

IV Giornata Nazionale del Software Libero

- Inizieremo con un breve cenno sulle reti, come funziona la trasmissione dei dati e su quali regole si fonda.
- Quindi passeremo rapidamente ad analizzare i problemi legati alla sicurezza di rete.
- Infine, cercheremo di comprendere come proteggere le nostre macchine dai possibili rischi.



LE RETI DI CALCOLATORI E “LA RETE DELLE RETI”

- 1962 idea di una rete mondiale
- 1969 Il Ministero della Difesa Statunitense crea l'ARPA incaricata di sviluppare una rete in grado di resistere ad una guerra nucleare; furono collegati quattro grandi computer nelle università del sud-ovest degli Stati Uniti.
- 1972 nasce il Transmission Control Protocol/Internet Protocol (TCP/IP).
- 1984 Viene introdotto il Domain Name System (DNS)
- 1985 Internet mette in contatto una larga comunità di ricercatori e sviluppatori
- 1989 Viene formato il RIPE
- 1993 I media cominciano a prendere notizie da Internet, la Casa Bianca e le Nazioni Unite vanno online.
- 1994 Netscape rende disponibili le prime copie del suo Netscape Navigator per il download attraverso Internet.
- 1995 Vengono fondati fornitori di accesso ad Internet (Provider) quali CompuServe, AOL e Prodigy.
- 1996 vengono censiti 100 000 siti Web, nel 1998 salgono a 3.7 milioni
- Oggi, solo in Italia, si contano circa 2,7 milioni di famiglie con un collegamento internet in casa.



MA COME FUNZIONA UNA RETE DI CALCOLATORI?

- ogni interfaccia di rete ha un proprio indirizzo;
- un'interfaccia di rete di un elaboratore può comunicare con un'interfaccia di un altro elaboratore solo se queste sono fisicamente connesse alla stessa rete;
- un'interfaccia di rete di un elaboratore può comunicare con un'interfaccia di un altro elaboratore solo se gli indirizzi di queste interfacce appartengono alla stessa rete.

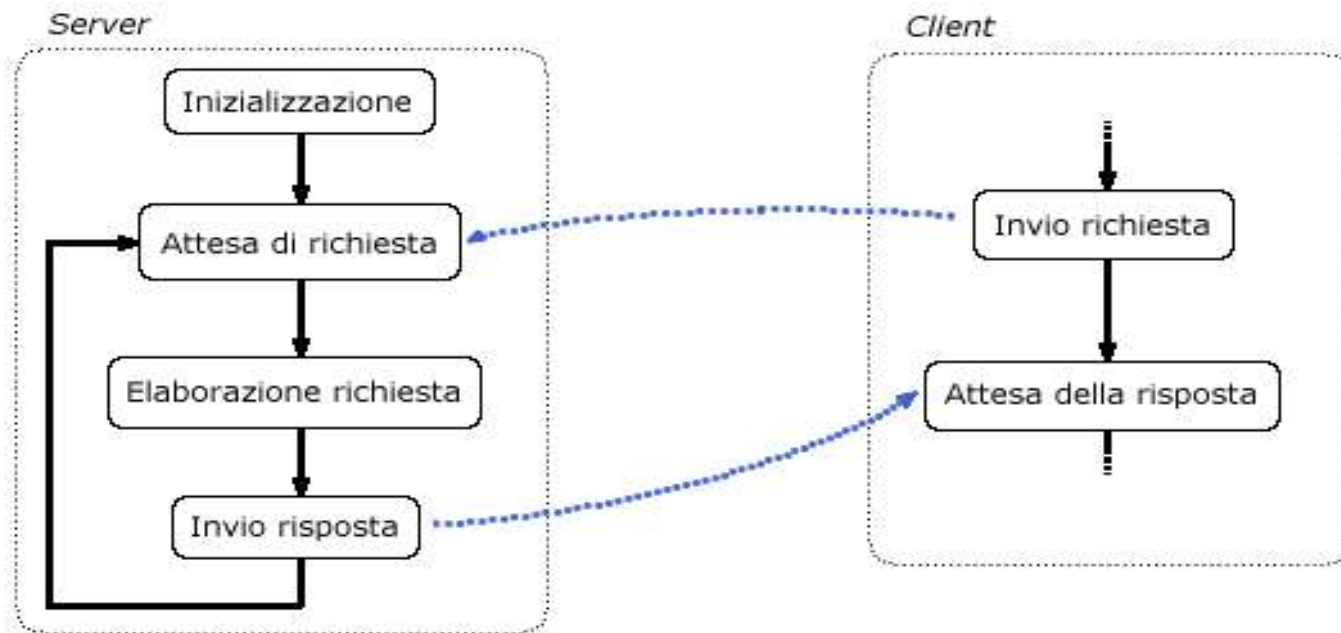
Un'applicazione in esecuzione su di un sistema potrà utilizzare un'interfaccia di rete per comunicare con un'altra applicazione che è in esecuzione, in generale, su un altro sistema.

In particolare si distinguono due categorie di applicazioni che utilizzano la comunicazione via rete: le applicazioni di tipo client e quelle di tipo server.



MA COME FUNZIONA UNA RETE DI CALCOLATORI?

Funzionamento Client-Server



MA COME FUNZIONA UNA RETE DI CALCOLATORI?

Anche le macchine possono essere divise in tre categorie:

- Client: richiedono un servizio – macchine utilizzate per lavorare (si parla infatti anche di workstation) e per utilizzare i servizi messi a disposizione dai server presenti sulla rete.
- Server: forniscono un servizio - generalmente si tratta di macchine con hardware in grado di offrire prestazioni elevate (bassi tempi di accesso ai dischi, grande quantità di memoria centrale, ...)
- Router: instradano i collegamenti.

Più semplicemente un **servizio** è una richiesta di dati, e la rete serve a trasferirli.



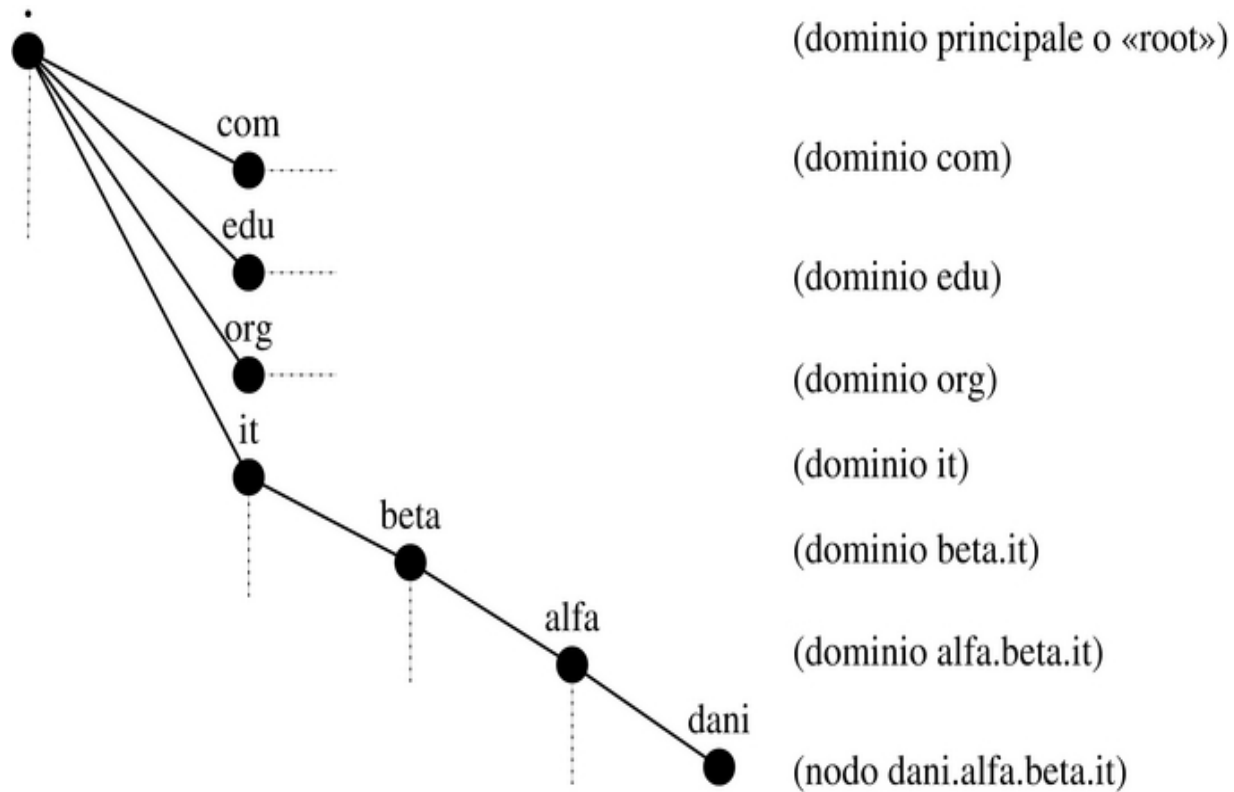
INDIRIZZAMENTO

- In internet, quando ci colleghiamo col nostro modem, ci viene dato un indirizzo IP (Es:212.100.231.233), a consegnarcelo è un server DHCP. Grazie a questo indirizzo siamo riconosciuti in modo univoco e possiamo accedere alle risorse dei server.
- Anche i server hanno degli indirizzi, dietro al nostro sito preferito c'è un indirizzo IP.
- Siccome ricordarsi gli IP è praticamente impossibile, quando ci colleghiamo ad un sito scriviamo un nome, ad esempio www.yahoo.com
- Esistono quindi dei server (i DNS), che associano ad ogni host l'indirizzo IP appropriato.



DNS

Il DNS



LA SUITE DI PROTOCOLLI TCP/IP

E' la base di tutti i protocolli utilizzati sui nostri pc casalinghi per accedere ad internet.

Il costituente base è il pacchetto di trasmissione, l'unità elementare di comunicazione che formato da due parti fondamentali:

- L'intestazione
- I dati (il *payload*)

Nell'intestazione vi sono tutte le informazioni che consentono il transito e l'instradamento del pacchetto, il payload contiene i dati da trasmettere.



LA SUITE DI PROTOCOLLI TCP/IP

IP (Internet Protocol)

E' specificato dalle RFC 790 e 791 e rappresenta il protocollo di base dello stack TCP/IP (Internet suite). Fornisce un servizio di consegna dei pacchetti, detti IP datagram, inaffidabile (unreliable), che esegue le operazioni necessarie al trasporto dell'informazione (best-effort), senza connessione (connectionless).

Per inaffidabile si intende che il protocollo, sebbene gestisca la trasmissione dei pacchetti, non ha alcun meccanismo in grado di verificare che un pacchetto IP arrivi correttamente a destinazione. Se qualcosa va storto, il protocollo IP si limita a scartare il pacchetto ricevuto e tenta di inviare un messaggio ICMP al mittente per informarlo dell'accaduto. Il protocollo IP è inoltre senza connessione, ovvero non mantiene nessuna informazione relativa allo stato della comunicazione tra due interfacce: ogni pacchetto viene gestito indipendentemente dagli altri. Potrebbe succedere che due pacchetti (pacchetto A e pacchetto B) inviati in sequenza giungano a destinazione nella sequenza inversa (pacchetto B, pacchetto A) per il fatto che ognuno di essi può seguire percorsi diversi nella rete. L'ordine dei pacchetti non viene ripristinato dal protocollo IP, ma anche questo è compito dei protocolli di livello superiore.



LA SUITE DI PROTOCOLLI TCP/IP

TCP (Transmission Control Protocol)

Serve a garantire affidabilità e certezza di arrivo dei dati. Ad esempio è usato per la navigazione web con il protocollo HTTP, o con il trasferimento file con il protocollo FTP.

