Lezione 7

Protezione e sicurezza dei sistemi operativi

- Il problema della sicurezza
- Cenni di crittografia
- Autenticazione
- Protezione delle risorse
- Attacchi alla sicurezza e strumenti di difesa

Il problema della sicurezza

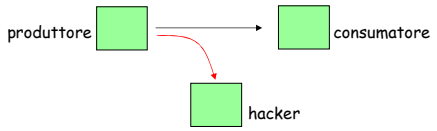
- Il miglior uso delle risorse si ottiene condividendo le risorse stesse
  - Sistemi multiutente
  - Reti di calcolatori
- Tale condivisione ha reso indispensabile introdurre misure di protezione e messa in sicurezza delle risorse
- Il problema delle sicurezza dei calcolatori consiste nel prevenire accessi non autorizzati a risorse e informazioni del sistema al fine di garantire
  - la riservatezza dei dati
  - L'integrita' dei dati
  - L'autenticita' dei dati
  - La disponibilita' del servizio

Il problema della sicurezza

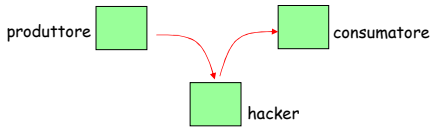
- Il problema della sicurezza puo' essere descritto mediante tre entita':
  - Un produttore di informazioni
  - Un consumatore di informazioni
  - Un cracker o hacker (to hack: rompere, tagliare, fare a pezzi)
- Schema generale che puo' essere applicato sia per sistemi multiutente che per reti di sistemi
- Produttore e consumatore possono coincidere
  - Es: un utente che scrive un file e il giorno dopo lo legge

Minacce alla sicurezza

- Intercettazione: minaccia alla riservatezza
  es: intercettazione in rete o copie non autorizzate

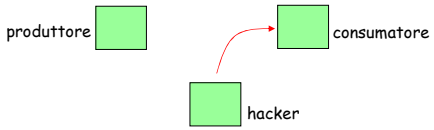


- Modifica: attacco alla integrita'
  Es: alterazioni del comportamento di programmi o dati sensibili

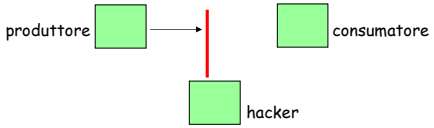


Minacce alla sicurezza

- produzione: minaccia alla autenticita'
  es: aggiungere record a file o produrre falsi messaggi in rete



- interruzione: attacco alla disponibilita'
  Es: saturazione dei servizi o distruzione dei dati



Autenticazione

- La condivisione delle risorse impone una autenticazione (chi puo' fare cosa) per l'accesso ad un sistema
- Ulteriore esempio di "migrazione delle funzionalita'" al crescere delle funzionalita' di un sistema
  - Anni '60 nei mainframe (Multics)
  - Anni '70 nei minicomputer (Unix)
  - Anni'80 – '90 nei PC (Windows e MacOS)
- Un utente puo' essere identificato da
  - Caratteristiche fisiche (es: impronte digitali, retina dell'occhio, firma, timbro della voce)
  - Proprieta' di oggetti (es: carte magnetiche, chiavi e smart card)
  - Conoscenza di dati (es: password, PIN, combinazioni di numeri)

Password

- Password
  - Piu' comune schema di autenticazione
  - L'utente sceglie una "parola chiave" che permette l'accesso al sistema
  - Il sistema confronta la parola chiave fornita dall'utente con quella memorizzata nel database (o file) delle password
- Inconvenienti:
  - Spesso gli utenti usano password facili da trovare
    - Es: Nomi di familiari, targa dell'auto, codice fiscale
    - Inconveniente molto piu' comune di quanto si pensi
    - Una volta nel sistema, un intruso puo' accedere al file delle passowrd degli altri utenti

