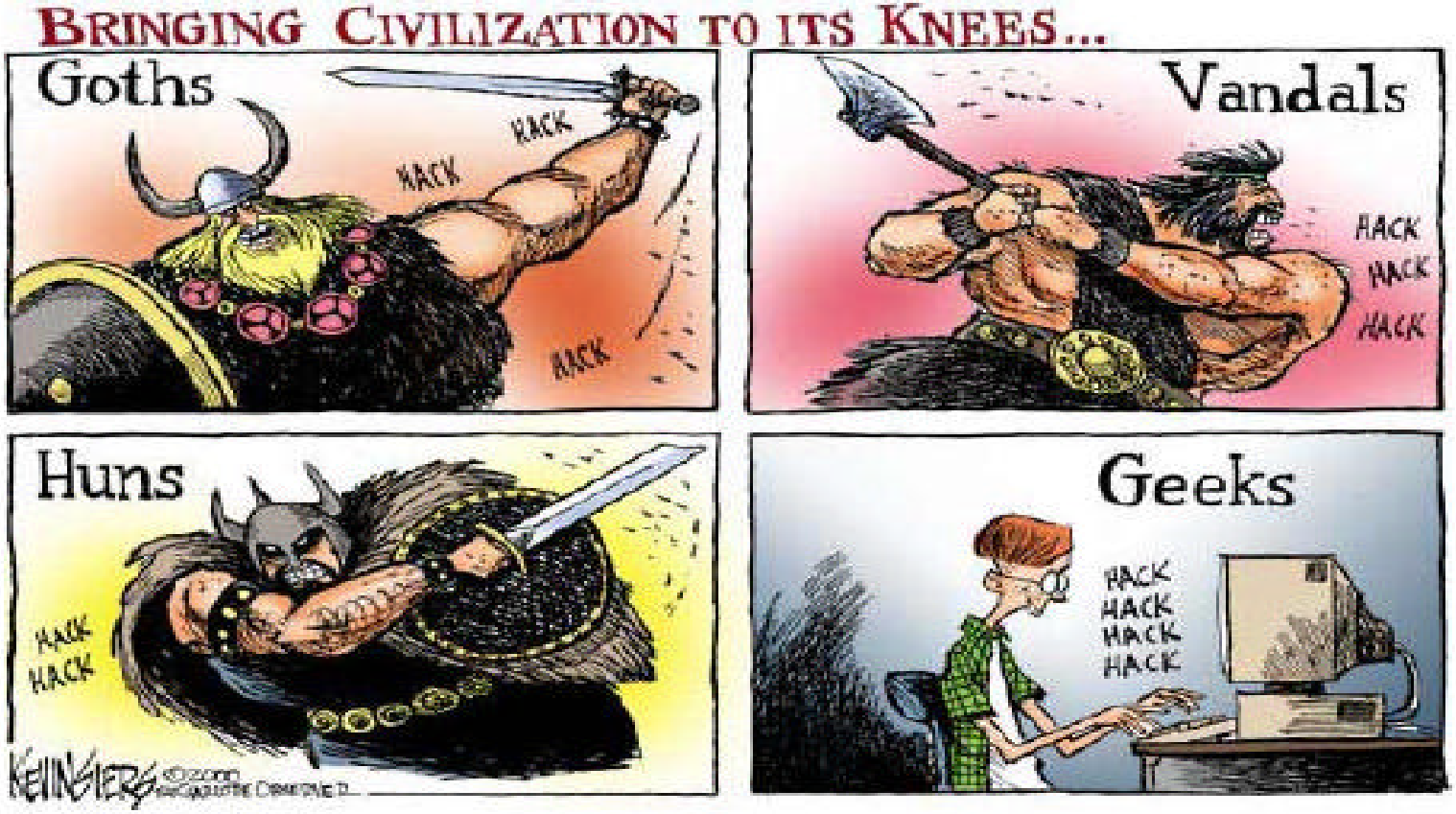
- Ingegnerizzazione e diffusione di massa, attraverso la stessa rete, delle tecniche di danneggiamento e sfruttamento illecito di risorse altrui
- Lo sviluppo di nuove tecniche e metodi di attacco e offesa procede più velocemente della ricerca e diffusione di nuovi paradigmi di sicurezza





Le nuove minacce

- Si moltiplicano gli episodi di *vandalismo informatico,* a breve si parlerà di *cyberterrorismo*
- Il danneggiamento o la *negazione (DoS) della connettività* è ora il principale problema di sicurezza



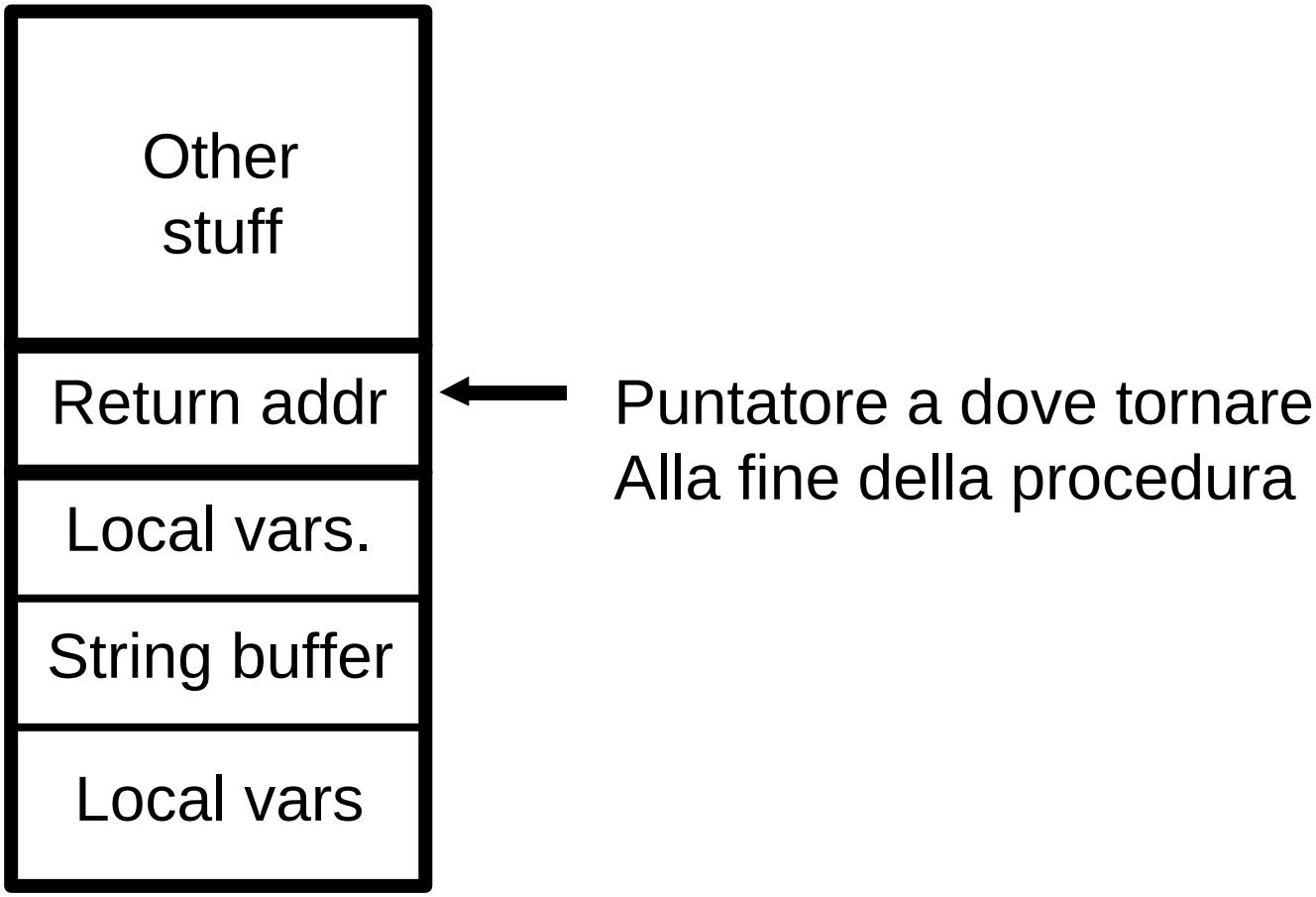
Gli obiettivi “sensibili”

- **I meccanismi di base dell’infrastruttura Internet**
  - DNS (Domain Name System)
  - BGPv4 (Border Gateway Protocol)
  - MPLS e servizi MPLS-based (VPN, FRR etc.)
- **La stabilità generica della rete: DoS, Worms e Virus**
- **I grandi portali servizi WWW (governativi, e-business, e-banking, informativi etc.)**