Per scambiare dati in modo affidabile viene utilizzato il protocollo TCP che effettua (senza che sia l'applicazione a occuparsi di questo) il controllo della connessione, la gestione degli errori di trasmissione, la ricostruzione della giusta sequenza dei pacchetti.

La connessione TCP ha tre fasi principali:

- **l'avvio della connessione,**
- **lo scambio dei dati,**
- **la chiusura della connessione.**



LA SUITE DI PROTOCOLLI TCP/IP

TCP (Transmission Control Protocol)

Per fare un esempio, se ci connettiamo al server HTTP all'indirizzo 192.168.2.12 dal mio computer che ha indirizzo IP 10.22.115.88.

- il mio computer apre una porta libera (es 1025) sulla sua interfaccia di rete, ed invia un pacchetto all'altro computer. Questo pacchetto TCP ha un flag (nell'intestazione) chiamato SYN che indica "*richiesta di inizio connessione*". Contiene anche la porta TCP su cui si vuole fare la connessione (1025) e la porta TCP del server (80);
- il server HTTP è in ascolto (*listen*) sulla porta 80. Alla ricezione del SYN viene mandato di risposta un pacchetto speciale con impostati i flag SYN e ACK, che significano "*richiesta di connessione accettata, attendo conferma*";
- il mio computer riceve questo pacchetto, risponde con un pacchetto con impostato il solo flag ACK, che significa "*confermo la connessione*"; la connessione è stabilita e inizia il trasferimento dei dati. Questa sequenza ha il nome di *three way handshake*.
- i due computer hanno instaurato un canale affidabile di collegamento per cui il mio computer chiederà una pagina web del sito e il server la invierà (suddividendola se necessario in più pacchetti, in base alla capacità della rete MTU). Per ogni blocco di dati ricevuto il mio computer risponderà con un pacchetto con il solo flag ACK.
- Alla fine della navigazione, il mio computer invierà un pacchetto TCP con il flag FIN e il server risponderà con un pacchetto FIN/ACK, a cui il mio computer risponderà con un pacchetto con solo il flag ACK. Anche per la chiusura del collegamento si ripete la stessa comunicazione a tre fasi.
- Se il server ha problemi o perde la connessione con il mio computer (ad esempio se non riceve i pacchetti ACK di risposta entro un certo tempo) termina la connessione inviando un pacchetto TCP con il flag RST, è una sorta di segnalazione che bisogna iniziare una nuova connessione da capo.



LA SUITE DI PROTOCOLLI TCP/IP

TCP (Transmission Control Protocol)

Time	Event	DIAGRAM
t	Host A sends a TCP SYN chronize packet to Host B	<p>The diagram shows the sequence of events between Host A and Host B:</p> <ul style="list-style-type: none">At time t, Host A sends a syn packet to Host B, which arrives at $t+1$.At time $t+2$, Host B sends its own syn packet to Host A, which arrives at $t+3$.At time $t+4$, Host A sends an ack packet to Host B, which arrives at $t+5$.
$t+1$	Host B receives A's SYN	
$t+2$	Host B sends its own SYN chronize	
$t+3$	Host A receives B's SYN	
$t+4$	Host A sends ACK nowledge	
$t+5$	Host B receives ACK . <i>TCP connection is established.</i>	



LA SUITE DI PROTOCOLLI TCP/IP

MTU

La MTU (Maximum Transmission Unit) di una rete è la lunghezza massima del pacchetto che vi può transitare. Questa generalmente dipende dal protocollo del livello link.

I pacchetti con dimensioni maggiori della MTU vengono suddivisi in frammenti (fragments), ovvero in pacchetti più piccoli dai router incontrati durante il percorso ed il pacchetto viene poi ricomposto dall'interfaccia di rete di destinazione.

Protocollo	MTU
Token Ring 16 Mbit/s	17914
Token Ring 4 Mbit/s	4464
Ethernet	1500
PPP	< 1500



LA SUITE DI PROTOCOLLI TCP/IP

UDP (User Datagram Protocol)

E' usato per servizi e connessioni dove sia l'affidabilità che la gestione dello scambio dati sono demandati all'applicazione (user è in questo senso). Ad esempio la risoluzione dei nomi DNS è fatta con UDP.

Viene chiamato *connectionless* cioè senza collegamento; usa pacchetti di dimensione massima di 512 bytes, che partono da una porta UDP di un computer e sono destinati ad una porta UDP su un altro computer. Come le porte del protocollo TCP, in UDP le porte servono a distinguere i servizi e i richiedenti. I pacchetti vengono spediti e basta, è lasciato all'applicazione il compito verificare se arrivano o non arrivano.

Se sono così poco affidabili a che servono?

Esistono dei servizi che più che l'affidabilità della consegna dei pacchetti puntano alla velocità di risposta ed al basso traffico in rete. Uno di questi è la risoluzione dei nomi con il servizio DNS. Dato che il servizio dei nomi è gerarchico ed una richiesta di risolvere un nome potrebbe finire chissà dove, si preferisce usare UDP, in quanto i pacchetti sono leggeri e non richiedono traffico aggiuntivo. L'affidabilità è basata su tempi di attesa prestabiliti: se non si riceve risposta in un tempo predefinito, si riprova un numero determinato di volte e poi si passa un messaggio di errore all'applicazione che ha richiesto il servizio.



LA SUITE DI PROTOCOLLI TCP/IP

ICMP (Internet Control Message Protocol)

Serve alla diagnostica della connessione. E' usato per controllare che un computer in rete sia raggiungibile, oppure per notificare errori di instradamento (routing) sulla rete.

E' il protocollo che lavora al livello più basso, proprio per la sua caratteristica di protocollo diagnostico. I pacchetti ICMP vengono ricevuti e gestiti direttamente dal gestore del protocollo IP, senza coinvolgere applicazioni a livello superiore. Al più, se l'uso di ICMP è richiesto dall'utente, ad esempio con il comando ping, vengono stampate a video le risposte avute e altre informazioni diagnostiche e qualitative.



LA SUITE DI PROTOCOLLI TCP/IP

ICMP (Internet Control Message Protocol)

- Per capire meglio di cosa parliamo, aprite una console sul vostro PC e digitate questo comando:

```
ping 66.102.9.104
```

- ed avrete come risposta:

```
PING 66.102.9.104 (66.102.9.104) from 127.0.0.1 : 56(84) bytes of data.
```

```
64 bytes from 127.0.0.1: icmp_seq=1 ttl=255 time=0.352 ms
```

```
64 bytes from 127.0.0.1: icmp_seq=2 ttl=255 time=0.108 ms
```

```
64 bytes from 127.0.0.1: icmp_seq=3 ttl=255 time=0.114 ms
```

In breve, si invia un pacchetto ICMP di tipo *Echo Request* (ping) ad un computer e questo risponde con uno di tipo *Echo Reply* (pong). Il tempo che intercorre fra l'invio della richiesta e la ricezione della risposta viene chiamato *round trip delay*, grossolanamente tradotto tempo di andata-ritorno.

Come vedete con un semplice pacchetto si ottengono delle informazioni, ossia se il computer è acceso e collegato in rete, il tempo di transito della connessione, indicazione della velocità di trasmissione dei dati, e sulla qualità della connessione, data dalla regolarità nel tempo di round trip.



LA PILA PROTOCOLLARE

- L'accesso alle funzionalità della comunicazione avviene a vari **livelli di astrazione** a partire da quello più basso che è quello dei **segnali** coinvolti nell'effettiva comunicazione (la cui natura e tipologia dipendono dalla natura del mezzo di trasmissione) fino a quello più elevato che è quello che “vedono” le **applicazioni** che vogliono comunicare attraverso la rete.
- In questo modo un'applicazione che intende inviare delle informazioni sulla rete, utilizzerà l'interfaccia software messa a disposizione dal livello più alto, che si preoccuperà di trattare opportunamente le informazioni passandole al livello immediatamente inferiore e così via fino ad arrivare al livello più basso in cui i segnali logici saranno trasformati in segnali elettrici o elettromagnetici e quindi inviati sulla rete.
- Allo stesso modo, l'interfaccia di rete che riceve le informazioni le tratterà in maniera opportuna passandole man mano ad un livello sempre più elevato, fino ad arrivare all'applicazione di destinazione.
- **Ogni livello passa le informazioni a quello immediatamente inferiore (in trasmissione) o superiore (in ricezione) tramite un insieme di funzioni che costituiscono l'interfaccia di comunicazione tra un livello e l'altro.**



LA PILA PROTOCOLLARE

- Per riassumere, un pacchetto che entra nel nostro computer fa questo percorso, dal basso verso l'alto:
 - Applicazione (UDP/TCP)
 - Protocolli superiori (ICMP/UDP/TCP)
 - Protocollo IP
 - Driver (ethernet/ppp)
 - Interfaccia fisica
- ed ovviamente il percorso contrario nel caso di pacchetto che esce.



LA PILA PROTOCOLLARE

Lo stack OSI:

Il modello che è stato la pietra miliare nella definizione degli stack di protocolli di rete, ma di cui non esiste nessuna implementazione pratica, è il modello OSI (Open System Interconnection) proposto da ISO2, che si suddivide nei 7 livelli di seguito elencati.

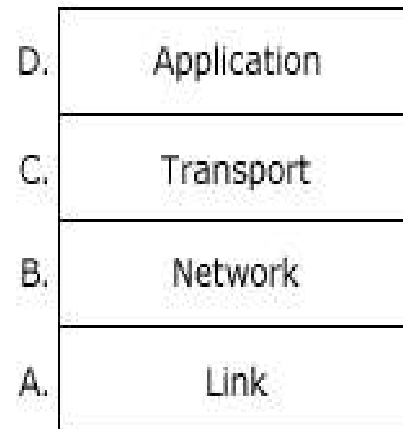
7.	Application
6.	Presentation
5.	Session
4.	Transport
3.	Network
2.	Data link
1.	Physical



LA PILA PROTOCOLLARE

Lo stack TCP/IP

Per le reti ci sono stati vari standard di protocolli di comunicazione che ovviamente hanno portato a problemi di interconnessione tra reti diverse. Quello che oggi prevale (e sarà destinato a farlo sempre di più) è il TCP/IP (che sarebbe meglio chiamare Internet suite), una suite di protocolli nata con Internet, per le reti geografiche (WAN), poco affidabili rispetto alle LAN, ma che ha buone prestazioni anche su LAN (sebbene su LAN esistano dei protocolli più efficienti in termini di rapporto tra i codici di controllo e le effettive informazioni da trasmettere).