Possibili soluzioni

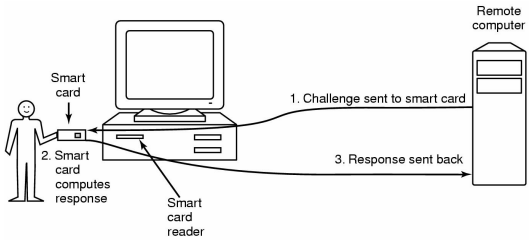
- alcuni sistemi operativi
  - Criptano il file delle password
  - Obbligano a cambiare spesso la password
  - Impediscono di far scegliere vocaboli presenti in vocabolari
  - Impediscono di far scegliere password troppo corte
  - Impongono combinazioni di caratteri e numeri
  - Usano tecniche che inseriscono caratteri in varie posizioni prima della cifratura (password "salate")

Plaintext	*Ciphertext*
password	cGFzc3dvcmQ=
psasaswlortd	cHNhc2Fzd2xvcnRk
newpassword	bmV3cGFzc3dvcmQ=
nsewaplatssewodrd	bnN1d2FwbGF0c3N1d29kcmQ=

Smart card

- Simile ad una carta di credito con un chip
  - Il chip puo' essere utilizzato per memorizzare dati (memory card) e/o istruzioni (microprocessor card)

- Autenticazione "domanda/ risposta"



Biometria

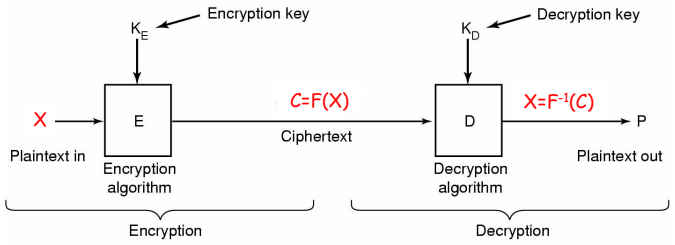
- Usa informazioni fisiche personali per riconoscere gli utenti
  - Impronte digitali, retina, forma del viso e delle mani, …
  - Si confrontano i dati acquisiti da appositi lettori con quelli un database
- Spesso usate in combinazione con altre procedure di identificazione
- La prossima frontiera: analisi del DNA




crittografia

- Il piu' comune metodo per proteggere le password e dati sensibili e' la crittografia (o cifratura)
- In generale la crittografia trasforma un testo da una forma leggibile (il testo in chiaro) ad una forma non leggibile (il testo cifrato) mediante una regola
- Il testo cifrato puo' essere memorizzato in un file oppure trasmesso in rete
- E' necessario poi decodificare il testo cifrato riconducendolo al testo in chiaro utilizzando la stessa regola
- Il testo rimane illegibile a chi non conosce la regola

crittografia



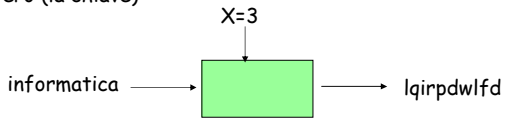
- la cifratura fa uso di una funzione F(X) che e' difficile da invertire
- la cifratura puo' fare uso di chiavi

crittografia simmetrica

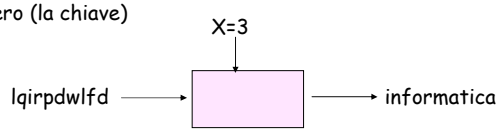
- Noto anche come cifratura a chiave segreta
- Usa la stessa chiave per cifrare e decifrare i messaggi
  - produttore
    - Cifra un messaggio usando la chiave segreta
    - Manda il messaggio al consumatore
  - consumatore
    - Decifra il messaggio usando la stessa chiave segreta
- La segretezza dipende dalla chiave, non dall'algoritmo
- *Esempio: Data Encryption Standard* sostituisce i caratteri e modifica il loro ordine in base alla chiave

Esempio di cifratura simmetrica

- Algoritmo di cifratura: sostituisci ogni carattere con quello seguente X caratteri nell'ordinamento alfabetico, dove X e' un intero (la chiave)



- Algoritmo di decifratura: sostituisci ogni carattere con quello precedente X caratteri nell'ordinamento alfabetico, dove X e' un intero (la chiave)



Inconvenienti della cifratura simmetrica

- Produttore e consumatore devono scambiarsi prima la chiave in forma sicura
- In caso di piu' consumatori e' auspicabile che il produttore usi chiavi differenti
- Chiave relativamente facile da trovare (sufficienti poche ore di CPU)

Approccio alternativo

Crittografia a chiave asimmetrica

- Detta anche cifratura a chiave pubblica
- Si basa sulla difficolta' di fattorizzare numeri interi di grandi dimensioni
  - Chiave privata nota solo al proprietario
  - Chiave pubblica (che dipende dalla privata) nota a tutti
- Esempio: RSA (Rivest, Shamir, Adleman, 1971)
  - La chiave pubblica è *(N, e)*, con *N* prodotto di due grandi numeri primi, *p* e *q*.
  - La decodifica è possibile solo se sono noti i fattori primi, p, q (la chiave privata).