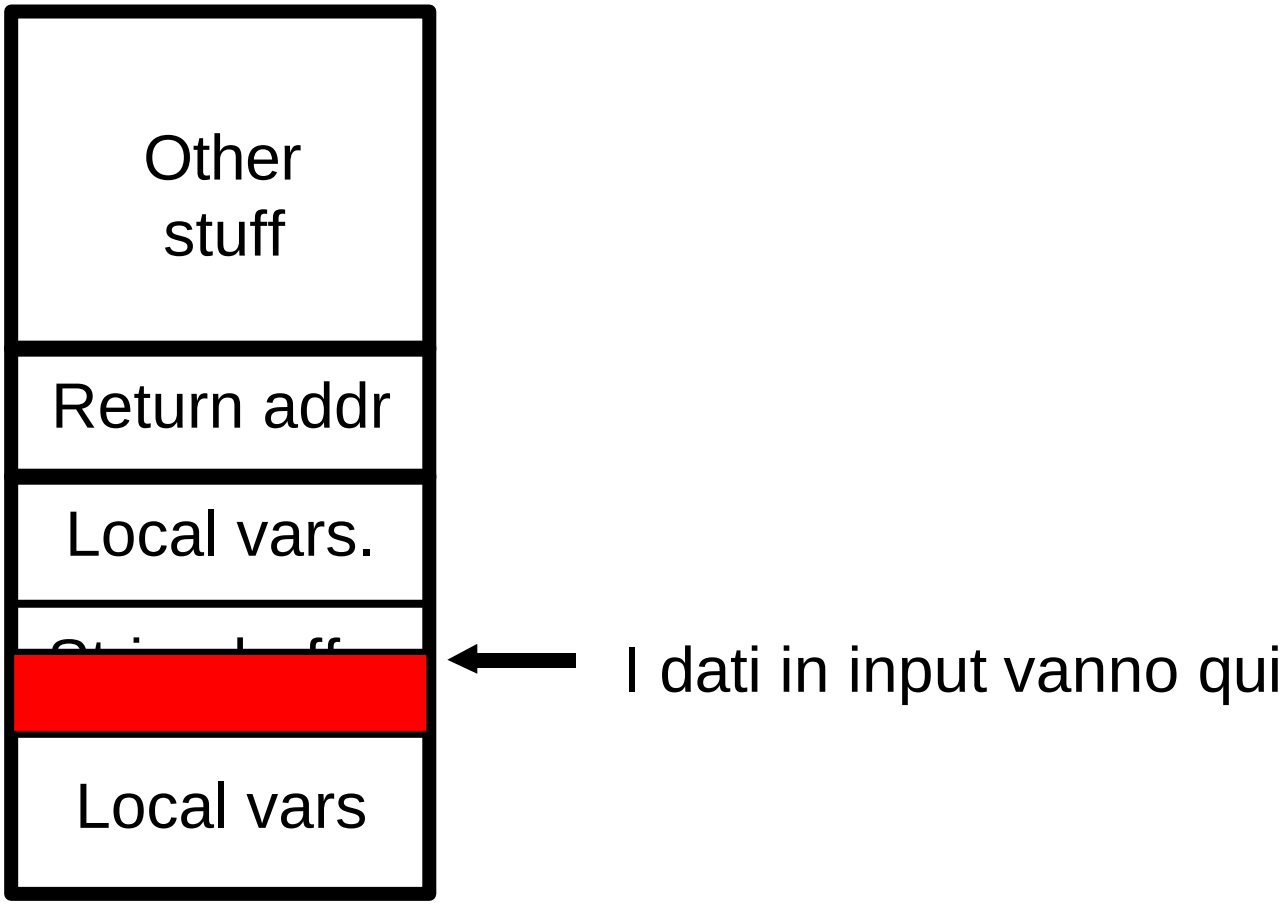
DNS/BIND: I rischi maggiori

- Il servizio DNS/Bind gira come root su hosts vitali per la funzionalità della rete (in particolare i root servers, ma non solo)
- Il software in molte implementazioni è voluminoso e ancora poco conosciuto
- E’ stato spesso vulnerabile ad attacchi di stack-smashing/buffer overflow
- E’ in taluni casi vulnerabile ad attacchi di Answer Spoofing e Cache poisoning
- Un DNS server è il miglior candidato per un DoS

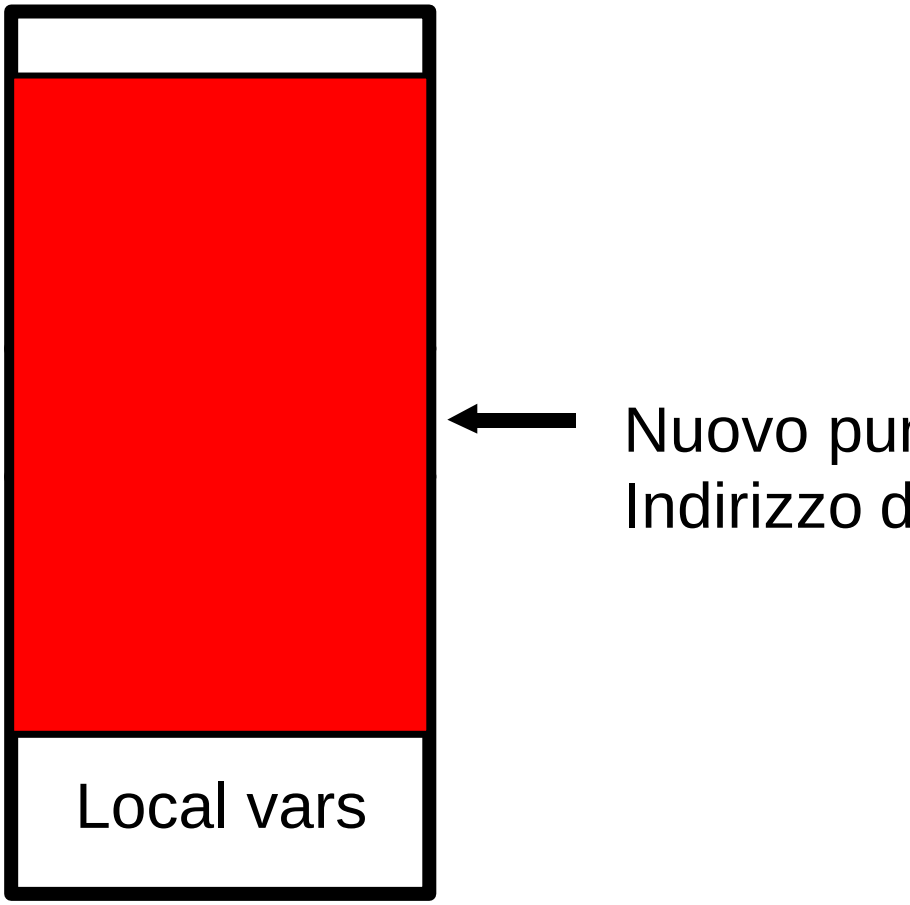
Stack Smashing: la dinamica



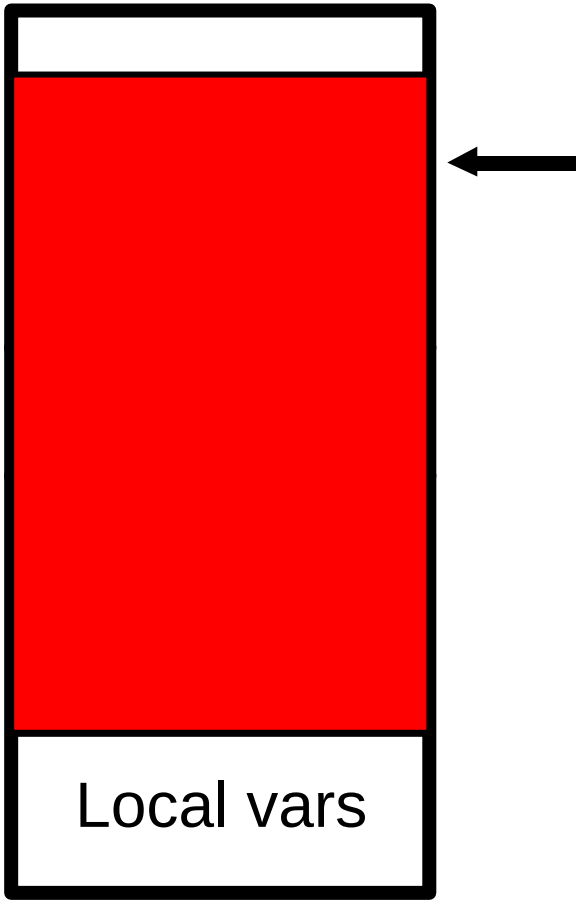
Stack Smashing: la dinamica



Stack Smashing: la dinamica



Stack Smashing: la dinamica

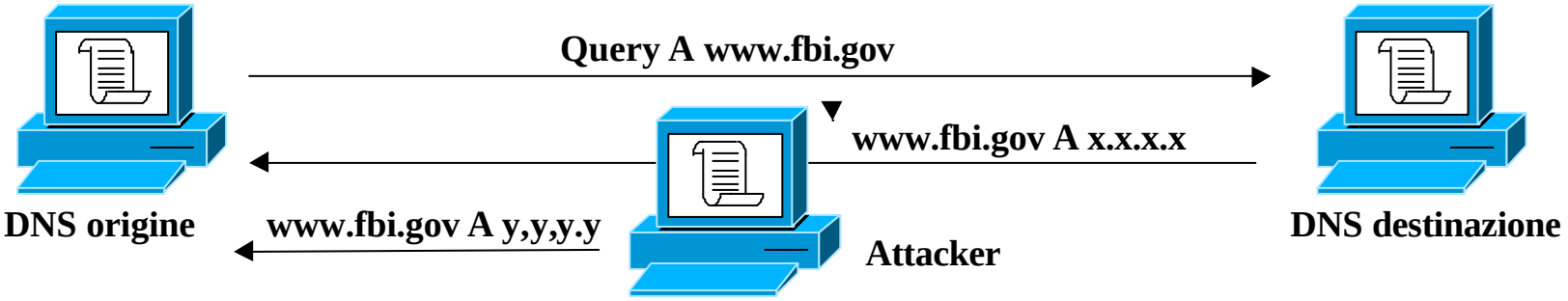


… nuovo codice caricato
Sullo stack

Il codice inserito sullo stack gira con i privilegi del programma ospite, spesso root

BIND: cache poisoning e spoofing

- L’attaccante fa in modo che venga fornita dolosamente una risposta scorretta a una query da parte di un DNS server referenziando un target diverso in una entry in cache che:
  - Permette lo spoof di una login con cattura di passwords e codici di carte di credito
  - Permette lo spoof di una pagina web, fornendo diversi risultati rispetto alla pagina originale
  - Predispone l’ambiente per attacchi man-in-the-middle, facendo relay delle informazioni verso il vero server
- Il TTL viene passato in maniera tale che l’entry fraudolenta resti in cache il più possibile
- E’ possibile inviare risposte DNS senza sollecitazioni da parte di una query e fare spoofing delle risposte