- A. Link corrisponde ai livelli 1 (Physical) e 2 (Data link) dello stack OSI.
- B. Network si mappa esattamente sul livello 3 (Network) dello stack OSI.
- C. Transport ingloba i livelli 4 (Transport) e 5 (Session) dello stack OSI.
- D. Application ingloba i livelli 6 (Presentation) e 7 (Application) del modello OSI



BREVE ILLUSTRAZIONE DELLA RETE LOCALE DEL LINUXDAY GENZANO:

La LAN è strutturata in 3 blocchi,

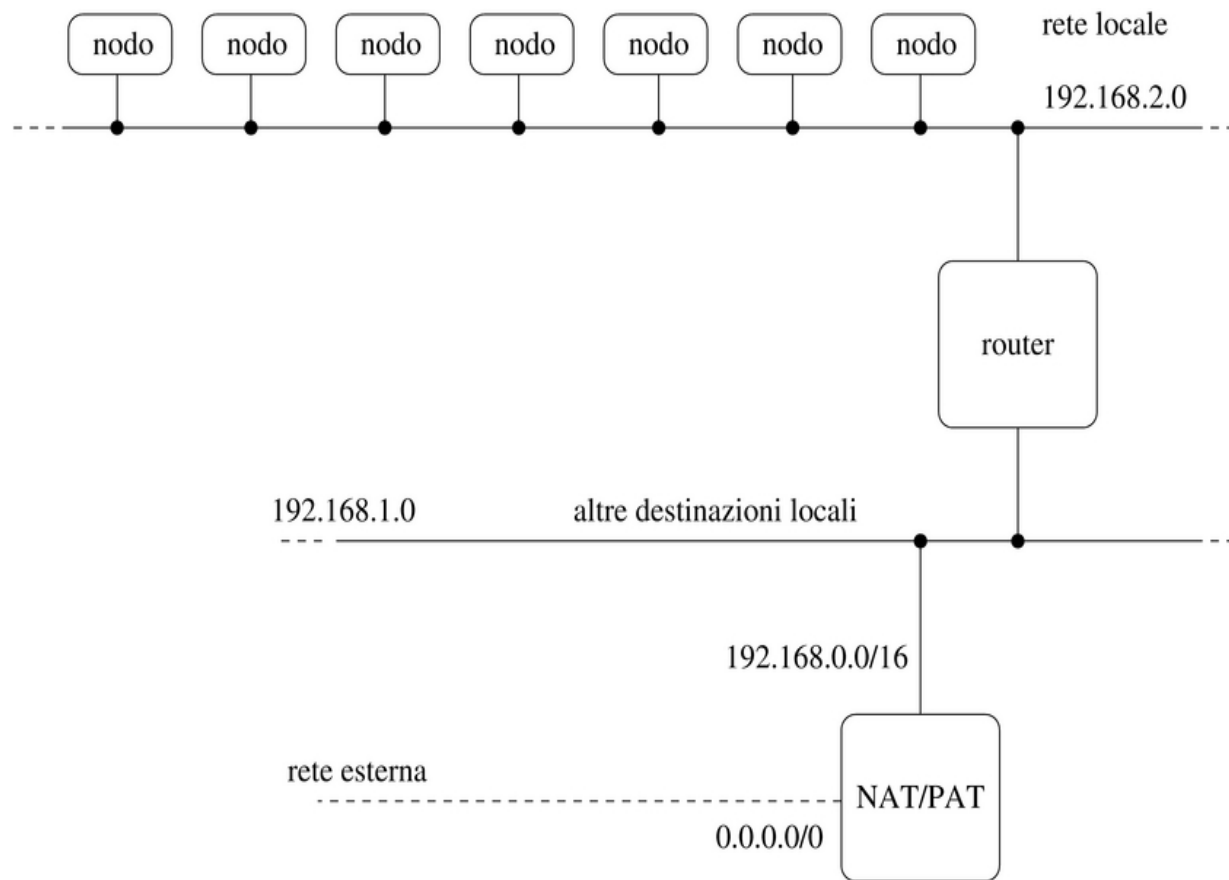
- il primo comprende il server HAL, il Proxy-Firewall e i client NFS e Diskless
- il secondo comprende il server Web e l'area destinata all'installazione
- l'ultimo è direttamente derivato dal primo, e comprende la subnet WiFi, formata dall' AP e dai portatili che utilizzano connessioni wireless.

Il server "Digital" fornisce connettività internet all'intera LAN, assicurando tramite server **Proxy (Squid+SquidGuard)** il controllo sui contenuti delle pagine web e dei download inoltre un **firewall (iptables)** protegge la rete dall'esterno, filtra i contenuti in uscita, e tramite **NAT**, permette a tutte le macchine di utilizzare un solo link, condividendo la banda (addirittura limitandola per alcune macchine del terzo blocco) e nascondendo la rete dietro ad **un solo indirizzo pubblico**.



BREVE ILLUSTRAZIONE DELLA RETE LOCALE DEL LINUXDAY GENZANO:

NAT



COSA SI INTENDE PER SICUREZZA NELL'AMBITO DI UNA RETE DI CALCOLATORI?

- Nel momento in cui si piazza in rete un proprio elaboratore, rendendolo accessibile al pubblico, si assumono delle responsabilità.
- In particolare, altri sistemi potrebbero risultare danneggiati da un attacco condotto con successo ai danni del proprio. Quindi, la cosa non può essere ignorata, anche quando per se stessi potrebbe non essere importante.

A questo livello si pone il concetto di sicurezza informatica

- Sebbene un sistema possa essere amministrato con tutti i criteri di sicurezza possibili, non bisogna escludere che ci sia comunque la possibilità che un attacco riesca a passare le misure di sicurezza presenti su un sistema.
- Pertanto occorrerà prevedere dei sistemi per limitare i danni di eventuali intrusioni segnalandole tempestivamente. Questo è il campo dei sistemi di rilevamento delle intrusioni (IDS - Intrusion Detection System).



COSA SI INTENDE PER SICUREZZA NELL'AMBITO DI UNA RETE DI CALCOLATORI?

Si intende quella legittima aspettativa che ha l'utente autorizzato all'uso di un computer di usufruire dei servizi dei file, dei documenti che il computer mette a disposizione. Nel momento in cui hanno accesso a documenti e servizi persone non autorizzate si ha quello che si chiama "buco di sicurezza" (della rete o del calcolatore).

Ma quando un dato è effettivamente trattato in modo "sicuro" ? Quali caratteristiche deve avere?

- **Riservatezza** le informazioni sono fruibili soltanto dal destinatario e non da altri;
- **Integrità** il destinatario deve essere in grado di verificare se le informazioni che gli sono arrivate hanno subito delle modifiche rispetto a quelle inviate dal mittente;
- **Autenticazione** il destinatario deve essere in grado di verificare se le informazioni ricevute sono state effettivamente inviate da chi afferma di essere il mittente;
- **Non ripudiabilità** il mittente che ha inviato le informazioni non può disconoscere di aver inviato le informazioni stesse;
- Per estensione, un **servizio** è sicuro quando fruibile dagli aventi diritto;
- Un **server** è sicuro quando i dati che trasmette ai client sono integri.



COSA SI INTENDE PER SICUREZZA NELL'AMBITO DI UNA RETE DI CALCOLATORI?

IN CHE MODO UN HOST PUÒ ESSERE VIOLATO?

I modi sono tanti e diversissimi fra loro, come pure diversi sono i tipi di intrusione, dal punto di vista dello scopo. Ci sono le intrusioni per danneggiare, quelle per rubare dati, e quelle per usare il nostro computer a fini poco etici a nostra totale insaputa.

Iniziamo a dividerle per modalità:

- **Mancanza di protezione adeguata**
- **Errori nel software**
- **Uso improprio di protocolli e servizi**



IN CHE MODO UN HOST PUÒ ESSERE VIOLATO?

Computer poco protetti

- Computer in cui gli *accessi leciti* sono poco o per nulla protetti.
- Mancanza di politiche di protezione adeguate.
- Server mal configurati, password semplici e prevedibili, o addirittura assenti.



IN CHE MODO UN HOST PUÒ ESSERE VIOLATO?

Errori nel software

E' il sistema più subdolo e pericoloso di intrusione. A differenza del punto precedente, in questo caso l'intrusione avviene anche se le nostre password e configurazioni sono blindate.

Prendiamo un esempio reale: il **virus Nimda**. Questo virus usa vari metodi per propagarsi e per infettare altri computer, ma in sostanza sono tutti basati su due bug in software ben noti: un web browser ed un web server commerciali. Per il browser sfrutta un errore nella intestazione dei messaggi con allegato, per cui (detto in modo impreciso, ma essenziale) lo inganna facendogli credere che un programma allegato ad un messaggio è un contenuto multimediale e lo apre in anteprima, mandandolo in esecuzione *senza che l'utente abbia modo di impedirlo*.

Una volta infettato un computer, si autospedisce come allegato agli indirizzi di posta elettronica che trova nella rubrica e nella cache delle pagine web, estraendoli dai link *mailto:* contenuti nelle pagine stesse. Se avete una connessione Internet, cerca dei siti web dove si installa, sfruttando un errore nel programma server nella decodifica degli URL, modificando pagine web e sostituendole con false form da cui si propaga ad altri computer quando utenti ignari visitano quelle pagine.



IN CHE MODO UN HOST PUÒ ESSERE VIOLATO?

Errori nel software

Altro tipo di danno che può essere causato è l'interruzione del servizio, o Denial of Service (**DoS**). Se un server web di un certo tipo contiene un errore che lo fa andare in crash quando riceve una specifica richiesta, qualcuno potrebbe sfruttare questo errore per mandare fuori servizio i server della concorrenza. Se poi i server vengono riavviati, basta rimandare la richiesta illegale per rimandarli *down* (in gergo si dice così quando un computer o un servizio di rete viene fermato o spento), rendendoli completamente inutilizzabili. Immaginate l'entità del danno se questo succede ad un sito di Internet Banking o di e-commerce...

Altri tipi di errori comprendono ad esempio il famigerato **buffer overflow** di cui avrete sentito parlare. Di solito questo tipo di errore è sfruttato per far eseguire al computer vittima codice binario contenuto nel pacchetto stesso, che **può essere un pacchetto perfettamente legale per il protocollo, ma imprevisto dal software**, ad esempio una query DNS con un nome di server più lungo di quello che dovrebbe, che non provoca errori né a livello di protocollo IP, né a livello superiore, ma potrebbe innescare un comportamento indesiderato.

Il trucco dei pacchetti con contenuto di lunghezza errata veniva usato anni or sono in server tipo *sendmail* (gestore di posta elettronica) o in *bind* (server DNS).



IN CHE MODO UN HOST PUÒ ESSERE VIOLATO?

Errori nel software

E' di questi giorni la segnalazione di un problema simile nel codice che gestisce le **immagini JPEG** in Windows. Il codice ha un errore che puo' essere per far eseguire dati dell'immagine errata come se fossero un programma. Un malintenzionato può creare una immagine particolare che se visualizzata da un browser causa l'errore e manda in esecuzione il codice nascosto dentro l'immagine. Dato che questo codice viene eseguito ad un livello privilegiato può virtualmente fare di tutto. In questo caso addirittura non c'è neanche bisogno di avere servizi di rete attivi, basta navigare su Internet e andare in un sito contenente le immagini deleterie.

Altro esempio di attacco realizzato con un pacchetto perfettamente legale per il protocollo, ma deleterio per il server, è stato usato dal worm SQHell (conosciuto anche come **SQL Slammer** o Sapphire). Questo virus si propaga tramite UDP verso la **porta 1434** dei server, e la *semplice ricezione* del pacchetto comporta l'infezione.

*Ora, si dovrebbero bloccare con una regola del firewall tutti i **pacchetti UDP provenienti da internet** che vanno a quella porta ma è da notare che questo attacco, meno di un anno or sono ha paralizzato per giorni molti siti istituzionali e di aziende di pubblici servizi...*



IN CHE MODO UN HOST PUÒ ESSERE VIOLATO?

Errori nel software

Esiste una soluzione in questi casi: ***correggere il software errato.***

Come? Applicando quella che in gergo è chiamata ***patch*** (pezza).

Se il software è commerciale, si è sottoposti alla prontezza del produttore, che non sempre è ricettivo su questo fronte, anche se in questi ultimi tempi qualcosa è cambiato, anche grazie ad una maggiore coscienza degli utenti.



IN CHE MODO UN HOST PUÒ ESSERE VIOLATO?

Uso improprio di protocolli e servizi

In questo caso, purtroppo, non c'è una negligenza di qualcuno o un errore nel software. Semplicemente, qualche malintenzionato sfrutta meccanismi propri di un protocollo o uno specifico servizio per creare problemi o per generare confusione.