L'algoritmo RSA

- Il consumatore
  - sceglie due interi p e q e calcola N=p*q Sceglie un numero intero d tale che sia primo con (p-1)*(q -1),
  - calcola quindi e come inverso di d modulo (p -1)*(q -1),
- La chiave pubblica del consumatore è (N, e)

- La codifica del messaggio M avviene calcolando $C = M^e \bmod N$ — E' sufficiente la chiave pubblica !!
- La decodifica del messaggio cifrato C con $M = C^d \bmod N$ — E' necessaria la chiave privata d !!

Esempio:

- Il messaggio da inviare e' M=3
- Il consumatore sceglie p=5, q=7 ➔ N=35
- Sceglie poi d primo con 24 ➔ es. d=11
- e inverso di d mod 24 ➔ e=11

Chiave pubblica = (35, 11)

- Messaggio cifrato

$C = 3^{11} \bmod 35 = 12$

- Messaggio decodificato

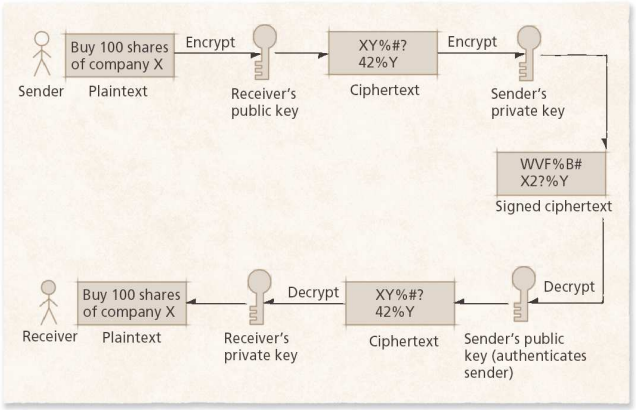
$M = 12^{11} \bmod 35 = 3$

Riservatezza vs autenticita'

- Risolve il problema della riservatezza del messaggio (solo il consumatore possiede la chiave privata) ma non quello della autenticita' (la chiave pubblica e' in possesso di tutti)
- E' possibile dimostrare che il meccanismo funziona anche cifrando il messaggio con la chiave privata e decifrandolo con quella pubblica

E' possibile autenticare il produttore
in base alla sua chiave privata

Autenticazione con algoritmo a chiave pubblica



Crittografia a chiave pubblica RSA-based

- Messaggi difficili da decifrare
  - (richiesti 1757 giorni di calcolo di una CPU a 2 GHz, nel 2002, per una chiave a 64 bit)
- Alla base di numerosi sistemi per le transazioni elettroniche sicure:
  - Siti di e-commerce
  - Firma digitale
  - Protocollo Secure Socket Layer

Firma digitale

- Permette di firmare documenti digitali in modo che non possano essere disconosciuti in un secondo momento (non ripudiabilita'):
  - 1. Il documento viene fatto passare attraverso una funzione di hashing difficile da invertire.
  - 2. Il risultato viene cifrato mediante un meccanismo di crittografia a chiave pubblica; il risultato che si ottiene applicando la chiave privata è detto *blocco della firma*.
- Se vi sono state modifiche, la funzione di hashing fornisce un risultato differente da quello ottenuto decriptando il blocco della firma con la chiave pubblica.

Message Digest (MD5), Secure Hash Algorithm (SHA),...