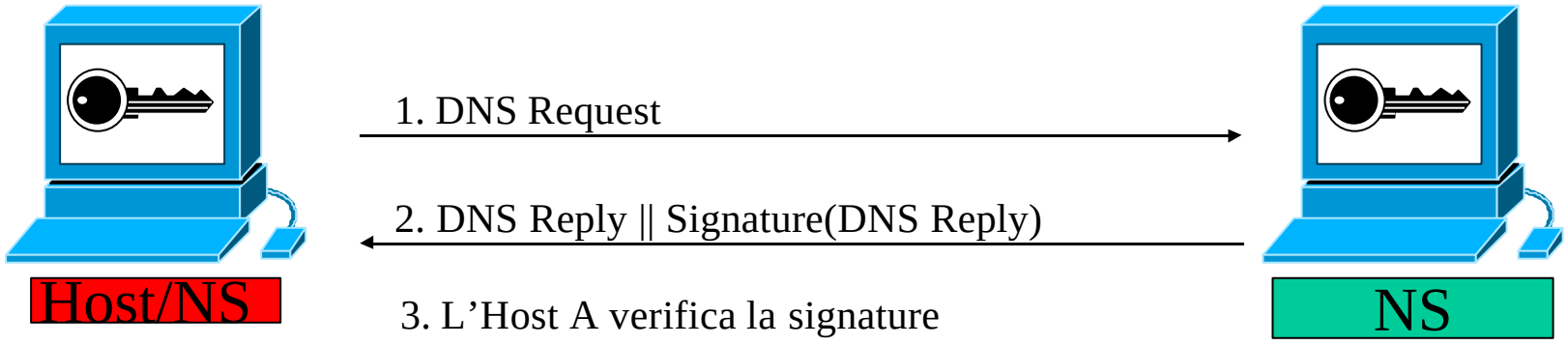
BIND: possibili contromisure

- Mantenere bind costantemente aggiornato all'ultima versione stabile

  http://www.isc.org/products/BIND/
- Evitare che BIND giri come root confinandolo (*chroot*) in un ambiente ristretto (*bind-in-a-jail*)
- Autenticazione di aggiornamenti e zone transfers tramite transaction signatures (TSIG – RFC 2845)
- Uso di DNS SECURE (DNSSEC - RFC 2535) che supporta l'autenticazione dell'origine e il controllo di integrità per ciascuna query attraverso meccanismi di firma digitale

BIND: DNSSEC in breve

- Ogni insieme di RR (RRset) inviato in risposta a una query sarà accompagnato da una digital signature generata attraverso la chiave privata dell'origine
- Il DNS server destinatario può verificare attraverso la digital signature l'autenticità e l'integrità del messaggio



- DNSSEC specifica tre nuovi RR:
  - KEY, che rappresenta la chiave pubblica di un server DNS
  - SIG, che contiene la digital signature di una richiesta o risposta
  - NXT, è usato per autenticare risultati negativi, indicando la non esistenza di RR per la risposta (come NXDOMAIN o NOERROR/0)

BGP: I rischi maggiori

Nei punti di peering e Internet Exchange (IX):

- Tutti i providers in un IX tipicamente condividono la stessa infrastruttura (un insieme di switch in HSRP o VRRP) che diventa critica
- Route reflectors e route Servers sono di solito realizzati su hosts UNIX (zebra, rsD, rsNG etc.)
- Le politiche di filtraggio dei pacchetti sono di frequente molto più “rilassate” negli IX
- Le relazioni fra prefissi e AS di origine, l’AS_path, e l’autenticità dei peers BGP non sono tipicamente mai verificate

BGP: gli attacchi più frequenti

- **SYNflood verso la porta 179/tcp usata da BGP**
- **Drop di sessioni BGP attraverso l'invio di RST sulla connessione TCP tramite hijacking della sessione o l'invio di messaggi fittizi OPEN/KEEPALIVE:**
  - Generazione di instabilità e route flapping, innesco dei meccanismi di dampening a scopo DoS
- **Modifica di parametri della sessione, capabilities, MED, communities etc.**
- **Iniezione dolosa di routes tramite tools esistenti:**
  - Annuncio di routes più specifiche in grandi netblocks per aumentare la taglia delle routing tables e creare problemi di memoria sui routers
  - Redirezione di grandi volumi di traffico verso destinazioni "black hole", o a scopo di DoS
  - Creazione di routing loops

BGP: Attacchi DoS - Funziona!

Abbattimento di sessioni

```
Attacker# tcpdump src port 179 or dst port 179
16:22:59.328544 10.0.0.3.179 > 10.0.0.5.32324: P 272350230:272350249(19) ack
4142958006 win 15531: BGP (KEEPALIVE) [tos 0xc0] [ttl 1]
16:22:59.527079 10.0.0.5.32324 > 10.0.0.3.179: . ack
272350249 win 15543 [tos 0xc0] [ttl 1]

Attacker# ./ttt -T 2 -D 10.0.0.5 -S 10.0.0.3 -x 179 -y 32324 -fR
-s 272350249

Nov 1 18:23:13.425: %BGP-5-ADJCHANGE: neighbor 10.0.0.3 Down Peer closed the
session
```

Iniezione di routes

```
Attacker# ./tcphijack -c 10.0.0.5 -s 10.0.0.3 -p 32324 -P test2.txt
tcphijack: listening on eth0.
pcap expression is 'host 10.0.0.5 and 10.0.0.3 and tcp port 32324'.
Press Control-C once for status, twice to exit.
We're sync'd to the TCP conversation. Sending Update.
Done.

2w1d: BGP(0): 10.0.0.5 rcvd UPDATE w/ attr: nexthop 10.0.0.5, origin i, metric 0, path 5
2w1d: BGP(0): 10.0.0.5 rcvd 1.0.0.0/24
```

BGP: cosa tenere sotto controllo

- L’AS_path e tutti i prefissi ricevuti da ISP di livello più alto
- Tutti i prefissi che gli altri ISP ricevono contenenti i propri prefissi (via route servers/looking glasses)
- Frequenti modifiche nei paths (nel best path)
- Propri prefissi che risultano originati da altri AS
- Modifiche corrispondenze ARP di peers BGP
- I log del BGP e tutti I messaggi diagnostici relativi

```
route-views.oregon-ix.net>sh ip bgp 192.133.28.0/24 longer
BGP table version is 14432944, local router ID is 198.32.162.100
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*  192.133.28.0     196.7.106.245                          0 2905 701 3549 137 137 137 i
*                   213.200.87.254         10              0 3257 3549 137 137 137 i
*                   193.0.0.56                             0 3333 1299 137 137 137 i
*                   209.10.12.156           0              0 4513 3549 137 137 137 i
```

BGP: politiche di filtraggio

- Mai accettare, annunciare o ridistribuire prefissi più specifici di /24 o netblocks disaggregati
- Specificare esplicitamente il numero massimo di prefissi accettati *maximum-prefix* (tenendo conto che l'attuale "routing table" conta circa ~140K routes)
- Non limitarsi a filtrare su base AS_path ma anche sulla base di espliciti prefissi
- Impostare espliciti filtri in ingresso e uscita per limitare i prefissi inviati e ricevuti
- Filtrare routes relative a reti riservate o non allocate
- E' necessario annunciare o ricevere la default route ?

BGP: politiche di filtraggio

- Le routes che vanno sempre filtrate (in/out)
  - RFC 1918 (10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16)
  - 0.0.0.0/x, 127.0.0.0/8
  - 169.254.0.0/16 (auto-configurazione, no DHCP)
  - 192.0.2.0/24 (TEST-NET)
  - 192.88.99.0/24 (RFC 3048, usata per 6to4 tunneling)
  - Blocchi riservati per Multicast (D Class) e reti “Martian” (E+)
  - Blocchi riservati IANA/ARIN (bogon networks)

```
router bgp 65282
neighbor 193.206.130.5 remote-as 137
neighbor 193.206.130.5 distribute-list 107 in
neighbor 193.206.130.5 distribute-list 108 out
                       oppure
neighbor 193.206.130.5 prefix-list rfc1918-sua in
neighbor 193.206.130.5 prefix-list rfc1918-sua out

access-list 108 deny ip host 0.0.0.0 any
access-list 108 deny ip 10.0.0.0 0.255.255.255 255.0.0.0 0.255.255.255
access-list 108 permit ip any any

        ip prefix-list rfc1918-sua deny 0.0.0.0/8 le 32
        ip prefix-list rfc1918-sua deny 10.0.0.0/8 le 32
```

BGP: difesa e contromisure

- Logging dettagliato di tutte le transazioni BGP
- Autenticazione MD5 delle sessioni con password differenti su tutti i peers
- Applicazioni di filtri per controlli di congruenza sull'AS_path
- Uso di indirizzi associati a interfacce "loopback" per le sessioni BGP per celare i dettagli delle sessioni stesse
- Protezione delle routes verso i DNS Root Servers con esclusione delle stesse dal processo di route dampening

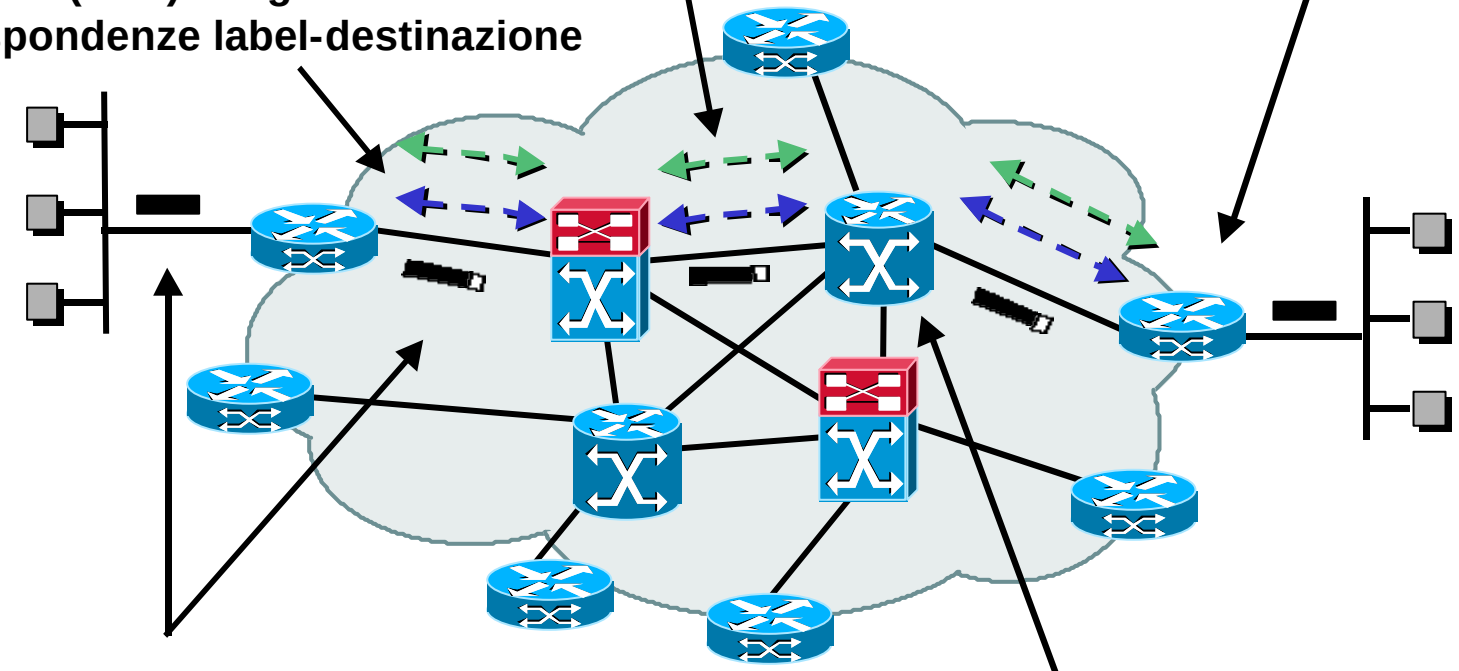
```
router bgp 65000
 no bgp dampening
 bgp dampening route-map dampening-list
 bgp log-neighbor-changes
 network x.x.x.x
 neighbor y.y.y.y remote-as 65001
 neighbor y.y.y.y password <MD5password>
 neighbor y.y.y.y version 4
 neighbor y.y.y.y prefix-list theirnetworks in
 neighbor y.y.y.y prefix-list ournetworks out
 neighbor y.y.y.y maximum-prefix 120000
 neighbor y.y.y.y route-map ourASpath out

ip prefix-list ournetworks seq 5 permit x.x.x.x/y
ip prefix-list ournetworks seq 10 deny 0.0.0.0/0 le 32
ip prefix-list theirnetworks seq 5 permit x.x.x.x/y
ip prefix-list protected-prefixes permit x.x.x.x/y
ip prefix-list <other prefix-list> permit x.x.x.x/y
ip as-path access-list 99 permit ^<AS>( <AS>)*$

route-map ourASpath permit 10
 match as-path 99

route-map dampening-list deny 10
 match ip address prefix-list protected-prefixes
route-map dampening-list permit 20
 match ip address prefix-list <prefix-list>
 set dampening <parametri dampening>
```