In dettaglio, un esempio di "attacco" utilizzato in passato, dove il malintenzionato opera in questo modo:

- manda richieste ICMP Echo Request con un indirizzo broadcast ad una rete di computer, spacciandosi per il computer che vuole attaccare. Un indirizzo broadcast è fatto in modo da essere interpretato come "tutti i computer di questa rete", ossia se vogliamo mandare il pacchetto a tutti i computer della rete 192.168.1.x lo mandiamo all'indirizzo 192.168.1.255
- i computer di questa rete rispondono tutti insieme alla richiesta con un pacchetto ICMP Echo Reply indirizzandolo al computer vittima; il computer vittima riceve una valanga di pacchetti che non ha richiesto e deve gestire in qualche modo; il risultato è che o il computer vittima dell'attacco soccombe per sovraccarico e perde la connessione alla rete, o semplicemente non risulta più raggiungibile da altri computer per saturazione della banda disponibile. Questo tipo di attacco, effettuato con modalità *ICMP flood* (inondazione ICMP) è stato denominato *smurf* (puffo). Ha due caratteristiche che lo rendono appetibile: è difficile risalire al colpevole e necessita di pochissimo traffico generato dall'attaccante per fare danni.
- **Contromisura, per questo esempio:** nessun computer di nessuna rete deve rispondere a messaggi ICMP Echo Request con indirizzo di destinazione broadcast.



TIPI DI INTRUSIONE

Sfruttando le modalità sopra elencate, vediamo come si possano creare dei software malevoli e come possano causare i danni più disparati.

I tipi di intrusione che seguono sono elencati in modo molto rapido, con il grado di pericolosità ed il metodo di propagazione, e se un firewall configurato correttamente, non con la configurazione predefinita all'installazione, può ridurre o eliminare la possibilità di esserne vittime.



TIPI DI INTRUSIONE

Virus

La parola virus è diventata di uso generico, indicando un software capace di autoreplicarsi, diffondersi e provocare danni all'insaputa del legittimo proprietario/utente, sfruttando le risorse disponibili all'interno del computer ospite, esattamente come la loro controparte biologica da cui prendono il nome.

Altro non è che un programma che si attiva con l'esecuzione da parte dell'utente o di un altro programma infetto o del programma che è il virus stesso. Si propaga in vari modi, e quando la propagazione avviene attraverso la rete si parla di **worm**. Possono contenere virus: **allegati di posta** mascherati da testo o screensaver, **file compressi, dischetti, file scaricati da reti Peer to Peer**. Possono sfruttare buchi nella sicurezza di un programma di posta elettronica per attivarsi alla semplice lettura del messaggio, anche senza aprire l'allegato, sfruttando la funzione di anteprima. Il grado di dannosità varia dalla sola propagazione, quindi consumo di risorse, al danneggiamento irreparabile di file, con conseguente blocco del funzionamento del sistema operativo vittima.

Bliss è stato il primo virus realizzato specificatamente per i sistemi GNU/Linux, che comunque potrebbe essere ricompilato facilmente per la maggior parte dei sistemi Unix. Le informazioni sul suo funzionamento sono state ottenute da un'analisi condotta da Ray Lehtiniemi, come documentato in Bliss, a Linux "virus" di Axel Boldt.

Bliss si attacca ai file eseguibili nella loro parte iniziale, aggiungendo in coda una stringa di riconoscimento. Quando si avvia un programma infettato in questo modo, in realtà si mette in funzione il virus, che fa le sue cose e poi estrae il file originale salvandolo temporaneamente in tmp/.bliss-tmp.pid, da dove poi provvede a metterlo in funzione.



TIPI DI INTRUSIONE

Trojan

Devono il loro nome al Cavallo di Troia, per il loro modo di attivazione. Si presentano come programmi utili per qualche cosa, che all'interno contengono invece codice deleterio. A volte sono versioni di programmi noti che in realtà sono stati modificati per diventare veicolo di infezione. Di solito non hanno un meccanismo proprio di propagazione, sfruttando appunto l'inganno dell'apparenza "utile".

Worm

Si propagano senza intervento diretto dell'utente attraverso la rete per cui è sufficiente essere connessi, anche senza navigare o leggere posta elettronica. Sfruttano errori nel software di servizi di rete per installarsi nei computer e da lì cercano altre vittime per infettarle. I danni possono andare dal consumo di risorse, alla distruzione di dati, al blocco del servizio.

Qui il firewall può molto, ma dipende dal servizio di rete che viene attaccato. Il worm Blaster sfrutta un errore nel servizio RPC di Windows NT/2000/Xp in ascolto sulla porta 135/TCP. Se si chiude questa porta a qualsiasi accesso, si impedisce a Blaster di infettare il PC. Non sempre però le cose vanno così bene.

La famiglia di worm descritta in Gaobot usa fra l'altro errori nel servizio di condivisione disco e password deboli sulle condivisioni stesse. Le porte di questi servizi sono la 139/TCP e 445/TCP ed entrambe servono per accedere ai **file ed alle stampanti condivise su una rete**. Quindi di solito di libero accesso per PC nella stessa sottorete.



TIPI DI INTRUSIONE

Backdoor

Letteralmente "porta sul retro". Era una pratica usata da molti sviluppatori quando temevano di non essere pagati per il loro lavoro, o quando si voleva avere accesso ai dati gestiti dall'applicazione aggirando le protezioni e l'accesso previsto per i normali utenti, prevenendo il caso non infrequente che l'utente rimanesse "chiuso fuori" da qualche manovra errata.

Ne è stato esempio la backdoor trovata in un database commerciale pochi giorni dopo il rilascio del sorgente in open source, di cui qui trovate la segnalazione: <http://www.kb.cert.org/vuls/id/247371>. Per anni gli utilizzatori del database sono stati ignari del fatto che chiunque poteva entrare a piacimento nel database come amministratore. E dato che il database era accessibile tramite rete alla porta 3050/TCP, provate a pensare ad un sito di e_commerce con i numeri di carta di credito memorizzati sullo stesso server che ospita il web server, quindi aperto ad Internet...

Da qualche tempo a questa parte, **molti virus e worm installano delle backdoor sul computer vittima creando dei punti di accesso da cui è possibile prendere il controllo completo del computer dall'esterno ad insaputa della vittima**. Oppure collegano il computer ad un server IRC da cui il virus può prendere ordini. E' una tecnica usata negli ultimi virus in circolazione per poi **coordinare attacchi in massa a siti pubblici**.



TIPI DI INTRUSIONE

Interruzioni di servizio (Denial of Service o DoS)

E' un tipo di intrusione realizzato senza toccare direttamente il computer vittima, sfruttando modi di funzionamento dei servizi di rete o falle nei server di rete. Un esempio è il TCP SYN Flood DoS. Abbiamo parlato del funzionamento della connessione TCP, per cui capirete sicuramente.

Immaginate di mandare ad un server tanti pacchetti TCP SYN come per iniziare la connessione alla stessa porta, ad esempio quella del servizio HTTP, la 80. Il server per ogni pacchetto che riceve risponde con un SYN/ACK e memorizza in una tabella l'ora di trasmissione della risposta, i dati della connessione richiesta ed aspetta il pacchetto ACK dall'altra parte. *Ovviamente per non sprecare risorse, questa tabella ha uno spazio limitato, ed ogni risposta in attesa che viene memorizzata ha un timeout: se dopo un certo numero di secondi non arriva l'ACK, quel dato viene scartato dalla tabella.*

Ma se si mandano migliaia di pacchetti TCP SYN al secondo può succedere di saturare la tabella del server prima che intervenga il timeout. Il risultato può essere o il blocco del server se il programmatore non ha previsto questa eventualità, cosa non infrequente, o l'impossibilità per altri utenti di accedere al server, da cui il nome *Denial of Service* (rifiuto del servizio).

Se poi l'attacco viene portato usando molti computer contemporaneamente, magari infettati con un **virus che installa una backdoor**, viene chiamato DDoS **Distributed Denial of Service** per indicare appunto la natura multipla dell'attacco. I virus della famiglia **MyDoom** sono progettati per questo tipo di attacco.



TIPI DI INTRUSIONE

RootKit

Bestia nera degli amministratori Unix/Linux. Letteralmente significa **attrezzatura per diventare root** e si spiega da solo. Sfruttando anche qui vulnerabilità note e buchi nella sicurezza di programmi server diffusi, si riesce a guadagnare un accesso al computer attaccato con privilegi di amministratore, root appunto.

Poi con varie tecniche si nasconde questo accesso abusivo con trucchi tipo installare versioni modificate, in gergo *trojaned*, delle utility di sistema per vedere le connessioni di rete, i processi che girano e gli utenti connessi, in modo che l'amministratore legittimo rimanga ignaro delle modifiche.

Si chiamano RootKit perché spesso esistono sotto forma di **programmi preconfezionati** che permettono di ottenere il risultato senza sapere nulla delle tecniche di intrusione e occultamento, semplicemente eseguendoli sul proprio computer e indicando l'indirizzo di rete del computer da attaccare.

Spesso l'attaccante non è un esperto di intrusioni e non ha nessuna idea sul come e perché funziona l'intrusione, ha solo trovato il programma pronto per l'uso. In questo caso ci si riferisce a questi personaggi, aspiranti cracker, come **Script Kiddies**.



TIPI DI INTRUSIONE E

Spyware

Questa è una categoria piuttosto recente di intrusioni, ed è relativamente pericolosa non in modo diretto, quanto per problemi di privacy. In sostanza questo tipo di intrusioni sono realizzate attraverso software ritenuti utili, analogamente al caso dei trojan, che invece di far danni alla maniera dei virus, si limitano a spiare alcune attività dell'utente durante la navigazione su Internet, comportamento da cui deriva il nome: *spy software*. Vengono raccolte informazioni sui siti che visitate, le parole chiave che inserite nei motori di ricerca, i dati che inserite nei moduli che riempite sul web e così via. I dati raccolti da questi programmi vengono silenziosamente inviati ad un server remoto del produttore del software che rivende questi dati statistici a chi interessano. Tutto questo spesso senza informarvene e ovviamente senza il vostro consenso. Alcuni tipi di spyware collezionano l'elenco del software installato sul computer o dei file multimediali, o ancora gli indirizzi di posta che avete nella vostra rubrica. Capite bene che è un vero e proprio spionaggio, e nel caso di dati sensibili costituisce un reato, oltre al fatto che si viene tenuti all'oscuro di questo comportamento.

Keylogger

E' l'antenato dello spyware, e di solito va in combinazione con virus, worm, rootkit o trojan. Una volta installato nel vostro computer, memorizza tutte le sequenze di tasti che digitate, e le invia ad un server remoto, dove chi ha creato il keylogger analizza le sequenze ed estrae cosette come password di accesso e codici di carte di credito. Può usare una backdoor o una mail per spedire i dati collezionati.



STUDIARE UNA POLITICA DI DIFESA

SISTEMI DI PROTEZIONE ED ANALISI

Vediamo rapidamente come:

- analizzare la nostra rete e le nostre macchine;
- utilizzare al meglio gli strumenti esistenti e garantire una protezione accettabile e sufficienti garanzie nell'integrità dei nostri dati;
- proteggere dall'esterno le nostre macchine.



STUDIARE UNA POLITICA DI DIFESA

SISTEMI DI PROTEZIONE ED ANALISI

Strumenti per il controllo e l'analisi del traffico IP

L'analisi del traffico della rete, sia per mezzo dell'intercettazione di tutti i pacchetti che attraversano una rete fisica, sia per mezzo del controllo di ciò che riguarda esclusivamente una singola interfaccia di rete del nodo locale, è molto importante per comprendere i problemi legati alla sicurezza e per scoprire inconvenienti di vario genere.



