Reti di calcolatori

Gestione e sicurezza nelle reti di calcolatori

Prof.ssa Simonetta Balsamo
Dipartimento di Informatica
Università Ca' Foscari di Venezia
balsamo@dsi.unive.it
http://www.dsi.unive.it/~reti

Gestione della rete

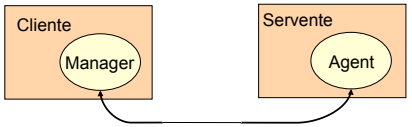
Controllo e monitoring della rete, componenti hw e sw
Identificazione di errori e delle loro cause
Problemi: **eterogeneità** della rete
dimensione
errori temporanei, meno gravi, intermittenti: più difficili da identificare
mascheramento degli errori da parte dei protocolli di rete
-> peggiori prestazioni per l'intera rete (tutti i nodi)

Software di gestione di rete
controllo delle componenti della rete (host, router, linee, bridges,…)
raccolta di informazioni sullo stato
derivazione di statistiche

Gestione della rete

Monitor
controllo basato su *monitoring*
basato sul protocollo cliente-servente - livello applicazione

manager (su client) programma del gestore di rete
agent (su server) programma eseguito sulla periferica di rete da monitorare



Monitor di rete

Il monitor usa le stesse modalità e funzioni da testare

In caso di guasto hw che interrompe la comunicazione, qualsiasi livello lo può rilevare

L'uso degli stessi protocolli di livello trasporto fa sì che il monitor osservi gli stessi ritardi degli altri pacchetti applicativi, rilevando eventuali ritardi anomali

SNMP **S**imple **N**etwork **M**anagement **P**rotocol
protocollo di gestione su Internet, definisce la interazione fra manager e agent
I messaggi sono in codifica ANS.1 (Abstract Syntax Notation.1)

Paradigma *fetch-store*

Due operazioni basilari:

- fetch — recupera il valore da una componente
- store — memorizza un valore in una componente

le operazioni di controllo sono *side effect* di memorizzazioni

Ogni oggetto ha un nome unico ed un valore che può essere ottenuto con *fetch* o memorizzato con *store*

Lo stato di una componente è descritta da oggetti con il loro valore

Identificazione degli oggetti

MIB **M**anagement **I**nformation **B**ase
insieme di informazioni (oggetti) utilizzate da SNMP

Separazione fra protocollo di comunicazione e definizione di oggetti

Paradigma *fetch-store*

SNMP non specifica le variabili MIB, ma solo il formato e la codifica

Le singole variabili MIB e come accedere (*fetch, store)* sono definite da un altro standard

-> flessibilità
-> adattabilità
-> possibilità di inserzioni

Oggetti MIB - specifica ASN.1

Gruppi di variabili corrispondono a protocolli (es. ARP, TCP, UDP, IP,...) o ad hw di rete (Ethernet, FDDI, Token Ring,…) o hw di componenti (bridge, printer,…)

Sicurezza

Definizione di rete sicura:

- accesso alle informazioni
- modifica
- …

Definizione di **politica** di **sicurezza**
cosa proteggere, non **come** implementare la protezione

Politica di sicurezza della rete ⟷ politica di sicurezza dell'host

Sicurezza: protezione delle risorse e delle informazioni
Definizione:

- del valore della informazione
- priorità di protezione
- complessità accettabile

Compromesso fra sicurezza e facilità di uso

Sicurezza

Esempi:
integrità di dati (cambiamenti)
disponibilità dei dati (guasti)
riservatezza, privatezza dei dati

Risorse:
di esecuzione
di memorizzazione
di comunicazione

Aspetti di sicurezza:
autenticazione
autorizzazione
riservatezza (privacy)
disponibilità (availability)
integrità
paternità

Autenticazione

Autenticazione

Verifica l'autenticità, ovvero l'identità dell'utente
- conoscenza di una informazione privata (password)
- uso di oggetti (smart card)
- uso di informazione privata personale fisiologica
(impronta digitale, fondo retina,…)

Mutua autenticazione

Autorizzazione

Specifica le azioni permesse dall'utente

Differenza fra autorizzazione e autenticazione.

Riservatezza

Riservatezza

Evitare l'accesso non autorizzato (lettura) alle informazioni.
Interpretazione delle informazioni (accesso e docodifica)

Integrità

Evitare la modifica non autorizzata (scrittura) delle informazioni
Garantire la comunicazione corretta.