Secure Sockets Layer (SSL)

- Protocollo che rende sicura la comunicazione tra due computer su internet
- Utilizza algoritmi RSA a chiave pubblica
  - Per autenticare i server mediante un certificato che contiene
    - dati del server
    - Algoritmo di cifratura a chiace pubblica
    - Una scadenza
    - una firma digitale
  - Per proteggere dati e informazioni quando passano su internet (es carte di credito)

Protezione delle risorse

- Una volta autenticato un utente il sistema operativo deve imporre delle limitazioni a cosa puo' fare

Implementazione di politiche di protezione

Ogni processo
- Deve avere l'accesso alle sole risorse per le quali ha l'autorizzazione
- deve poter accedere alle sole risorse di cui ha correntemente bisogno

Principio del privilegio minimo

Privilegio minimo

Il principio del privilegio minimo ha lo scopo
di limitare i danni da parte di processi difettosi

ESEMPI

1. Se un processo P richiama la procedura A , questa deve accedere solo alle variabili di A e non a tutte le variabili di P
2. Se un processo P richiama un compilatore, al compilatore deve essere consentito l'accesso solo al file che deve compilare. Mentre al processo P non deve essere consentito l'accesso a dati e strutture dati del compilatore

Domini di protezione

Il principio del privilegio minimo e' realizzato mediante
i domini di protezione

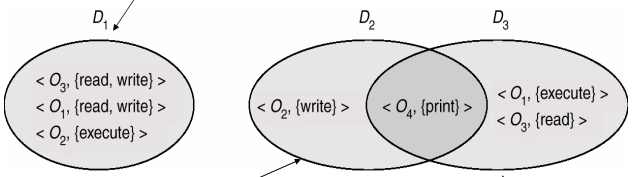
Diritto d'accesso = <nome-oggetto, insieme-diritti>
dove insieme-diritti è un sottoinsieme di tutte le operazioni che possono essere eseguite da un processo sull'oggetto.

Dominio di protezione : insieme di diritti d'accesso

Un dominio di protezione è un insieme di operazioni che possono essere effettuate da un processo su un oggetto e determina le risorse a cui esso può accedere.

esempio

I processi nel dominio D1 possono
- leggere e scrivere O3
- leggere e scrivere O1
- eseguire O2



I processi nel dominio D2 possono
- scrivere O2
- stampare O4

I processi nel dominio D3 possono
- stampare O4
- Eseguire O1
- leggere O3

Struttura dei domini di protezione

- L'associazione tra processi e domini può essere
  - **Statica**: bisogna soddisfare il principio del privilegio minimo fin dall'inizio.
  - **Dinamica**: deve essere disponibile un meccanismo per passare da un dominio all'altro.
- La realizzazione dei domini può avvenire in diverse maniere:
  - dominio = utente
  - dominio = processo
  - dominio = procedura
- In un sistema operativo multiprogrammato il duplice modo di funzionamento determina due domini.
- Due domini non sono sufficienti ad assicurare la protezione tra utenti.

esempio

- Sistema che gestisce gli esami degli studenti
- 3 ruoli (domini)
  - Docenti, studenti, commissione
- 2 oggetti
  - Compiti, valutazioni
- 3 permessi
  - Lettura, modifica , creazione
- I docenti creano, leggono e modificano sia compiti che valutazioni
- Gli studenti leggono le valutazioni e leggono e creano i compiti
- La commissione legge e modifica le valutazioni

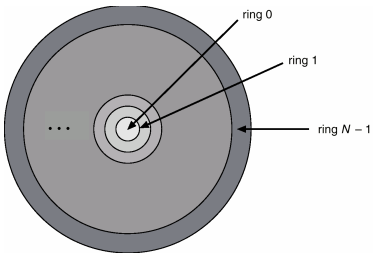
Role Based Access Control
(modello RBAC)

Altri modelli di controllo

- Controllo discrezionale (modelo DAC)
  - Il creatore di un oggetto determina e controlla i permessi per quell'oggetto
  - Es. UNIX mediante comando chmod
- Controllo obbligatorio (modello MAC)
  - Schema di permessi centralizzato
  - Impiegato in sistemi ad alta sicurezza

Esempio MULTICS

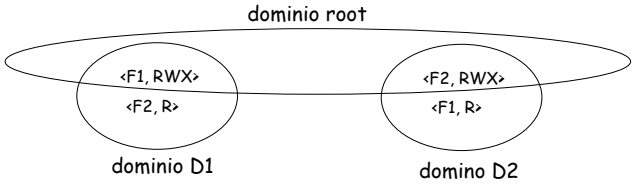
1. 8 domini "concentrici"