SISTEMI DI PROTEZIONE ED ANALISI

Netstat

Netstat è un programma specifico di GNU/Linux, in grado di mostrare in modo agevole alcune informazioni contenute nella directory `/proc/net/`. Le informazioni disponibili riguardano esclusivamente la sfera del nodo locale, comprese le connessioni che lo riguardano.

```
# netstat --inet
```

Emette l'elenco dei socket di dominio Internet, ovvero tutte le comunicazioni aperte tra i programmi attraverso i protocolli TCP/IP.

```
# netstat --inet -e
```

Come nell'esempio precedente, aggiungendo l'indicazione degli utenti proprietari dei processi che attuano le connessioni.

```
# netstat --tcp -a
```

Mostra la situazione delle porte TCP, in particolare quelle dei servizi in ascolto.



SISTEMI DI PROTEZIONE ED ANALISI

Fuser

Fuser è un programma specifico per sistemi GNU/Linux, che consente di individuare facilmente il processo elaborativo che ha aperto un file, oppure una porta (TCP o UDP). L'esempio seguente mostra il comando necessario a conoscere il numero identificativo del processo che ha aperto la porta TCP 22:

```
# fuser -n tcp 22
```

```
22/tcp:          598
```

Successivamente, conoscendo il numero UID del processo, con l'aiuto di ps, si può scoprire chi è:

```
# ps ax | grep " 589 "
```

```
598 ?          s      0:00 /usr/sbin/sshd
```



SISTEMI DI PROTEZIONE ED ANALISI

Tcpdump

Tcpdump è lo strumento fondamentale per l'analisi del traffico che avviene nella rete fisica a cui si è collegati. Permette sia di ottenere una visione sintetica dei pacchetti, sia di visualizzarne il contenuto in esadecimale. Inoltre, è possibile definire un filtro ai pacchetti da prendere in considerazione. Purtroppo, il suo utilizzo efficace richiede un'ottima conoscenza dei protocolli TCP/IP.

I pacchetti vengono analizzati solo nella prima parte, normalmente di 68 byte, perdendo le informazioni successive. Eventualmente, questa dimensione può essere aumentata, anche se in generale ciò è sconsigliabile dal momento che richiederebbe un tempo di elaborazione maggiore, portando anche alla perdita di pacchetti.

Tcpdump può generare un risultato in esadecimale, oppure può emettere i pacchetti così come sono. Per poter interpretare il contenuto dei pacchetti, è necessario conoscere la loro struttura, in base ai protocolli relativi.



SISTEMI DI PROTEZIONE ED ANALISI

Ethereal

Ethereal è un programma per **l'analisi del traffico di rete, fino al livello due del modello ISO-OSI (collegamento dati), riuscendo a riconoscere all'interno di questo una serie di protocolli al livello tre e quattro del modello ISO-OSI (rete)**. In particolare, individua correttamente molti protocolli collegati a IPv4 e IPv6.

Ethereal è pensato principalmente per accumulare il traffico intercettato, allo scopo di consentire un'analisi dettagliata di questo in un momento successivo; nello stesso modo è predisposto per accedere a informazioni di questo genere accumulate da programmi diversi, così come è in grado di esportare i propri dati in formati alternativi.

Ethereal consente anche una visualizzazione in tempo reale del traffico in corso, in modo analogo a quanto fa **IPTraf**, con la differenza che le informazioni fornite sono molto più chiare. In questo senso, Ethereal è un ottimo strumento didattico per lo studio delle reti.

Ethereal viene usato normalmente attraverso il sistema grafico X e deve funzionare con i privilegi dell'utente root, per poter accedere direttamente all'interfaccia di rete da sondare.



SISTEMI DI PROTEZIONE ED ANALISI

File Edit Capture Display Tools Help

No. .	Time	Source	Destination	Protocol	Info
51	115.254771	roggen.brot.dg	dinkel.brot.dg	TCP	1025 > telnet [ACK] Seq=3854203109 Ack=3614
52	115.273440	dinkel.brot.dg	roggen.brot.dg	TELNET	Telnet Data ...
53	115.274514	roggen.brot.dg	dinkel.brot.dg	TCP	1025 > telnet [ACK] Seq=3854203109 Ack=3614
54	115.274699	dinkel.brot.dg	roggen.brot.dg	TELNET	Telnet Data ...
55	115.275234	roggen.brot.dg	dinkel.brot.dg	TCP	1025 > telnet [ACK] Seq=3854203109 Ack=3614
56	115.581118	dinkel.brot.dg	roggen.brot.dg	TELNET	Telnet Data ...
57	115.581917	roggen.brot.dg	dinkel.brot.dg	TCP	1025 > telnet [ACK] Seq=3854203109 Ack=3614
58	115.654587	dinkel.brot.dg	roggen.brot.dg	TELNET	Telnet Data ...
59	115.655294	roggen.brot.dg	dinkel.brot.dg	TCP	1025 > telnet [ACK] Seq=3854203109 Ack=3614
60	124.870001	fe80::250:baff:fe71:d	ff02::1	ICMPv6	Router advertisement
61	153.339857	fe80::250:baff:fe71:d	ff02::1	ICMPv6	Router advertisement

Frame 52 (428 on wire, 428 captured)

- Ethernet II
- Internet Protocol, Src Addr: dinkel.brot.dg (192.168.1.1), Dst Addr: roggen.brot.dg (192.168.1.2)
- Transmission Control Protocol, Src Port: telnet (23), Dst Port: 1025 (1025), Seq: 3614437742, Ack: 3854203109, Len
- Telnet

```

0000  00 80 ad c8 a9 81 00 50 ba 71 d9 c1 08 00 45 10  .....P .q....E.
0010  01 9e f0 e1 40 00 40 06 c5 14 c0 a8 01 01 c0 a8  ...@.@.....
0020  01 02 00 17 04 01 d7 6f f1 6e e5 ba 78 e5 80 18  .....o .n..x...
0030  16 a0 02 a4 00 00 01 01 08 0a 00 0c 01 94 00 00  .....
0040  9a 5f 4c 69 6e 75 78 20 64 69 6e 6b 65 6c 20 32  .._Linux dinkel 2
0050  2e 34 2e 31 39 20 23 31 20 46 72 69 20 4e 6f 76  .4.19 #1 Fri Nov
0060  20 38 20 31 38 3a 33 31 3a 33 34 20 43 45 54 20  8 18:31 :34 CET
0070  32 30 30 32 20 69 36 38 36 20 75 6e 6b 6e 6f 77  2002 i68 6 unknow
0080  6e 20 75 6e 6b 6e 6f 77 6e 20 47 4e 55 2f 4c 69  n unknow n GNU/Li
0090  6e 75 78 0d 0a 0d 0a 4d 6f 73 74 20 6f 66 20 74  nux...M ost of t
00a0  68 65 20 70 72 6f 67 72 61 6d 73 20 69 6e 63 6c  he progr ams incl
00b0  75 64 65 64 20 77 69 74 68 20 74 68 65 20 44 65  uded wit h the De
00c0  62 69 61 6e 20 47 4e 55 2f 4c 69 6e 75 78 20 73  bian GNU /Linux s
  
```

Filter: / Reset Apply Internet Protocol (ip)

SISTEMI SICURI DI TRASMISSIONE

la crittografia

dal greco $\kappa\rho\upsilon\pi\tau\omicron\varsigma$ (kryptòs = nascosto) e $\gamma\rho\alpha\phi\epsilon\iota\nu$ (gràfein = scrivere) è la scienza che studia i **metodi per “cammuffare” le informazioni** in maniera tale che chi eventualmente le legge non sia in grado di risalire all'informazione originale se non è a conoscenza della **chiave di decodifica** del messaggio cifrato.

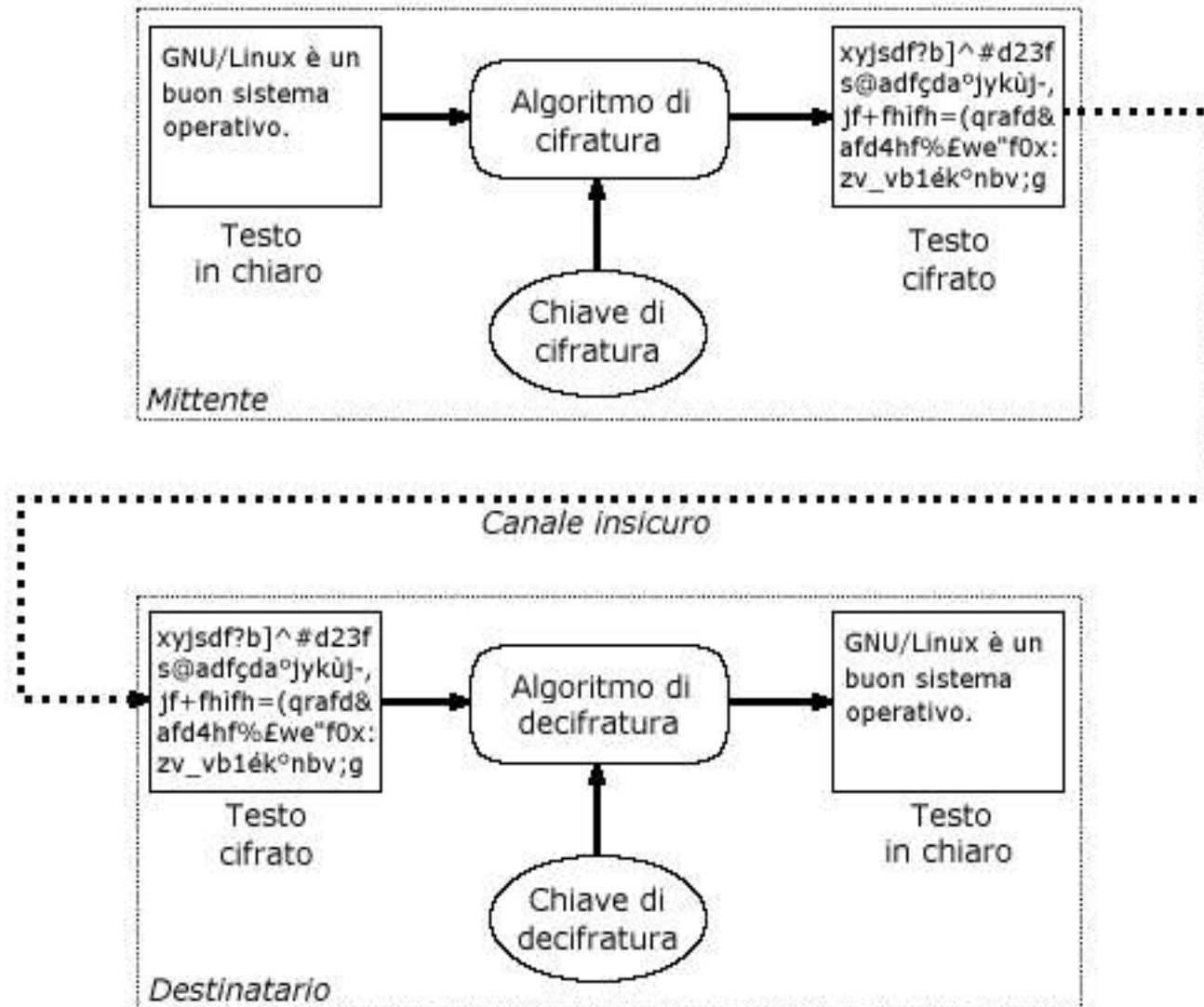
La chiave è appunto il valore di un parametro che fornito al meccanismo di cifratura considerato, è in grado di permettere la decodifica corretta dell'informazione.

Questi metodi garantiscono la **protezione delle informazioni trasmesse da un mittente ad un destinatario** attraverso un canale insicuro, al quale chiunque può accedere, in particolare forniscono un certo livello di riservatezza (o privacy) alle informazioni trasmesse.