Disponibilità

Garantire l'accesso e l'uso delle risorse con continuità

Paternità

Sicurezza e organizzazione

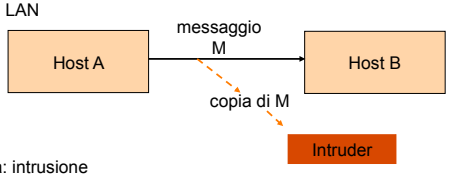
Definizione responsabilità e controllo della informazione
chi è responsabile
come viene **controllata** la responsabilità
es: accounting, autorizzazione

Integrità
a diversi livelli sono stati introdotti i **meccanismi di controllo degli errori**
es : CRC, checksum,…
-> garanzie parziali
possibili falsi positivi (errore di trasmissione sul valore del checksum)
possibili falsi negativi (attacco che falsifica il CRC o checksum su dati modificati)

Controllo di accesso tramite password
Accesso alle risorse. In un singolo host il controllo della password è più semplice rispetto alla rete

Sicurezza e organizzazione

Esempio: LAN



Problema: intrusione

Soluzione: cifratura del messaggio

Problema: controversia di comunicazione
A e B possono mentire senza che l'altro possa dimostrarlo

Soluzione: A firma il messaggio, leggibile da tutti

Tipi di intrusione

Tentativi di intrusione
- passivi
  - lettura di informazioni altrui
  - analisi del sistema e del traffico di rete
- attivi
  - modifica di informazioni altrui
  - cancellazione di informazioni altrui
  - analisi del sistema e del traffico di rete
  - falsificazione di identità
  - accesso non autorizzato a risorse
  - blocco di servizi altrui

Principio **minimo** di sicurezza:
- **protezione** dagli attacchi **passivi**
- **riconoscimento** degli attacchi **attivi**

Metodi e strategie di sicurezza

Accesso improprio a risorse:
- Leaking — acquisire informazioni senza autorizzazione
- Browsing — lettura di tutti i pacchetti
- Interferencing — ricavare informazioni dai dati stessi
- Masquerading — fingersi un altro utente

Politiche di protezione dei dati:
- accesso ai dati in base all'id (*user id*) e al gruppo di appartenenza
- livelli di protezione crescente
  - es*: unclassified, classified, secret, top secret*
- Orange Book — riservatezza dei dati e *mandatory access control*, sviluppato in ambiente militare US
- applicazione dei criteri si sicurezza dell'Orange Book
- Red Book — a sistemi interconnessi in rete
- Lavander Book — a sistemi di basi di gestione di basi di dati (DBMS)

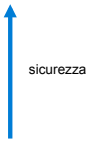
Definizione di criteri comuni

Orange Book

Standard per sistemi aperti e sicurezza
Livelli e protezione - struttura di progetto incrementale:
ogni livello ingloba gli elementi di sicurezza dei precedenti e vi aggiunge quelli specifici del livello.

Si assume che gli utenti del sistema possano essere in malafede

Categorie	Caratteristiche
A	informazioni classificate, modello di sicurezza
B	sistemi non discrezionali, protezione obbligatoria dal sistema
C	sistemi discrezionali, protezione a discrezione dell' utente
D	non sicuri



Orange Book

Classe C (protezione possibile)
- autenticazione degli utenti (pw)

Classe C1
- controllo degli accessi alle risorse
- controllo dell'accesso alla memoria

Classe C2
- controllo di accesso alle risorse con diversa granularità (utente)
- cancellazione della memoria prima dell'assegnamento all'utente
- auditing

Classe B (protezione possibile)

Classe B1
- diritti associati ad ogni oggetto (processi, risorse) controllati dal SO
- i dispositivi devono trattare tali diritti

Classe B2
- trusted path fra SO e utente
- dinamica del livello di sicurezza dei processi (notifica al SO)
- nucleo del SO strutturato (security kernel)
- canali e banda
- controllo stretto delle modifiche di sistema per le parti critiche alla sicurezza

Classe B3
- controllo degli accessi con *overrule*
- audiitng attivo
- secure crashing e restaring

Classe A (modello formale di sicurezza)
- come sopra ma con verifica di progetto

POSIX

POSIX.6 **Protection, Audit and Control Interface Standard**

per realizzare applicazioni portabili
definizione di **interfacce** per la gestione delle informazioni di sicurezza

auditing
discretionary access control
mandatory access control
information labels
privilege