- Anello 0 ha i maggiori privilegi
- Se j < i allora $D_i \subseteq D_j$
- Un cambio di dominio e' permesso solo a condizioni molto rigide
- Con 2 livelli si ha il modello di UNIX
- Impossibilita' a realizzare il principio del privilegio minimo

Esempio UNIX

- Ad ogni utente e' associato un dominio
- All'utente root sono associati tutti i diritti di accesso
- Esempio: 2 utenti D1 e D2
- File F1 di D1 e file F2 di D2



Matrice di accesso

- Il piu' semplice modo per realizzare un modello di protezione e' basato sulla

Matrice di controllo degli accesi

- Collega oggetti e domini con gli appropriati diritti di accesso

domain \ object	F_1	F_2	F_3	printer
D_1	read		read	
D_2				print
D_3		read	execute	
D_4	read write		read write	

Cambio di dominio

- Mediante la matrice degli accessi e' possibile anche definire a chi e' permesso effettuare un cambio di dominio

domain \ object	F_1	F_2	F_3	laser printer	D_1	D_2	D_3	D_4
D_1	read		read			switch		
D_2				print			switch	switch
D_3		read	execute					
D_4	read write		read write		switch			

Modifica dei domini

- Mediante la matrice degli accessi e' possibile anche modificare i domini

Ai processi in esecuzione nel dominio D2 e' permesso estendere i permessi di lettura dell'oggetto F2 al dominio D3

domain \ object	F_1	F_2	F_3
D_1	execute		write*
D_2	execute	read*	execute
D_3	execute		

(a)

domain \ object	F_1	F_2	F_3
D_1	execute		write*
D_2	execute	read*	execute
D_3	execute	read	

(b)

Realizzazione della matrice degli accessi

1. tabella globale: una tabella che contiene tutti i permessi
   - Matrice sparsa e di grandi dimensioni

Approcci alternativi

2. Leggere la matrice di accesso per colonne (lista di controllo degli accessi)
   - Ogni oggetto ha l'elenco delle azioni consentite

3. Leggere la matrice di accesso per righe (lista delle abilitazioni)
   - Ogni processo ha le sue abilitazioni

Lista di controllo degli accessi

- lista di controllo degli accessi che conserva solo i diritti effettivamente esistenti

```
File A:
   <Alice, {read*, write*}>
   <Bob, {read*, write}>
   <Chris, {read}>
File B:
   <Alice, {read*, write*}>
   <Bob, {read*, write}>
   <David, {read}>
Printer:
   <Alice, {print*}>
   <Bob, {print}>
   <Chris, {print}>
```

- E' possibile "distribuire" la matrice degli accessi sugli oggetti (es. Unix)
- Maggiore flessibilita' ed efficienza

Attacchi alla sicurezza

- I principali attacchi alla sicurezza di un sistema sono
  - criptoanalisi
  - Cavalli di Troia e login camuffate
  - Sfruttamento della vulnerabilita' del software
  - Virus e worms
  - Negazione del servizio

criptoanalisi

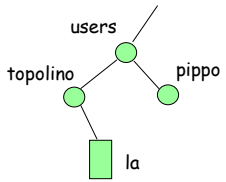
- Un attacco di criptoanalisi
  - Tenta di decifrare un testo cifrato senza possedere le chiavi
  - principali approcci
    - Conoscenza dell'utente
    - Forza bruta
    - Analisi dell'algoritmo di cifratura (spesso noto) per individuare relazioni tra i bit della chiave e i bit del testo cifrato
    - Analisi statistiche
- Contromisure
  - Data di scadenza
  - Gestione attenta delle chiavi

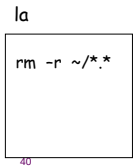
Cavalli di Troia

- Programma che viene fatto avviare da utenti ignari con dei trabocchetti (es. sfruttando l'ordine di ricerca degli eseguibili nel path, o sfruttando errori di battitura)

Esempio:
Un utente pippo con path
path = . : /bin : /usr/bin : /usr/local/bin : /usr/bin/X11 ……

Che per sbaglio digita la invece di ls nella directory di topolino





login camuffate (login spoofing)