MPLS: il paradigma di base

1a. A livello di protocolli IGP (OSPF, ISIS) si Stabilisce la raggiungibilità dele reti destinazione

1b. Attraverso il Label Distribution Protocol (LDP) vengono stabilite le corrispondenze label-destinazione

4. L'egress LER rimuove la label e inoltra i pacchetti



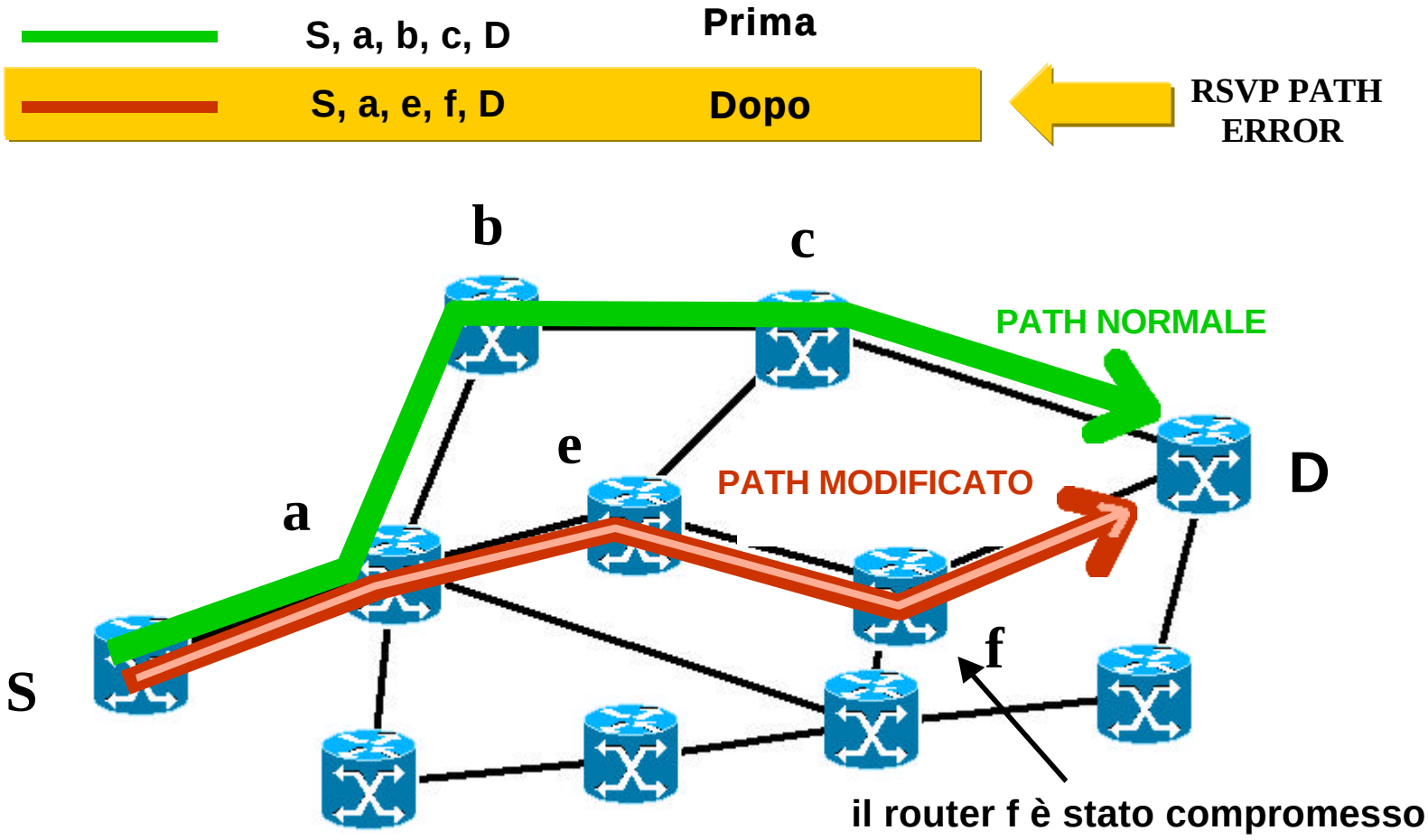
2. L'Ingress LER riceve i pacchetti, realizza funzionalità L3 a valore aggiunto e impone la label

3. I nodi LSR commutano i pacchetti su un LSP lungo la nuvola MPLS effettuando il label-swapping

MPLS: gli attacchi più frequenti

- Iniezione dolosa di pacchetti con Label artefatte:
- Iniezione di dati artefatti nei protocolli di segnalazione e di controllo delle risorse
  - LDP o RSVP per corrompere la logica di propagazione delle label
  - (MP-)BGP e IGPs usati per il resource discovery (OSPF/ISIS) per modificare in maniera dolosa la topologia delle VPN
- Exploit dei meccanismi di FRR (Fast Reroute) e di TE per redirigere il traffico (tramite messaggi RSVP-No Route-PathError contraffatti) verso altri routers del backbone MPLS eventualmente compromessi

Esempio MPLS path error spoofing



MPLS: difesa e contromisure

- Filtraggio via ACL su border router del dominio MPLS (PE)
  - LDP: Porte 646/UDP e 646/TCP
  - RSVP: IP proto 46,134 porte 363/UDP e 1698,1699 TCP/UDP

```
access-list 101 deny 46 any any
access-list 101 deny 134 any any
access-list 101 deny udp any any eq 363
access-list 101 deny udp any any eq 646
access-list 101 deny tcp any any eq 646
access-list 101 deny udp any any range 1698 1699
access-list 101 deny udp any any range 1698 1699
```

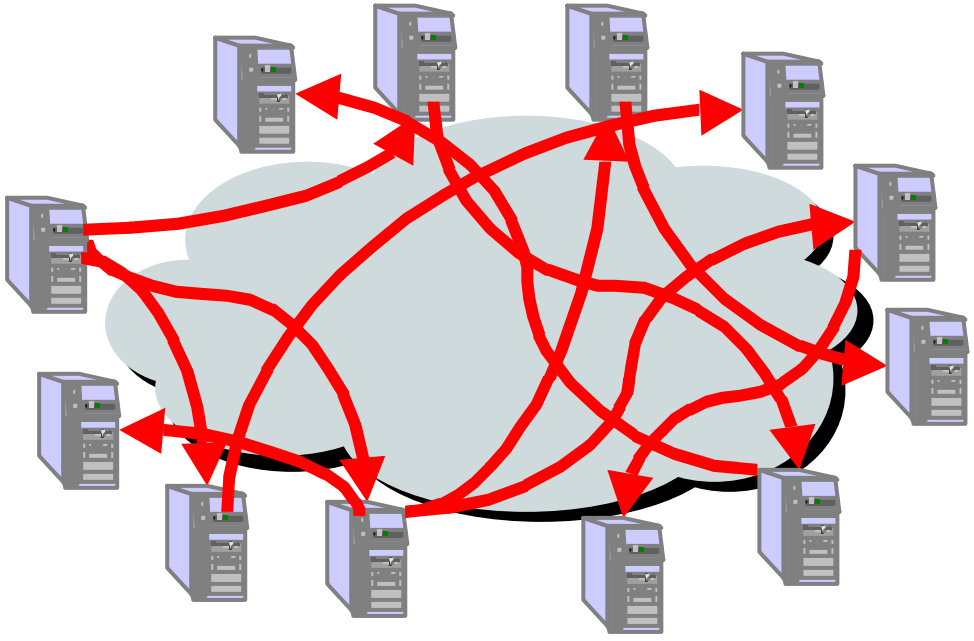
- Il dominio MPLS deve fermarsi ai router PE e mai arrivare ai router CE
- Tutti i protocolli di routing usati a scopo di segnalazione (MP-BGP per VPN) e resource discovery per Traffic Engineering devono prevedere sessioni autenticate (possibilmente) con MD5

```
interface xy
!ip ospf authentication-key <key>
 ip ospf message-digest-key 1 md5 <key>
router ospf 1
 area 0 authentication [message-digest]
```

```
interface xy
 isis password <password> level-<z>

router isis
 domain-password <password>
 area-password <password>
```