La cifratura a **chiave asimmetrica** utilizza due chiavi diverse: una per la cifratura dell'informazione e l'altra per la decifratura. Le informazioni cifrate con una delle due chiavi possono essere decifrate solo con l'altra.



SISTEMI SICURI DI TRASMISSIONE



SISTEMI SICURI DI TRASMISSIONE

La firma digitale

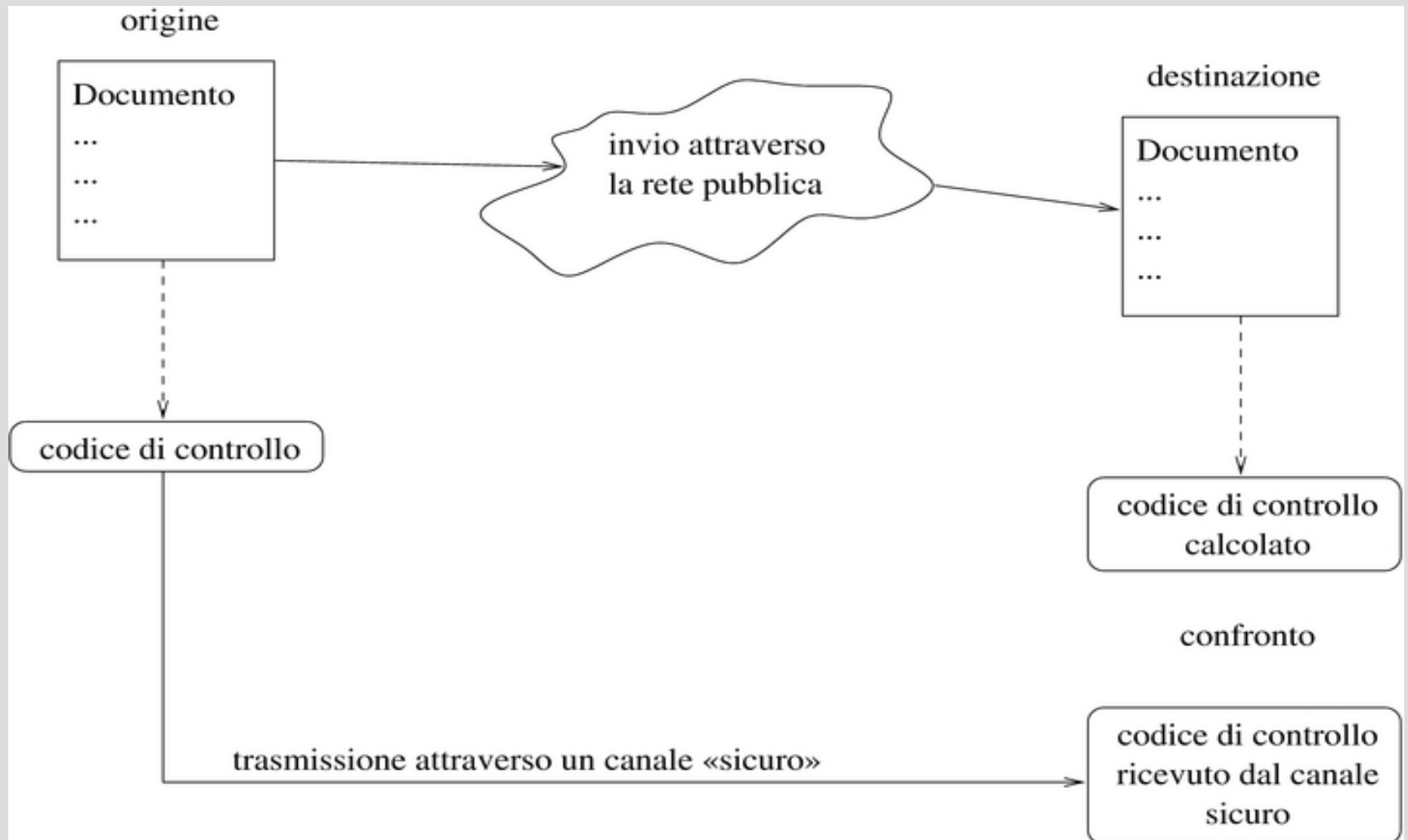
La firma digitale è una garanzia del messaggio o documento elettronico, alla stregua della sottoscrizione di un documento cartaceo, **attestandone con certezza l'integrità, l'autenticità e la non ripudiabilità** anche dal punto di vista legale, poiché la legislazione italiana attribuisce ad un documento elettronico con firma digitale lo stesso valore dello stesso in forma cartacea sottoscritto con firma autografa (v. art. 15 comma 2 della legge n. 59 del 15/3/1997 "Bassanini-1", D.P.R. n. 445 del 28/12/2000, D.P.C.M. 08/02/1999)

La firma elettronica ha lo scopo di certificare l'autenticità dei dati. Per ottenere questo risultato occorre garantire che l'origine di questi sia autentica e che i dati non siano stati alterati.

Per dimostrare che un documento elettronico non è stato alterato, si utilizza la tecnica del codice di controllo, che in pratica è un numero (o una stringa), che si determina in qualche modo in base al contenuto del documento stesso. L'algoritmo che genera questo codice di controllo è tanto più buono quanto è minore la probabilità che due documenti diversi generino lo stesso codice di controllo. Questo valore è una sorta di «riassunto» matematico del documento elettronico originale, che può essere fornito a parte, attraverso un canale ritenuto sicuro, per permettere al destinatario di verificare che il documento è giunto intatto, ricalcolando il codice di controllo che deve risultare identico.



SISTEMI SICURI DI TRASMISSIONE



SISTEMI SICURI DI TRASMISSIONE

I certificati digitali

Un certificato digitale è un documento che attesta la relazione di **appartenenza di una chiave pubblica ad certa una entità** (una persona, una azienda, una macchina, ...).

Tale legame è garantito dall'ente emittente il certificato, ovvero una terza parte fidata che costituisce l'autorità di certificazione o **Certification Authority (CA)**.

Un certificato digitale contiene la chiave pubblica ed il nominativo dell'entità di cui viene garantita la corrispondenza, indicazioni relative all'algoritmo utilizzato per la generazione della chiave, una data di scadenza, il nome della CA che ha rilasciato il certificato, il suo numero di serie e la firma digitale della CA stessa a garanzia del fatto che il certificato digitale è stato rilasciato proprio da tale CA. In questo modo, chiunque può verificare l'autenticità del certificato con la chiave pubblica della CA e quindi essere sicuro che la chiave pubblica contenuta nel certificato appartenga proprio all'entità specificata dal certificato stesso.



SISTEMI SICURI DI TRASMISSIONE

GNU Privacy Guard

GNU Privacy Guard, o più comunemente GNUPG o ancora **GPG**, è una suite per la gestione di chiavi crittografiche e della cifratura stessa. Tale strumento costituisce il backend utilizzabile da qualunque altra applicazione che desideri far uso di chiavi per cifrare/decifrare informazioni crittografate.

GPG è un sostituto di PGP (Pretty Good Privacy), un'applicazione creata da P. Zimmermann nel 1991, implementa il protocollo OpenPGP (RFC 2440) utilizzando soltanto algoritmi di cifratura non coperti da patent (DSA, RSA, AES, 3DES, Blowfish, Twofish, CAST5, MD5, SHA-1, RIPE-MD-160 e TIGER). Le chiavi sono memorizzate in insiemi detti portachiavi o keyring che sono rappresentati da file all'interno della directory ~/.gnupg.



SISTEMI SICURI DI TRASMISSIONE

SSH- Secure SHell

Il protocollo SSH permette di accedere in maniera sicura alla shell di una macchina remota. Questo fa sì che tale protocollo sia molto utilizzato dagli amministratori di rete.

Il server che gestisce la comunicazione cifrata è il daemon **sshd**. Esso viene lanciato in genere al boot del sistema e rimane in attesa di una connessione da parte di un eventuale client. Non appena riceve una richiesta di connessione da parte di un client, esso genera un processo figlio che si occuperà di gestire l'autenticazione e la comunicazione con il client ed il processo principale ritornerà in attesa di un'altra eventuale richiesta di connessione.

Il comando `ssh` è un client di comunicazione che, utilizzando il protocollo SSH (SSH1 o SSH2), permette di effettuare il **login su una macchina remota**. Se il login va a buon fine si avrà l'accesso alla shell definita per l'utente che ha effettuato il login e si potrà così interagire con il sistema remoto come se si impartissero i comandi direttamente sulla sua tastiera.



SISTEMI PER LA VERIFICA

Verifica dell'integrità dei file

Attraverso l'accumulo di codici di controllo è possibile verificare l'integrità di file e di directory, contro l'uso improprio del sistema, comprendendo eventualmente l'azione di un virus.

AIDE & tripwire

AIDE e Tripwire sono programmi per la verifica dell'integrità dei file;

attraverso il confronto con le informazioni accumulate precedentemente segnalano le aggiunte, le rimozioni e le alterazioni di file e directory.

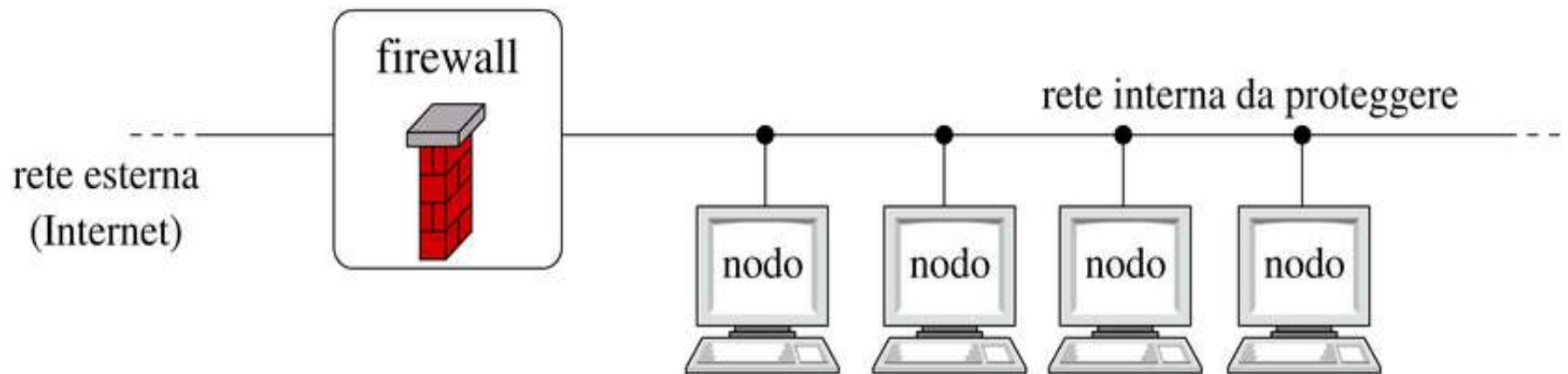
Si tratta di strumenti preziosi per scoprire gli utilizzi impropri del sistema o l'azione di cavalli di Troia.



SISTEMI PER LA PROTEZIONE

il firewall

In commercio esistono apparecchi dedicati a svolgere il compito di firewall, spesso denominati firewall hardware per contraddistinguerli dai software che svolgono tale compito su macchine con sistemi operativi multipurpose, utilizzabili cioè per poterci lavorare in generale, che possono avere, come GNU/Linux, un firewall software. Questo non deve trarre in inganno poiché la politica di firewalling è comunque gestita attraverso un insieme di regole che vengono attuate attraverso un software.