- Programmi che attivano false schermate di login che catturano login e password degli utenti e poi si disattivano
- Versioni moderne chiedono di collegarsi a falsi siti web delle banche per aggiornare login e password

Date: Sat, 5 Oct 2001 05:41:06 GMT
From: banca federico II <federico2@banche.it>
Subject: aggiornamento password

Gentile utente

a seguito dell'aggiornamento del software dei nostri server, e' caldamente invitato a confermare la sua login e la sua password presso il nostro sito web all'indirizzo

http://federico2.banche.it

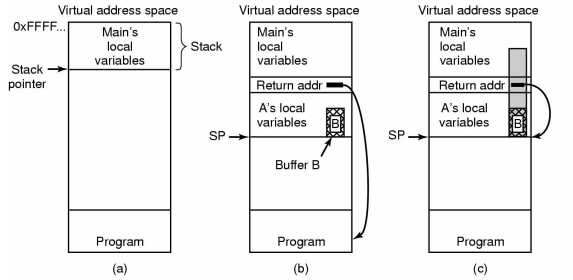
In caso contrario non saremo piu' in grado di garantire i servizi di banca on line da lei sottoscritti.

Cordiali saluti

Sfruttamento della vulnerabilita' del software

- Attacco con overflow del buffer
  - Avviene quando si inviano ad un programma piu' dati di quanto un buffer puo' contenere, corrompendo o sovrascrivendo dati o codice esistente
  - Un buffer overflow ben disegnato puo' rimpiazzare il codice eseguibile in una applicazione modificandone il comportamento
  - A secondo dell'utente e dell'applicazione si puo' ottenere accesso all'intero sistema
- Contromisure
  - In fase di sviluppo del software bisognerebbe verificare la presenza di punti deboli (es. Controllare la lunghezza delle stringhe in input)

Esempio di buffer overflow



a) Programma in esecuzione
b) Viene chiamata la procedura A e l'indirizzo di ritorno e' memorizzato nello stack
c) Viene scritto il buffer B con una quantita' di dati superiore alla sua capacita' ➔ si sovrascrive l'indirizzo di ritorno, che modificato opportunamente puo' attivare il codice accuratamente memorizzato nel buffer B

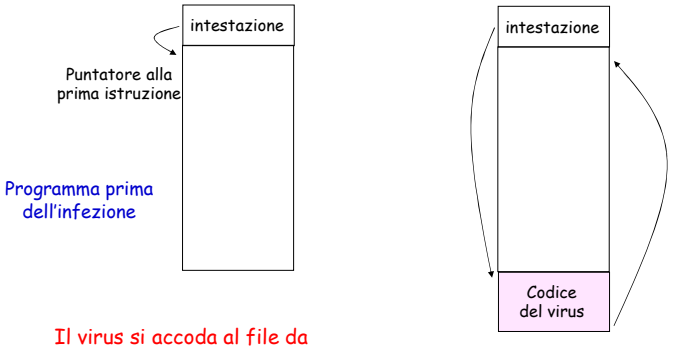
Virus

- Programma che si puo' riprodurre, attaccando il suo codice ad un altro programma
- Si diffonde spedito come allegato di posta elettronica o nascosto in file
- Per replicarsi si allega o sovrascrive altri file
- Si attiva a seguito di una specifica azione (apertura di un allegato o esecuzione di un programma)
- Danni di vario tipo, da quelli "goliardici" a quelli criminali di distruzione dell'hard disk
- Alcuni virus, detti virus polimorfici, in fase di replicazione possono cambiare aspetto (ad esempio aggiungendo istruzioni inutili)

Vari tipi di virus

- Virus compagni
  - Si attivano quando un dato programma viene eseguito, ma non infetta i file
- Virus di programmi eseguibili
  - I piu' comuni. Si nascondono all'interno di altri programmi e si attiva quando il programma ospite viene eseguito
- Virus residenti in memoria e nel settore di avvio
  - Sempre residente in memoria, nascondendosi alterando le bitmap e si attiva quando avviene una interruzione
- Virus delle macro
  - Codificati all'interno di macro di documenti Word o Excel
- Virus del codice sorgente
  - Invece degli eseguibili infetta file sorgenti C