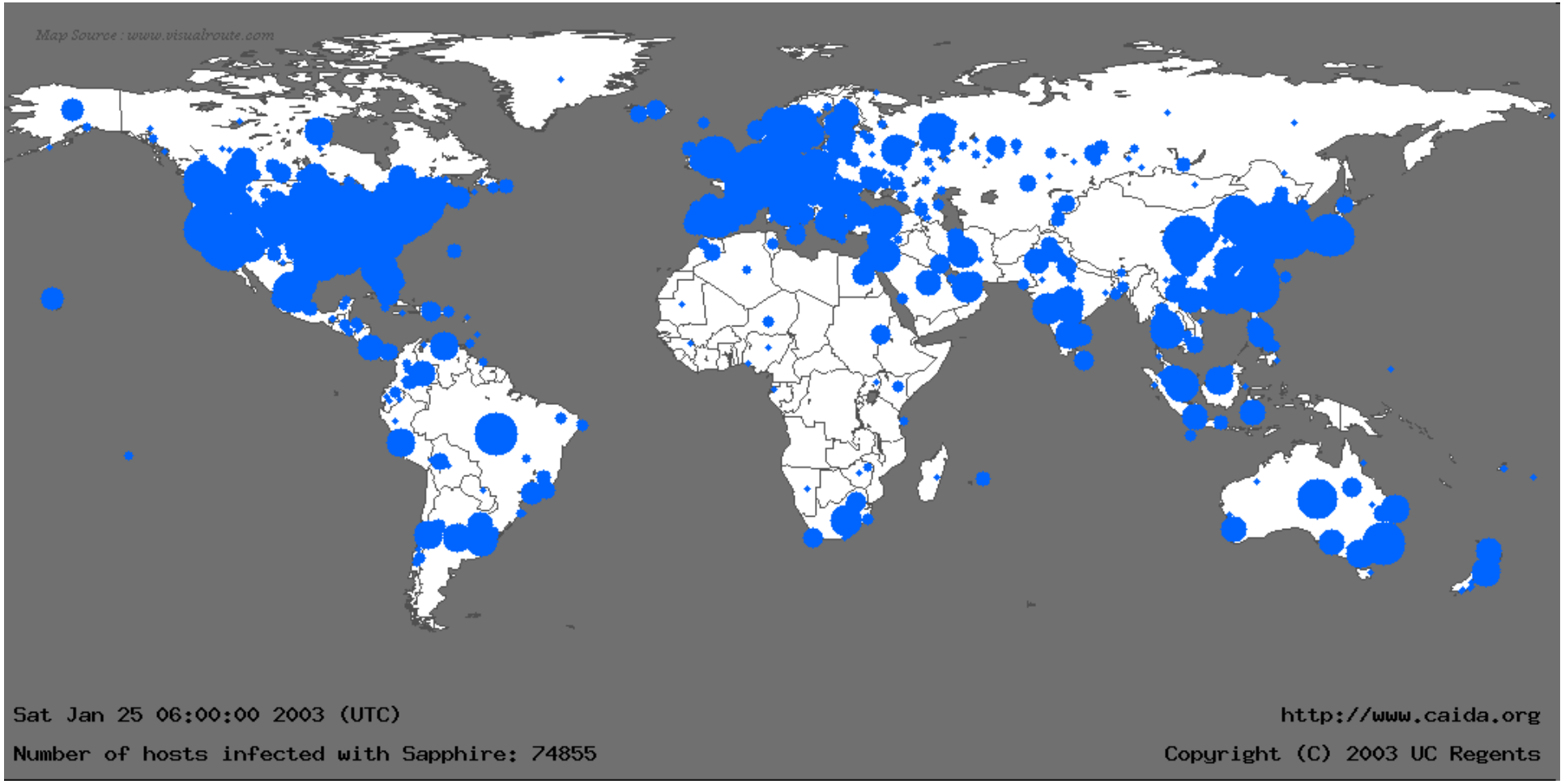
Worms e Virus

- Codice autoreplicante che effettua scansioni a tappeto sulla rete e si propaga autonomamente sugli hosts vittima sfruttando vulnerabilità note, infestando gli stessi e riattivandosi in molteplici copie. In questo i worms si contraddistingono dai Virus che si agganciano ad altro codice “ospite” del cui avvio si servono per avviare la propagazione.



Worms

- Hanno effetti devastanti sulla stabilità della rete Internet proporzionalmente alla loro velocità di propagazione

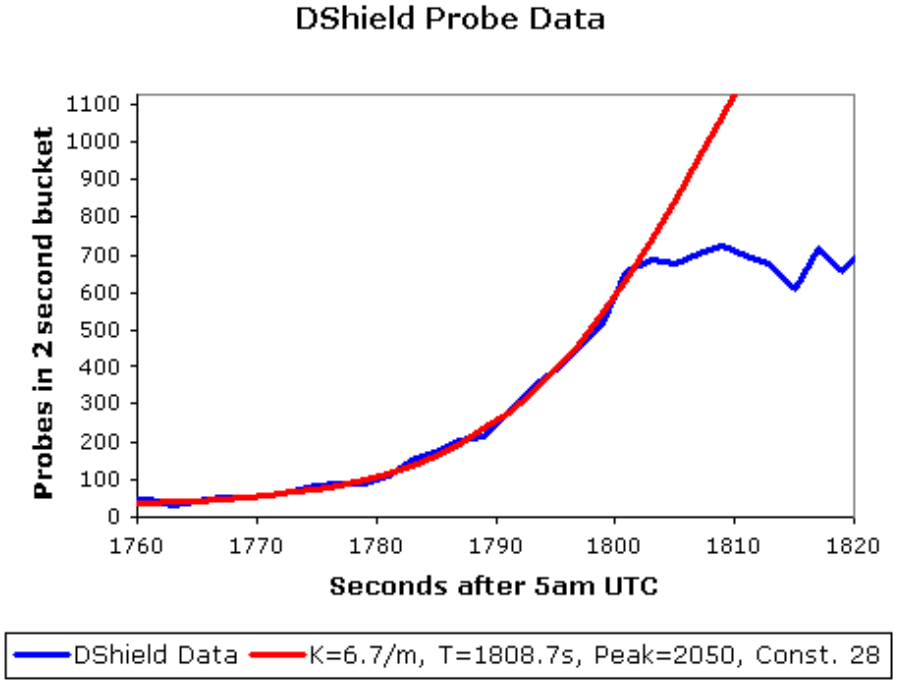


I più recenti WORMs

- Code Red 1 e 2
- Code Blue
- Nimda
- SQL Slammer
- Blaster
- Nachi/Welchia

- 1 min: scansione massiva (55 milioni di IP/sec)
- < 2 min: saturazione totale banda di accesso
- < 15 min: completata scansione del 90% di Internet
- < 40 min: infettati 100k hosts

- Sfruttano vulnerabilità note dei sistemi operativi più diffusi (MSWindows, Linux, etc)
- Spesso si propagano massivamente via e-mail



Individuazione worms

- Presenza di un numero anomalo di flussi particolari di traffico

```
Router>show ip cache flow | include 0000 0800
SrcIf SrcIPaddress DstIf DstIPaddress Pr SrcP DstP Pkts

Fa2/0 XX.XX.XX.242 Fa1/0 XX.XX.XX.119 01 0000 0800 1
Fa2/0 XX.XX.XX.242 Fa1/0 XX.XX.XX.169 01 0000 0800 1
Fa2/0 XX.XX.XX.204 Fa1/0 XX.XX.XX.63  01 0000 0800 1
Fa2/0 XX.XX.XX.204 Fa1/0 XX.XX.XX.111 01 0000 0800 1
Fa2/0 XX.XX.XX.204 Fa1/0 XX.XX.XX.95  01 0000 0800 1
Fa2/0 XX.XX.XX.204 Fa1/0 XX.XX.XX.79  01 0000 0800 1
```

- Carico di CPU anomalo su routers e servers

```
Router#sh proc cpu
CPU utilization for five seconds: 99%/46%; one minute: 78%; five minutes: 37%
 PID Runtime(ms)   Invoked      uSecs   5Sec   1Min   5Min TTY Process
   1       48424    104314        464  0.00%  0.00%  0.00%   0 Load Meter
      ……
  11    39654652   5944427       6670  1.46%  1.87%  2.73%   0 ARP Input
      ……
  29    16207464   5400436       3001 52.85% 38.51% 13.57%   0 IP Input
      ……
  30           0         1          0  0.00%  0.00%  0.00%   0 ICMP event handl
  80        2444       204      11980  0.89%  0.74%  0.47%  67 Virtual Exec
```

- Attività anomala nei logs dei servers con presenza di specifiche signatures che individuano la presenza dei worms

Signature: Nimda/W32

```
GET /scripts/root.exe?/c+dir
GET /MSADC/root.exe?/c+dir
GET /c/winnt/system32/cmd.exe?/c+dir
GET /d/winnt/system32/cmd.exe?/c+dir
GET /scripts/..%5c../winnt/system32/cmd.exe?/c+dir
GET /_vti_bin/..%5c../..%5c../..%5c../winnt/system32/cmd.exe?/c+dir
GET /_mem_bin/..%5c../..%5c../..%5c../winnt/system32/cmd.exe?/c+dir
GET /msadc/..%5c../..%5c../..%5c/..\xc1\x1c../..\xc1\x1c../..\xc1\x1c../
winnt/system32/cmd.exe?/c+dir
GET /scripts/..\xc1\x1c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..\xc0/../winnt/system32/cmd.exe?/c+dir
GET /scripts/..\xc0\xaf../winnt/system32/cmd.exe?/c+dir
GET /scripts/..\xc1\x9c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..%35c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..%35c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..%5c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..%2f../winnt/system32/cmd.exe?/c+dir
```

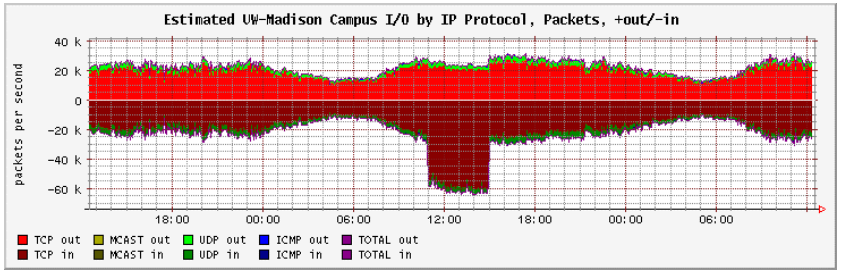
Individuazione Worm dai volumi di traffico