SISTEMI PER LA PROTEZIONE

Esistono almeno due tipologie di firewall

- **Proxy server:** esegue un filtraggio sui pacchetti che lo attraversano, ai livelli più alti dello stack OSI. Un firewall di questo tipo (spesso denominato soltanto proxy server), può permettere, oltre alla gestione di caching delle pagine visitate, anche l'accesso o meno a determinate pagine web basandosi sul contenuto delle stesse.
- **Packet filter:** esegue un filtraggio sui pacchetti che lo attraversano, dal livello fisico fino al livello di trasporto dello stack OSI . Ad esempio, un firewall di questo tipo può scartare i pacchetti che arrivano da interfacce di rete con un determinato indirizzo IP, o far passare soltanto il traffico relativo a determinate porte (TCP o UDP).

Parleremo solo di Firewall su macchine GNU/Linux

Per realizzare un firewall è sufficiente un calcolatore anche di tipo non recente (un compatibile Intel 80486 con 16 Mbyte di RAM, ad esempio) ed una distribuzione con kernel Linux recente. Un firewall non necessita di interfaccia grafica.

Un firewall ha usualmente almeno due schede di rete, una viene collegata verso l'esterno ed una collegata con la nostra rete interna, un esempio è la macchina Digital che vedete qui sul palco, sulla quale funziona anche un server Proxy http.



SISTEMI PER LA PROTEZIONE

Proxy Server

Il filtro a livello applicazione viene comunemente chiamato **server Proxy** o più semplicemente Proxy; questi tipi di server comunicano con la rete esterna per conto degli host della rete interna, in altre parole i Proxy controllano il traffico tra due reti. L'indirizzo che un server esterno riceve è infatti quello del Proxy e questo rappresenta anche un metodo di protezione delle informazioni della rete interna.

Con questo tipo di filtri abbiamo così una netta distinzione tra rete interna e esterna, infatti ogni pacchetto viene ricevuto processato e inoltrato dal Proxy sia verso l'interno che verso l'esterno, non **c'è quindi un collegamento fisico tra le due reti**. I firewall a livello applicazione operano **sui protocolli di livello di applicazione quali HTTP, FTP, SMTP, BOOTP, TFTP, etc.** abilitandoli disabilitandoli o limitandone l'uso.

Per la gestione dei server Proxy bisogna usare applicazioni client che lo supportino, esempio nel client Web della netscape è possibile settare il proxy per la navigazione.



SISTEMI PER LA PROTEZIONE

Il funzionamento di un firewall

Il principale lavoro di un firewall è di esaminare ogni pacchetto che transita nel suo spazio di rete e di controllare che questi pacchetti siano rispondenti a certe restrizioni, chiamate anche *regole del firewall*. I pacchetti che soddisfano questi requisiti sono liberi di passare, quelli che non sono conformi vengono scartati o rifiutati.

Di solito ha una struttura piuttosto semplice, come semplici sono le regole che è in grado di applicare. Ad esempio, per il protocollo TCP potrebbe essere impostato semplicemente per rifiutare o eliminare tutti i pacchetti in arrivo che abbiano il flag SYN impostato, ossia di richiesta inizio connessione. Il risultato è che chiunque richieda una connessione al nostro computer, si vede rispondere o "connection refused" o "connection timed out".

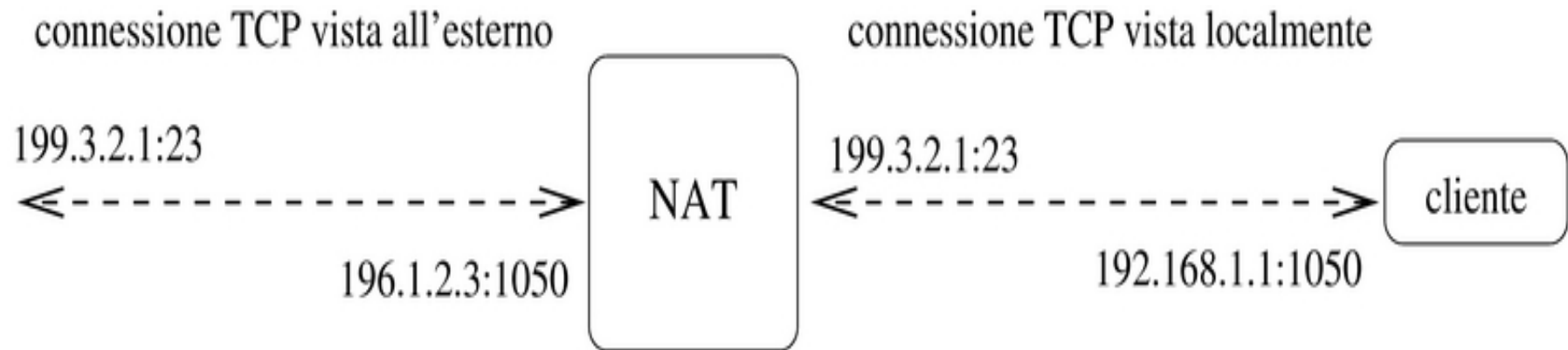
Facciamo una diversa ipotesi. Abbiamo un computer con due interfacce di rete, una per Internet ed una sulla nostra rete interna. Su questo computer è in esecuzione un server FTP che deve essere usato unicamente dagli utenti interni, mentre non deve essere accessibile dall'esterno. In questo caso il firewall è utile e deve essere impostato in modo da bloccare i pacchetti TCP SYN provenienti dall'esterno e diretti sulla porta 21, quella dell'FTP. Se sullo stesso computer è attivo anche un server HTTP sulla porta 80, e deve essere accessibile sia dall'interno che dall'esterno, allora il firewall conterrà una regola che permette il transito a tutti i pacchetti TCP diretti alla porta 80.



SISTEMI PER LA PROTEZIONE

Il funzionamento di un firewall

Questo tipo di firewall usano anche sistemi di traduzione e manipolazione dei pacchetti, chiamati NAT (Network Address Translation), in modo da nascondere gli indirizzi dei computer interni. Il vantaggio è doppio: da un lato si ha la protezione perché i computer interni non sono mai raggiungibili, e dall'altro si usa un solo indirizzo sulla rete esterna.



SISTEMI PER LA PROTEZIONE

Il funzionamento di un firewall

Un sofisticato sistema di filtraggio dei pacchetti di rete è integrato nel kernel di Linux, ed è controllabile con le utility **iptables**, che permettono di inserire, cambiare e cancellare regole a piacere, basandole su un numero impressionante di parametri, relativi al protocollo, alle interfacce di rete, alla sorgente ed alla destinazione, al tipo di servizio, ai flag TCP e tantissimi altri.

Dato che iptables permette soltanto di manipolare singole regole, l'uso presuppone una buona conoscenza dei protocolli IP e del concetto di regole di filtro dei firewall.

Esistono delle interfacce grafiche per aiutare un po' nella configurazione, ma in definitiva occorre sapere dove mettere le mani.

Di contro, la possibilità di inserire regole estremamente dettagliate e mirate, permette di calibrare il comportamento del firewall di Linux in modo efficiente e professionale.



SISTEMI PER LA PROTEZIONE

Qualche dettaglio su iptables

Il meccanismo di funzionamento di iptables non è proprio semplice, per cui farò solo un accenno:

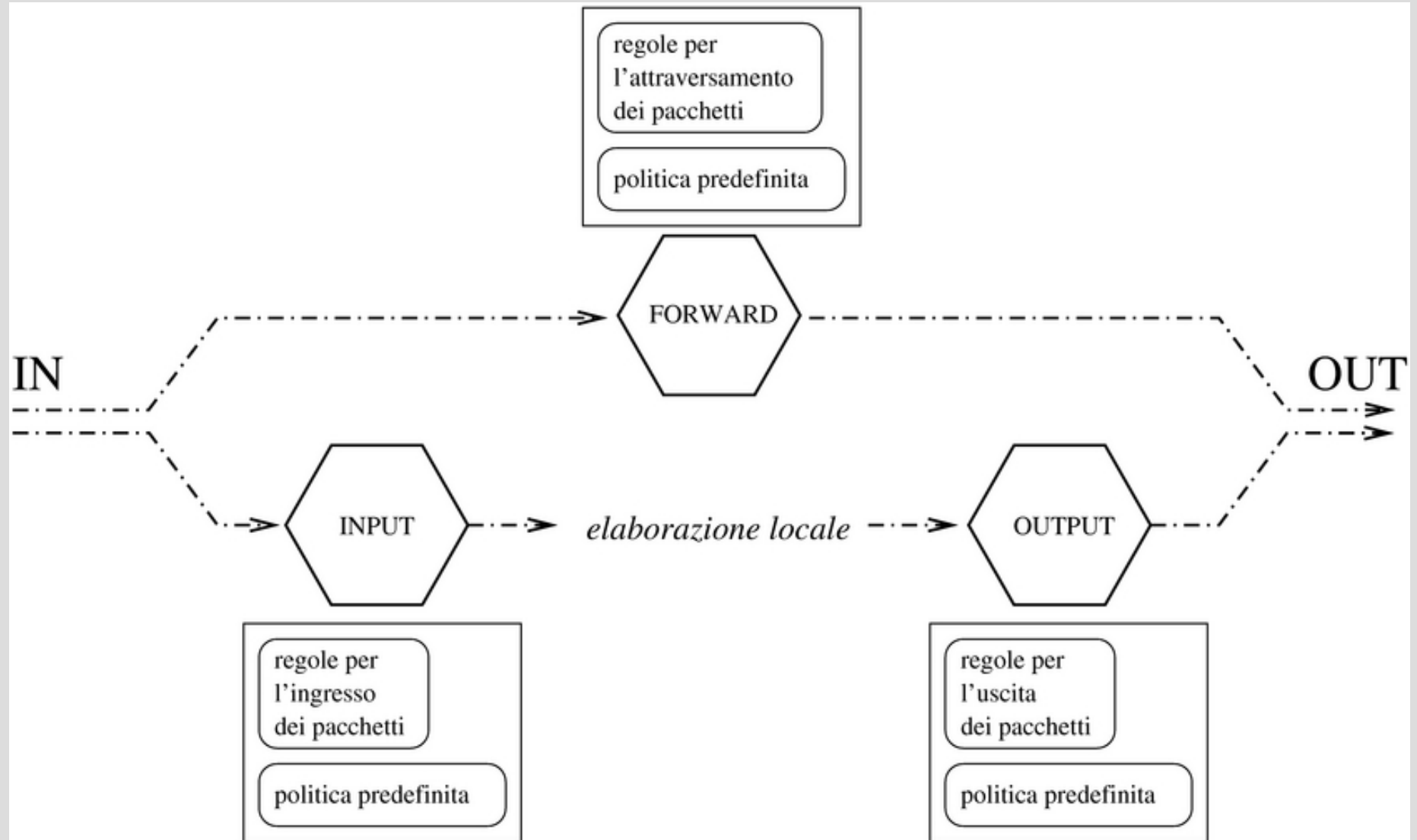
Il filtro è basato su **tre principali tables** (tabelle) denominate *filter*, *nat* e *mangle*, ognuna delle quali può contenere un numero illimitato di regole suddivise in categorie di applicazione.

La tabella relativa alla gestione del firewall è denominata filter e si compone di tre punti di controllo: INPUT, FORWARD e OUTPUT, che indicano rispettivamente i pacchetti in ingresso, quelli in transito e quelli in uscita.

Gli obiettivi più frequenti sono due, ACCEPT e DROP, che rispettivamente si riferiscono al permesso di attraversamento del punto di controllo, oppure al blocco ed eliminazione del pacchetto intercettato.



NETFILTER



NETFILTER

1. Un pacchetto proveniente da un'interfaccia qualunque, diretto allo stesso firewall, è soggetto al controllo di ingresso;
2. un pacchetto passante viene sottoposto al controllo di inoltra;
3. un pacchetto che deve uscire attraverso un'interfaccia del firewall, perché generato da un processo locale, è sottoposto al controllo di uscita.