Esempio di virus degli eseguibili



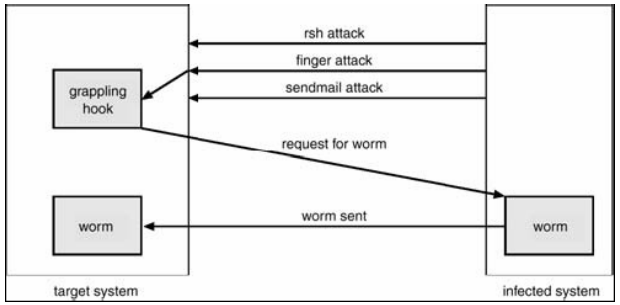
Il virus si accoda al file da infettare modificando i puntatori alle istruzioni

worms

- Codice eseguibile che si diffonde infettando file di sistemi connessi in rete
- Non richiede una azione specifica per attivarsi ma si propaga da solo
- Sfrutta le caratteristiche del networking di UNIX (accesso remoto) e i bug di alcuni programmi come finger e sendmail.

Esempio (worm di Morris, 1988)

- Composto da due programmi
  - Un rampino che sfrutta la vulnerabilita' di alcuni programmi UNIX per installarsi sul sistema vittima
  - Il programma worm che viene caricato dal rampino



Software Antivirus

- Varie tecniche per trovare e rimuovere un virus, ma nessuna completamente efficace
  - Ricerca di istruzioni caratteristiche nel codice del virus (firma del virus)
    - Richiede conoscenze della struttura del codice del virus
    - Usa una lista di virus noti, ma puo' essere inefficace contro varianti e virus polimorfici
  - scansione euristica
    - Individua e sospende i programmi con comportamento anomalo (es. replicazione, residenza in memoria..)
    - Efficace contro virus nuovi ma soggetto a molti falsi allarmi
- La maggior parte dei software antivirus usa entrambe le tecniche

Attacchi per l'interruzione del servizio (DOS)

- Denial of service (DOS)
  - Impedisce al sistema di servire richieste legittime
  - Molto diffuso verso web server (es. Yahoo e eBay nel 2000)
  - Nella maggior parte dei casi l'attacco e' portato inondando il server di richieste saturando le risorse di rete
  - Richiede una rete di computer che lavorano contemporaneamente
  - Non distruggono informazioni ma possono bloccare sistemi critici come quelli per telecomunicazioni o centri di controllo del traffico aereo
  - Varianti: modifica delle tabelle di instradamento dei router o modifica degli indirizzi dei DNS
- Contromisure:
  - firewall per il filtraggio dei pacchetti
  - Alcuni presenti negli stessi s.o. (ad es. Windows XP)

Firewalls

- Protegge una rete locale da intrusi esterni alla rete
- Politiche possibile per un firewall
  - impedire la trasmissione o ricezione di dati per cui non e' stata espressa autorizzazione
  - Permettere la trasmissione o ricezione di tutti i dati percui non e' stato espresso un divieto
- Tipi di firewall
  - firewall per il filtraggio di pacchetti
    - Ispeziona i pacchetti alla ricerca di inconsistenze
    - Seleziona i pacchetti soprattutto in base all'indirizzo, trascurando gli attacchi provenienti da sistemi fidati
  - Barriera a livello di applicazione
    - Ispeziona i pacchetti alla ricerca di codice dannoso

Altri strumenti di protezione

- Sistemi per la scoperta di intrusioni (IDS)
  - Controllano la rete e registrano le richieste di particolari servizi alla ricerca di comportamenti anomali
- Aggiornamenti della sicurezza (patches)
  - Mirano a correggere errori e bug nei sistemi operativi trovati successivamente al rilascio (es. windows update)
- File system sicuri
  - Uso di crittografia per la protezione di un file system. Ad ogni utente e' assegnata una chiave e un certificato

Classificazioni della sicurezza

- "The Orange Book"
  - Nome ufficiale "Department of Defense Trusted Computer System Evaluation Criteria"
  - Definisce le caratteristiche di sicurezza dei sistemi operativi
  - I sistemi sono classificati sulla base di 4 livelli di sicurezza
    - A, B, C and D con varie sottoclassi
    - Livello D = sistema non sicuro
    - Livello A = sistema molto sicuro
    - UNIX e Windows XP ben configurati sono classificati C2