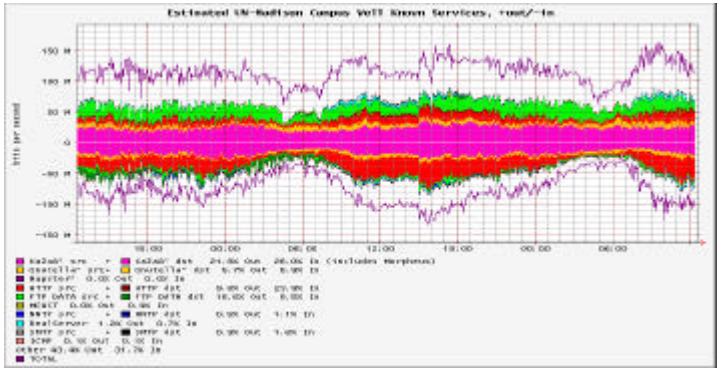
Riscontro di eventuali anomalie da DoS nei volumi e nei pattern di traffico

- **TCP : ~90 % (HTTP, FTP e P2P)**
- **UDP : ~10 % (DNS, SNMP, streaming)**
- **ICMP : <1 % - IGMP : <1 %**

```
router>sh ip ca fl
Protocol         Total    Flows   Packets Bytes  Packets Active(Sec) Idle(Sec)
--------         Flows     /Sec     /Flow  /Pkt     /Sec      /Flow     /Flow
TCP-FTP       15012652      6.8         4   177     30.7        3.2      10.6
TCP-WWW      678916247    310.1         5   491   1676.6        1.3      12.0
TCP-SMTP      13324793      6.0        19   466    118.8        5.2      11.7
TCP-other   1227556769    560.8        19   638  10925.9        6.0      17.1
ICMP          36738413     16.7         5    62     94.9        5.1      19.1
IGMP               369      0.0         1   228      0.0        0.1      16.3
Total:       2649573771   1210.4        11   598  14444.4        4.1      16.3
```

Strumenti come MRTG, RRDtool e Netflow possono essere usati per individuare facilmente variazioni ed anomalie nei trend di traffico





Prevenzione WORMS - NBAR

- NBAR (Network Based Application recognition), disponibile su routers Cisco dalla release 12.1(5)T consente la classificazione content-based del traffico

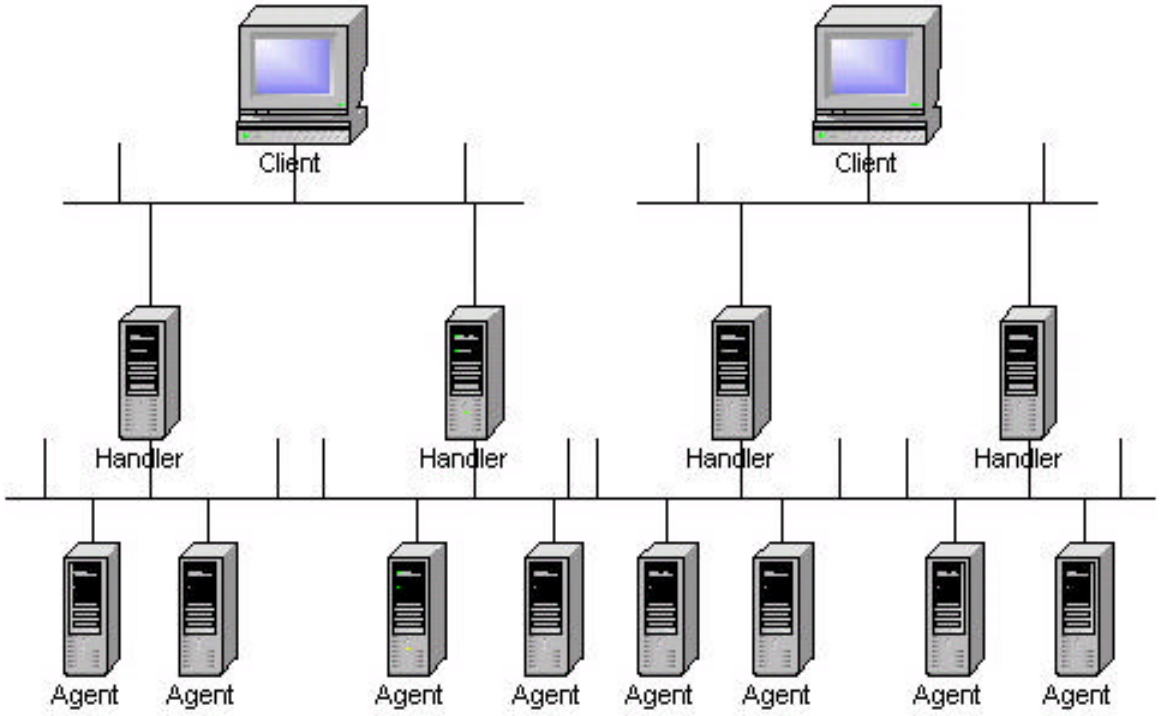
```
! Matching del traffico basato su signature
class-map match-any http-hacks
 match protocol http url “*cmd.exe*”

! Policy map per marcare il traffico entrante
policy-map mark-inbound-http-hacks
 class http-hacks
 set ip dscp 1

! Applica la policy di marking
 int xy
  service-policy input mark-inbound-http-hacks

! Blocca il traffico marcato via ACL
 access-list 100 deny ip any any dscp 1 log
 access-list 100 permit ip any any

! Applica l’ ACL all’interfaccia protetta
 int xy
  ip access-group 100 out
```

Prevenzione WORMS - NBAR

- **Restrizioni e limitazioni**
  - Supporta fino a 24 matches distinti fra URLs, hosts e MIME types
  - Nel matching non può andare oltre I primi 400 bytes in ogni URL
  - Non può trattare pacchetti frammentati, nè il traffico cifrato (HTTPS) nè i pacchetti locali al router
  - Non supporta la codifica Unicode (UTF-8/%u)
- **Implicazioni funzionali e prestazionali**
  - Comporta un aggravio di CPU ~20%
  - Richiede obbligatoriamente il CEF
  - Effettua il 3-way handshake in modalità proxy
- **E’ opportuno un tuning delle risorse usate**

```
ip nbar resources 600 1000 50
scheduler allocate 30000 2000
```

Distributed Denial of Service



Il **Client** controlla
E attiva l'attacco

Gli **Handler** sono host compromessi che Controllano gli **agents** Schermando I clients

Gli **Agents** sono host compromessi che hanno il compito di realizzare effettivamente gli attacchi

Distributed Denial of Service

Le fasi e la dinamica di un DDoS

- **Scansione di decine di migliaia di hosts per l'individuazione di vulnerabilità note e sfruttabili**
- **Exploit delle vulnerabilità a scopo di compromissione degli host conquistandone l'accesso**
- **Installazione dei tools per la realizzazione del DDoS**
- **Sfruttamento degli hosts conquistati come base di partenza per ulteriori scansioni e compromissioni reiterando il punto 3**
- **Una volta installati i DDoS tools su un numero sufficiente di hosts si procede all'avvio dell'attacco attivando handlers e agents a partire da un client remoto**

Distributed Denial of Service

Caratterizzazione e tipologie

Esiste un certo numero di DDoS tools caratterizzati dalle tecniche di distribuzione dell'attacco fra clients, agent, handlers, dalle porte di default (che possono comunque variare) e dai meccanismi usati per la loro comunicazione

Trinoo	**1524 tcp** **27665 tcp** **27444 udp** **31335 udp**
TFN	**ICMP ECHO/ICMP ECHO REPLY**
Stacheldraht	**16660 tcp** **65000 tcp** **ICMP ECHO/ICMP ECHO REPLY**
TFN2K	**Specificata a runtime o scelta random come combinazione di pacchetti UDP, ICMP and TCP**

Tutte le tecniche in questione realizzano replicatamente attacchi DoS classici (ICMP Bombing, SYN-Flood, Smurfing etc)

Distributed Denial of Service

Tecniche di difesa e contromisure

- Abilitazione di CEF e Unicast Reverse path Forwarding

```
ip verify unicast reverse-path
```

- Applicazione dei filtri anti-spoofing in ingresso e in uscita

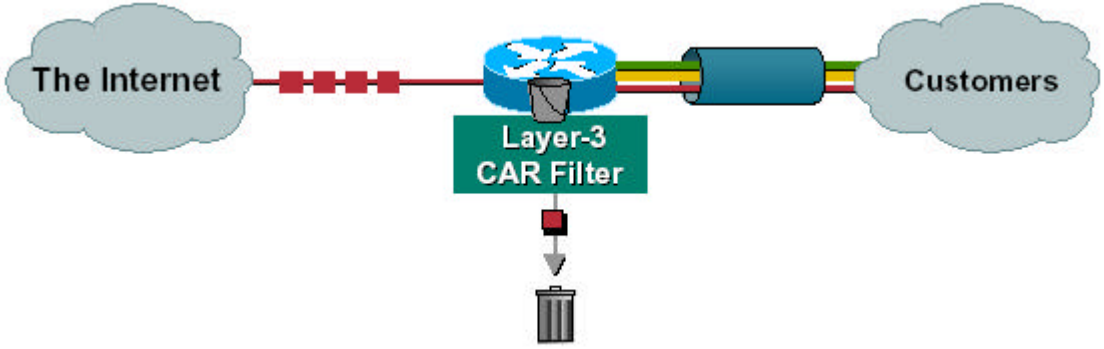
```
access-list 110 deny ip 165.21.0.0  0.0.255.255   any log
access-list 110 permit ip any any
access-list 111 permit ip 165.21.0.0 0.0.255.255 any
access-list 111 deny ip any any log
```

- Limitazione in banda dei flussi di traffico ICMP e relativi ai SYN

```
access-list 102 permit icmp any any
access-list 103 deny tcp any any established
access-list 103 permit tcp any any
interface Serial3/0/0
        rate-limit input access-group 102 256000 8000 8000
conform-action transmit exceed-action drop
        rate-limit input access-group 103 256000 8000 8000
conform-action transmit exceed-action drop
```

TCP SYN Flooding – Filtraggio in banda

E’ possibile reagire attivamente durante un attacco di tipo “SYN flooding” per ridurne drasticamente l’impatto, limitando in banda il flusso di traffico offensivo tramite la QoS facility “Committed Access Rate” (CAR) integrata nell’ambito dei meccanismi CEF e “DISTRIBUTED CEF”