Quando un pacchetto IP viene analizzato in un punto di controllo e all'interno di questo non c'è alcuna regola che lo prenda in considerazione, la sua sorte è stabilita dalla politica predefinita (policy) per quel contesto.

Dato che i pacchetti che entrano, escono e transitano in un computer in rete possono essere tanti, vengono assegnati a delle categorie, e per ogni categoria possono essere applicate regole differenti. In ogni categoria le regole vengono applicate a catena, da cui la sequenza di regole sotto una categoria viene chiamata *chain* (catena).



NETFILTER

Le tables esistenti sono:

filter con tre chain che sono: INPUT per le regole da applicare ai pacchetti che entrano da una qualsiasi interfaccia e vanno alle applicazioni in esecuzione all'interno del computer, OUTPUT per le regole riguardanti i pacchetti originati dalle applicazioni in esecuzione nel computer e diretti all'esterno tramite una qualsiasi interfaccia di rete, FORWARD per le regole relative ai pacchetti che entrano da una qualsiasi interfaccia di rete e sono destinati ad una altra interfaccia per via del meccanismo di routing.

Nat ha anche lei tre chain: PREROUTING per le regole da applicare ai pacchetti prima del routing, POSTROUTING per le regole da applicare ai pacchetti dopo il routing, OUTPUT per le regole da applicare ai pacchetti, generati dalle applicazioni locali, immediatamente prima che escano da una interfaccia di rete.

Mangle ha ben cinque chain, ossia tutte le chain delle altre due tables, e si usa per manipolare i pacchetti in modi più sofisticati, che al momento non ci interessano.

Ogni chain di ogni tabella ha poi una politica di decisione predefinita, che viene applicata a tutti i pacchetti che non ricadono in nessuna regola. Questa politica può essere:

ACCEPT: accetta il pacchetto, ed è la politica predefinita

DROP: ignora il pacchetto

QUEUE e RETURN al momento non ci interessano



NETFILTER

Se date questo comando:

```
iptables -L
```

vi viene risposto qualcosa di simile:

```
Chain INPUT (policy ACCEPT)
```

```
target      prot opt source                destination
```

```
Chain FORWARD (policy ACCEPT)
```

```
target      prot opt source                destination
```

```
Chain OUTPUT (policy ACCEPT)
```

```
target      prot opt source                destination
```

che ci informa che la table filter (quella di default per il comando iptables) ha impostata su tutte le chain la politica ACCEPT, ossia tutti i pacchetti sono liberi di circolare dentro e fuori dal nostro PC.



NETFILTER

Netfilter è un meccanismo di firewalling dotato di **stateful inspection** dei pacchetti, reso possibile dal **connection tracking**, ovvero un meccanismo con il quale Netfilter tiene traccia dello stato delle connessioni. Tale ruolo è implementato da Conntrack che risiede nel kernel (oppure può essere caricato come modulo ip_conntrack) e consente di gestire gli stati elencati in tabella:

Stato	Descrizione
NEW	indica che il pacchetto è relativo ad una nuova connessione.
ESTABLISHED	indica che il pacchetto appartiene ad una connessione già instaurata.
RELATED	indica che il pacchetto si riferisce ad una connessione già instaurata (ESTABLISHED) – ad es. un pacchetto ICMP che segnala un errore della comunicazione TCP o il flusso dati di un server FTP (la connessione è sulla porta 21 ma il flusso dati avviene sulla porta 20).
INVALID	indica che non è possibile identificare lo stato relativo alla comunicazione alla quale si riferisce il pacchetto (in genere è una buona idea scartare tutti i pacchetti la cui comunicazione è in questo stato).
SNAT	indica che l'indirizzo IP del mittente del pacchetto precedentemente ricevuto è diverso da quello di destinazione del pacchetto in uscita.
DNAT	indica che l'indirizzo IP del destinatario del pacchetto precedentemente inviato è diverso da quello del mittente del pacchetto ricevuto.



NETFILTER

Alcuni esempi: due regole per chiudere verso Internet le porte privilegiate

```
iptables -t filter -A INPUT -p udp --dport 0:1023 -i ppp0 -j REJECT
```

```
iptables -t filter -A INPUT -p tcp --syn --dport 0:1023 -i ppp0 -j REJECT
```

Un generico comando iptables si divide in varie parti che possiamo raggruppare in tre sezioni:

- selezione della table e della chain a cui va applicata la regola
- identificazione di protocollo, interfaccia, sorgente, destinazione
- operazione da fare sui pacchetti che soddisfano la regola

tornando ai nostri due comandi, la **prima sezione** è per tutti e due:

-t filter -A INPUT dove viene selezionata la table *filter* (di default, quindi il -t filter si può omettere) e di questa la chain INPUT, ossia quella su cui transitano i pacchetti provenienti dall'esterno e diretti alle applicazioni all'interno del computer.

La **seconda sezione** è:

-p udp --dport 0:1023 -i ppp0 per il primo comando e

-p tcp --syn --dport 0:1023 -i ppp0 per il secondo.



NETFILTER

Si leggono in questo modo:

- p **udp** (oppure -p tcp) il pacchetto deve essere in protocollo UDP (o TCP per il secondo comando)
- syn** (applicabile solo al protocollo TCP) il pacchetto deve avere il flag SYN attivo ed i flag ACK e FIN inattivi. Indica una richiesta di connessione iniziata dal computer remoto
- dport 0:1023** deve essere diretto ad una porta UDP (o TCP) compresa tra 0 e 1023, ossia le porte privilegiate. La notazione a:b indica un intervallo
- i **ppp0** deve entrare dall'interfaccia ppp0. Occorre prestare attenzione e specificare sempre l'interfaccia di entrata o uscita dei pacchetti quando si inseriscono regole bloccanti. Se non specificate l'interfaccia, la regola viene applicata a tutti i pacchetti che transitano su tutte le interfacce, quindi anche l'interfaccia loopback col risultato che alcuni servizi nel vostro computer smettono di funzionare senza motivo apparente.

L'**ultima sezione** indica cosa fare dei pacchetti che soddisfano la regola ed è denominato *TARGET*: nel nostro caso vengono rifiutati, adottando il normale comportamento dello stack IP ossia il REJECT (quello che segue dopo il parametro -j è il *target*).

Si può usare anche DROP come target, ed in questo caso il pacchetto che soddisfa i requisiti di selezione nella regola viene semplicemente scartato come se non fosse mai arrivato. Con il REJECT, se qualcuno tenta una connessione al nostro computer riceve un *connection refused*, mentre nel secondo caso riceve un *connection timed out* dopo una certa attesa.



POLITICHE DI SICUREZZA

Abbiamo visto un po' di cose, come ho detto e come avrete capito, la sicurezza del vostro computer non è solo nel firewall, o solo nelle password, o chissà dove. E' anche e soprattutto nel *vostro comportamento*. Come potete migliorarlo, aumentando la sicurezza del sistema?

Chiudete le porte che non servono e disattivare i servizi inutili e gestite il software che potrebbe aprire una porta a vostra insaputa

ossia, spegnere o disattivare definitivamente i servizi che non usate. Per esempio, se avete un server telnet (porta 23/TCP), il portmapper (porta 111/TCP e UDP) o un server http (porta 80/TCP) attivi e non li usate, disattivateli in modo permanente. Usate il comando netstat per vedere quali porte di rete avete aperte, e decidete quali non servono. Il modo dipende dal sistema operativo e dalla versione.

Evitare di abilitare l'esecuzione di script e programmi incorporati in documenti ottenuti attraverso la rete (file HTML e posta elettronica principalmente).



POLITICHE DI SICUREZZA

Tenete aggiornato il software

non solo quello di rete. Non deve essere un impegno costante, ma ogni tanto date una occhiata al sito del sistema operativo che avete installato, e controllate che non siano usciti aggiornamenti critici per la sicurezza, ovviamente solo dei software che usate e solo per l'uso che ne fate. Installare un aggiornamento di Apache se non l'avete mai usato, è del tutto inutile, è arrivato anzi il momento di rimuoverlo. Tenete sotto controllo in particolare i browser web, i programmi per la posta elettronica, i programmi per chat e Instant Messenger, i programmi per scambio file P2P (Peer to Peer) e non dimenticate gli stessi firewall ed antivirus.

Attenzione però a non rincorrere l'ultima versione di ogni programma! Non è detto che sia più stabile, anzi, a meno di non aver bisogno dell'ultima funzionalità (o voler partecipare allo sviluppo), evitare l'installazione di prodotti immaturi e attendere che vengano testati in modo sufficiente dalla collettività.



POLITICHE DI SICUREZZA

Usate software originale

in quanto i software non originali ed i vari crack/keygen usati per eliminare le protezioni da copia hanno una probabilità altissima di contenere dell'altro.

Molti virus si propagano tramite reti P2P mascherandosi come programmi per togliere protezioni ai software commerciali.

Quando potete, usate software libero.

Ho giornalmente l'esperienza di persone che girando per siti "poco raccomandabili" in cerca del crack per un software originale, si buscano dialer e spyware a non finire, rimanendo così senza il software tanto agognato e col computer inutilizzabile a meno di una formattazione.

È un problema che tocca poco o nulla chi usa Software Libero ;))

Noi siamo contrari al software pirata, semplicemente (anche perché non ne abbiamo bisogno ;))



POLITICHE DI SICUREZZA

Usate un firewall

Se avete dei servizi attivi che non volete o non potete disattivare. Se avete una connessione ADSL ed usate un computer per la condivisione della connessione Internet al vostro portatile, o alla rete interna della vostra azienda, viene di solito naturale mettere su quel computer anche un servizio di condivisione disco come Samba o NFS e il servizio di smistamento della posta, anche se per motivi di sicurezza è altamente sconsigliato usare lo stesso computer come firewall e come server.

In questo caso un firewall configurato opportunamente aumenta la protezione del vostro computer e della vostra rete.

Ma se un utente della rete interna naviga su un sito "apposito" e contrae un virus o scarica un dialer, il firewall non vi aiuta.

Non affidatevi ciecamente ad uno strumento solo

Un firewall non deve farvi sentire al sicuro. Né vi deve risparmiare di controllare periodicamente gli aggiornamenti per i vostri software. Utilizzare un sistema di scansione realizzato appositamente per verificare le alterazioni nei file, come AIDE e Tripwire



LINUX E LA SICUREZZA

Il vantaggio dell'OpenSource.

Ovviamente, uno dei vantaggi dell'OpenSource è di avere a disposizione il **sorgente del prodotto** ed è quindi possibile risolvere il problema facilmente.

Le aziende che vendono software proprietario affermano che avere accesso al codice sorgente facilita la vita dei crackers in quanto essi non si trovano davanti una scatola chiusa dove devono trovare, a tentoni, eventuali errori ma, avendo il sorgente a disposizione, possono trovare un errore e usarlo per tentare di fare breccia sul computer dove il software è installato.

Ma, in realtà, le cose non stanno proprio così in quanto non è solo il cracker ad avere accesso al sorgente ma è tutta la comunità di Internet quindi quello che ha trovato il cracker lo può trovare anche uno sviluppatore serio che mi contatta e mi dice : "Guarda Ciccio che qui hai sbagliato". Ma se anche nessun'altro se ne accorge e se ne accorge solo il cracker, beh tempo 12-24 ore e state sicuri che sulla rete si possono trovare già due o tre patch, soluzioni al problema perché, appunto, c'è il sorgente.

Se invece si tratta di un software chiuso, allora devo aspettare quei due o tre mesi affinché il produttore rilasci i famosi "service packs" di 20 e oltre MByte.



LINUX E LA SICUREZZA

Grazie per l'attenzione, sono a disposizione per eventuali domande.

“Ci vuole tutta una vita per capire che non è necessario capire tutto.”

(Proverbio cinese)