Non influenzare le sessioni TCP già completamente stabilite

```
access-list 103 deny tcp any host 10.0.0.1 established
```

Limita in banda tutto il restante traffico (le sessioni in SYN)

```
access-list 103 permit tcp any host 10.0.0.1
```

Applica il filtro in banda (8Kbps) sulla border interface

```
interface Serial3/0/0
     rate-limit input access-group 103 8000 8000 8000
     conform-action transmit exceed-action drop
```

Prevenzione avanzata: BGP/Null

Su tutti i router del proprio AS aggiungere una static route alla null interface per un’apposito ip di una rete di test

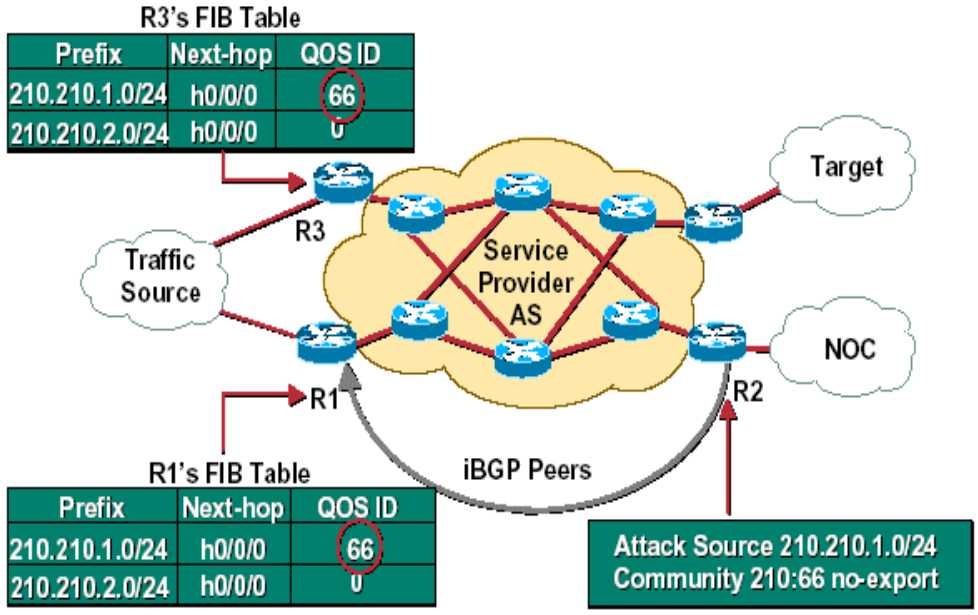
A livello di router principale di accesso al backbone assegna come next-hop l’IP di cui sopra alla rete da bloccare per DoS e ridistribuire via BGP il tutto agli altri router di backbone dell’AS ma non agli altri peers esterni. La funzionalità di unicast RPF bloccherà a livello di line card e non di RProc il traffico di DoS.

```
router bgp <AS>
 network <sourceOfDDOS> mask <netmask> route-map ddos-nh

route-map ddos-nh
 set ip next-hop <TEST-NETIPaddr>
ip route <TEST-NET> 255.255.255.0 Null0
```

Prevenzione avanzata BGP/CAR/FIB

Attraverso uno speciale valore della community marca le reti origine del traffico pericoloso, da limitare in banda, a livello di router principale di accesso al backbone e succesivamente propaga tale community sui tuoi peers

```
router bgp <AS>
 network <destDDOS> mask <netmask>
 neighbor x.x.x.x route-map ddos-rl out
 neighbor x.x.x.x send community
access-list 10 permit <destDDOS>
route-map ddos-rl
 match ip address 10
 set community <AS>:66 no-export
ip route <destDDOS> 255.255.255.0 Null0
```



Prevenzione avanzata BGP/CAR/FIB

Su ciascun router dell'AS rimappa la QoSID nella FIB in base allo specifico valore della community e applica eventuali limitazioni in banda sulla base della QoSid

```
router bgp <AS>
 table-map ddos-rl
ip community list 1 permit <AS>:66
route-map ddos-rl
 match community 1
 set ip qos-group 66
interface xy
 bgp-policy source ip-qos-map
 rate-limit input qos-group 66 ...
```

Caratterizzazione DoS via ACL

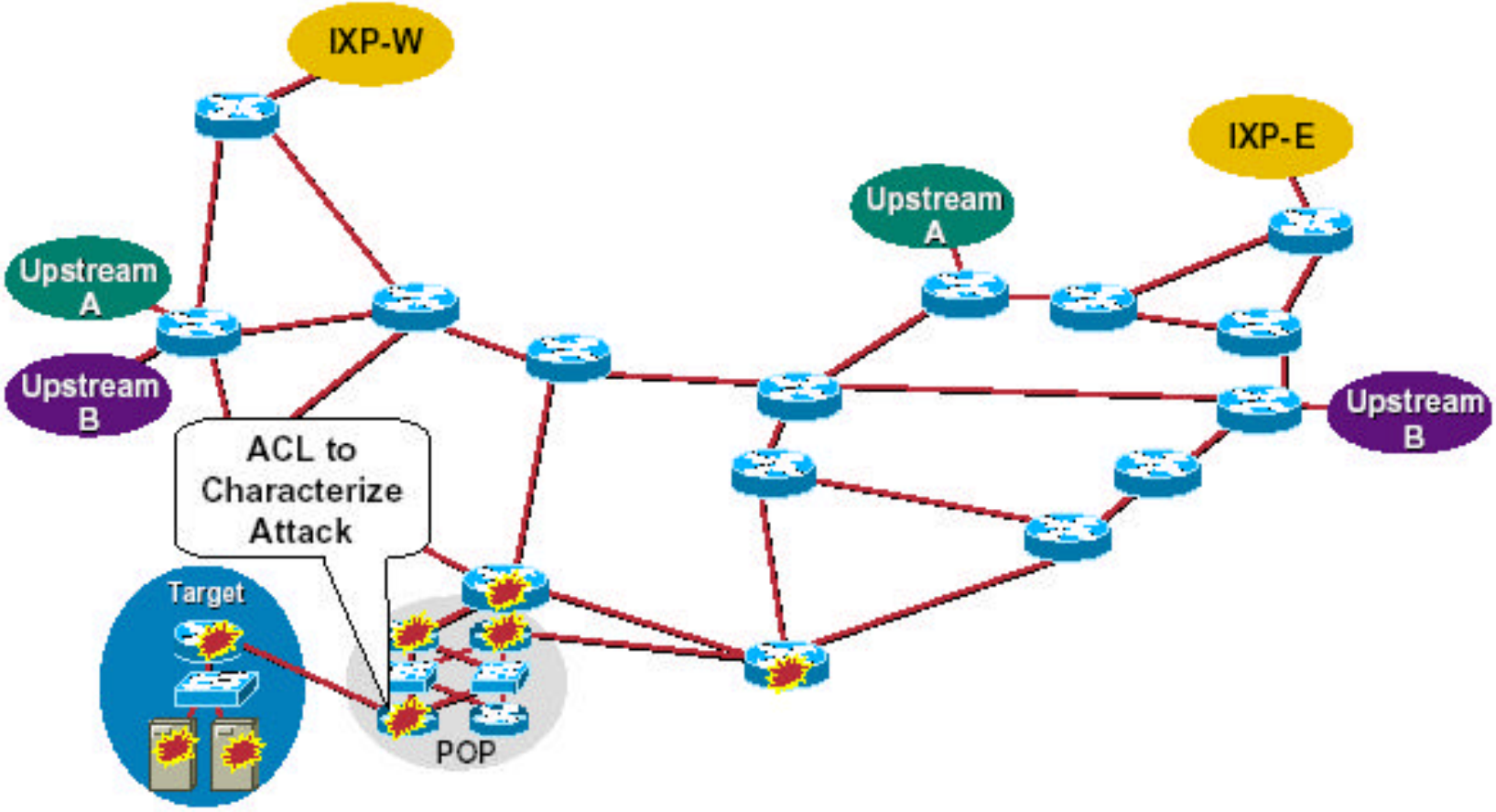
In assenza di un'analizzatore di protocollo o di uno sniffer è ugualmente possibile individuare e caratterizzare i principali attacchi di tipo DoS in corso attraverso l'analisi dei "firing counters" di un ACL "di servizio" opportunamente costruita allo scopo:

```
access-list 169 permit icmp any any echo
access-list 169 permit icmp any any echo-reply log-input
access-list 169 permit udp any any eq echo
access-list 169 permit udp any eq echo any
access-list 169 permit tcp any any established
access-list 169 permit tcp any any
access-list 169 permit ip any any
```

```
# show access-list 169
Extended IP access list 169
permit icmp any any echo (2 matches)
permit icmp any any echo-reply (21374 matches)
permit udp any any eq echo
permit udp any eq echo any
permit tcp any any established (150 matches)
permit tcp any any (15 matches)
permit ip any any (45 matches)
```

Caratterizzazione DoS via ACL

- Le ACL vanno predisposte quanto più possibile prossime all'obiettivo dell'attaco



Caratterizzazione DoS via ACL

Smurfing: Vittima

Il numero di echo-reply ricevuti è elevatissimo rispetto a quello dei request

```
# show access-list 169
…
permit icmp any any echo (2145 matches)
permit icmp any any echo-reply (213746421 matches)
…
```

Gli indirizzi sorgente degli echo reply sono raggruppabili in un insieme limitato di origini che individuano gli amplificatori o “reflectors”

```
# show log
%SEC-6-IPACCESSLOGDP: list 169 denied icmp 192.168.45.142
(Serial0 *HDLC*) -> 16.2.3.7 (0/0), 1 packet
%SEC-6-IPACCESSLOGDP: list 169 denied icmp 192.168.45.142
(Serial0 *HDLC*) -> 16.2.3.7 (0/0), 1 packet
%SEC-6-IPACCESSLOGDP: list 169 denied icmp 192.168.212.72
(Serial0 *HDLC*) -> 16.2.3.7 (0/0), 1 packet
%SEC-6-IPACCESSLOGDP: list 169 denied icmp 192.168.212.72
(Serial0 *HDLC*) -> 16.2.3.7 (0/0), 1 packet
```

Caratterizzazione DoS via ACL

Smurfing: Amplificatore

Il numero di echo-request ricevuti è elevatissimo rispetto a quello dei reply

```
# show access-list 169
permit icmp any any echo (214576534 matches)
permit icmp any any echo-reply (4642 matches)
```

Gli indirizzi di destinazione degli echo request individuano dei broadcast diretti ed in genere riportano come sorgente sempre lo stesso indirizzo

```
# show log
%SEC-6-IPACCESSLOGDP: list 169 denied icmp 192.168.45.142
(Serial0 *HDLC*) -> 16.2.3.255 (0/0), 1 packet
%SEC-6-IPACCESSLOGDP: list 169 denied icmp 192.168.45.142
(Serial0 *HDLC*) -> 16.2.3.255 (0/0), 1 packet
```

Si riscontra un elevato numero di broadcast sulla LAN interna

```
# show int fast 4/0/0
FastEthernet4/0/0 is up, line protocol is up
...
     442344667 packets input, 3565139278 bytes, 0 no buffer
     Received 1247787654 broadcasts, 0 runts, 0 giants, …
```

Caratterizzazione DoS via ACL

Fraggle: Vittima

Il numero di udp echo-reply ricevuti è elevatissimo rispetto a quello dei request

```
# show access-list 169
…
permit udp any any eq echo (9845 matches)
permit udp any eq echo any (1374421 matches)
…
```

Gli indirizzi sorgente degli echo reply sono raggruppabili in un insieme limitato di origini che individuano gli amplificatori o “reflectors”

```
# show log
%SEC-6-IPACCESSLOGDP: list 169 denied udp 192.168.45.142
(Serial0 *HDLC*) -> 16.2.3.7 (0/0), 1 packet
%SEC-6-IPACCESSLOGDP: list 169 denied udp 192.168.45.142
(Serial0 *HDLC*) -> 16.2.3.7 (0/0), 1 packet
%SEC-6-IPACCESSLOGDP: list 169 denied udp 192.168.212.72
(Serial0 *HDLC*) -> 16.2.3.7 (0/0), 1 packet
%SEC-6-IPACCESSLOGDP: list 169 denied udp 192.168.212.72
(Serial0 *HDLC*) -> 16.2.3.7 (0/0), 1 packet
```

Caratterizzazione DoS via ACL

Fraggle: Amplificatore

Il numero di udp echo-request ricevuti è elevatissimo rispetto a quello dei reply

```
# show access-list 169
permit udp any any eq echo (45653 matches)
permit udp any eq echo any (64 matches)
```

Gli indirizzi di destinazione degli echo request individuano dei broadcast diretti ed in genere riportano come sorgente sempre lo stesso indirizzo

```
# show log
%SEC-6-IPACCESSLOGDP: list 169 denied udp 192.168.45.142
(Serial0 *HDLC*) -> 10.2.3.255 (0/0), 1 packet
%SEC-6-IPACCESSLOGDP: list 169 denied udp 192.168.45.142
(Serial0 *HDLC*) -> 10.2.3.255 (0/0), 1 packet
```

Si riscontra un elevato numero di broadcast sulla LAN interna

```
# show ip traffic
IP statistics:
…
Bcast: 1147598643 received, 65765 sent
  Mcast: 188967 received, 459190 sent
```

Caratterizzazione DoS via ACL

SYN Flood

Il numero di pacchetti relativi alla fase di 3-way handshake (seconda linea) supera abbondantemente quello di pacchetti su connessioni già stabilite

```
# show access-list 169
…
permit tcp any any established (150 matches)   [socket stabilite]
permit tcp any any (3654 matches)              [socket in syn]
…
```

È inoltre possibile constatare dall'output delcomando **`show log`** la presenza di indirizzi sorgente non validi, oggetto di spoofing.

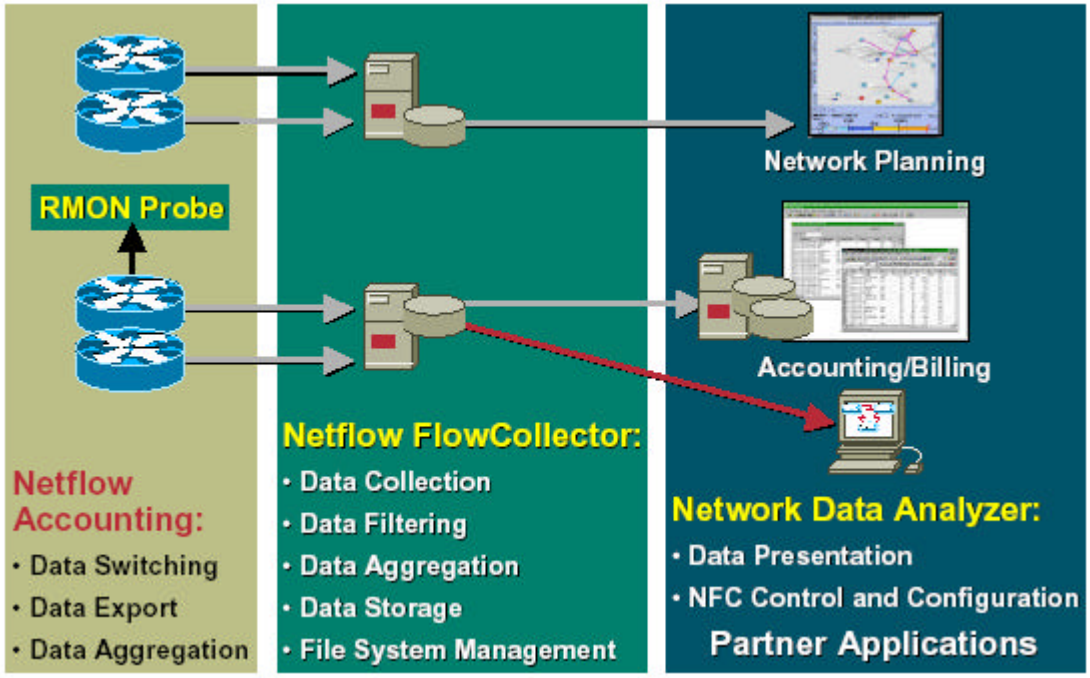
Ping Flood

Il numero di echo-request e reply ricevuti è elevato con i request che in genere superano i reply. Gli indirizzi sorgente non sono oggetto di spoofing.

```
# show access-list 169
…
permit icmp any any echo (214576534 matches)
permit icmp any any echo-reply (4642 matches)
…
```

Caratterizzazione DoS via Netflow

- Tutti i dati di accounting (flussi di traffico, protocolli etc.) possono essere raccolti ed inviati periodicamente da ciascun router a un apposito data-collector per successive analisi



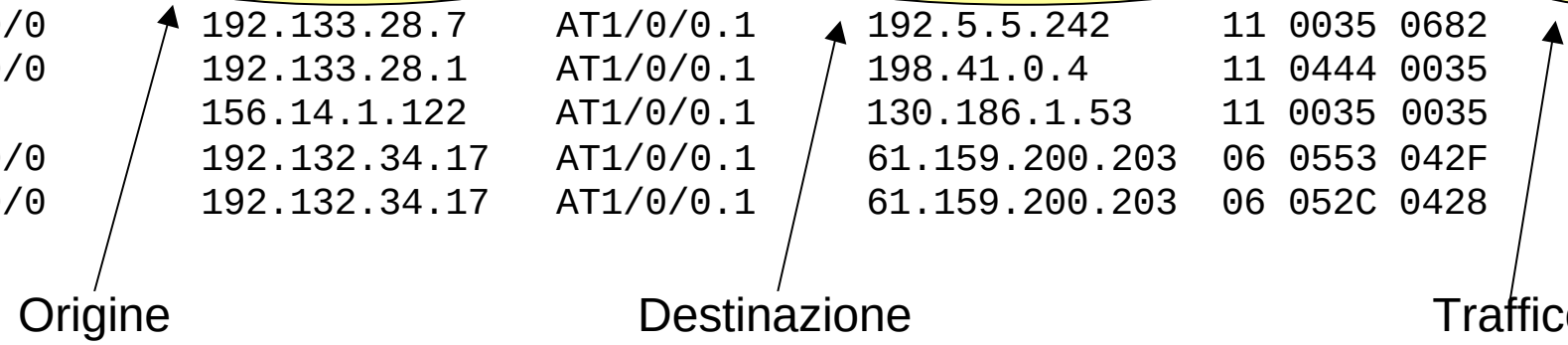
- Abilitazione Netflow

```
ip flow-export version 5 origin-as
ip flow-export destination x.x.x.x

interface xy
 ip route-cache flow
```

Individuazione DoS via netflow

E’ possibile individuare la presenza e gli estremi di un DoS in atto attraverso l’analisi della netflow cache riscontrando flussi anomali di traffico che si discostano in maniera evidente dal modello di baseline

```
#show ip cache flow
…
SrcIf         SrcIPaddress    DstIf         DstIPaddress    Pr SrcP DstP  Pkts
Fa4/0/0       192.132.34.17   AT1/0/0.1     148.240.104.176 06 080C 1388     1
Fa4/0/0       192.132.34.17   AT1/0/0.1     63.34.210.22    06 0AEB 0666    15K
Fa4/0/0       192.133.28.1    Fa4/0/0       143.225.219.187 11 0035 9F37     1
Fa4/0/0       192.132.34.17   AT1/0/0.1     216.207.62.22   06 0FD2 0578  7195
Fa4/0/0       143.225.231.7   AT1/0/0.1     143.225.255.255 11 007F 007D     1
Fa4/0/0       192.132.34.17   AT1/0/0.1     148.240.104.176 06 0015 1381    13
Fa4/0/0       192.132.34.17   AT1/0/0.1     148.240.104.176 06 0015 1382    12
Fa4/0/0       192.133.28.7    AT1/0/0.1     164.124.101.44  11 0035 0035     2
Fa4/0/0       143.225.209.72  AT1/0/0.1     209.178.128.121 01 0000 0000   561K
Fa4/0/0       192.133.28.7    AT1/0/0.1     192.5.5.242     11 0035 0682     1
Fa4/0/0       192.133.28.1    AT1/0/0.1     198.41.0.4      11 0444 0035     1
Se6/7         156.14.1.122    AT1/0/0.1     130.186.1.53    11 0035 0035     1
Fa4/0/0       192.132.34.17   AT1/0/0.1     61.159.200.203  06 0553 042F    75
Fa4/0/0       192.132.34.17   AT1/0/0.1     61.159.200.203  06 052C 0428    12K
…
```



Domande?