Stabilire i diritti d'accesso: **ACL** **A**ccess **C**ontrol **L**ist
lista dei diritti di accesso per ogni oggetto
(non specifica implementativa)

Controlla i diritti di accesso alle risorse da parte dei gruppi e degli utenti
Permette ad alcuni utenti di eseguire operazioni 'sensibili' alla sicurezza solo in certe condizioni e per un tempo dato.
Es: Unix *super user* : meccanismo di *overridding* dei privilegi

POSIX

mandatory access control

Il sistema (e non il processo) impone e assicura la protezione degli oggetti
soggetto=processo
oggetto=file, processo
Ogni soggetto e oggetto ha una etichetta MAC (mandatory access control)

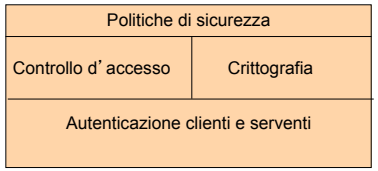
Etichette ordinate: se L1>L2 un soggetto senza privilegi non può rendere disponibile l'informazione con etichetta L1 ad un soggetto con etichetta L2

Restrizioni di accesso ai file
lettura consentita solo se l'etichetta del processo lettore L1>L2, etichetta del file (*no read up*)
scrittura consentita solo se l'etichetta del processo lettore L1<L2, etichetta del file (*no write down*)

Es: un processo *secret* non può leggere da un *top secret*, e non può scrivere su un file con etichetta *confidential* (*confidential* < *secret* < *topsecret*)

Sicurezza nel modello cliente/servente

Sicurezza della comunicazione : **canale**
Comunicazione basata sul **sospetto reciproco**
Messaggi inviati e autorizzazioni usate **una volta sola** per evitare/limitare le intrusioni



Crittografia e codici

Codici di Sostituzione
permutazione sull'alfabeto del testo chiaro
simbolo -> byte alfabeto -> 256 simboli
trasformazione dei simboli; non cambia il loro ordine
256! permutazioni possibili
chiave di cifratura: **tabella** di corrispondenza dei simboli (256x2)
Es. Codice di Cesare, sostituzione monoalfabetica che sostituisce
ad ogni carattere quello successivo di 3 posizioni nell'ordine alfabetico

Codici di Trasposizione
stabilito un periodo P di trasposizione, si sceglie una permutazione su [1, P] e si fa corrispondere alla successione 1,2,…,P la sua permutazione
Es. P=5 la successione 1 2 3 4 5 è sostituita p.es. da 5 3 4 1 2
ad ogni carattere quello corrispondente nella permutazione
i simboli non cambiano, cambia il loro ordine
il testo è diviso in blocchi di P byte, anagrammando ciascuno secondo la permutazione scelta
La decodifica utilizza la permutazione inversa

Crittoanalisi

Derivare il testo in chiaro dal testo cifrato, senza conoscere la chiave di lettura
Utilizza qualche conoscenza

con solo testo cifrato	proprietà statistiche del linguaggio usato probabilità di occorrenza delle parole
con testo in chiaro noto	parole certamente presenti nel testo in chiaro e testi cifrati successivamente resi in chiaro e pubblicati (testo_in _chiaro, testo_cifrato)
con testo in chiaro scelto	testi cifrati corrispondenti a un qualunque testo in chiaro utile (testo_in _chiaro scelto, testo_cifrato)
con testo cifrato scelto	testo cifrato utile e corrispondente testo in chiaro (testo_in _chiaro, testo_cifrato scelto)

Crittografia

Crittografare il messaggio
rendere un messaggio non intelleggibile a chiunque eccetto il destinatario

Diversi metodi

codifica basata su **chiavi** e **algoritmi** di codifica e decodifica

a chiave segreta **(algoritmi simmetrici)**
a chiave pubblica **(algoritmi asimmetrici)**
con funzioni hash

Problemi:
segretezza della chiave
segretezza dell'algoritmo

Crittografia a chiave segreta

L'algoritmo di cifratura deve essere noto
Nessun sistema è **assolutamente sicuro**
(impraticabilità dei sistemi teoricamente sicuri)
L'attacco deve essere **irrealizzabile** in pratica
(tempo di vita delle informazioni)

Crittografia a chiave segreta condivisa

A codifica il messaggio *M* con la funzione codifica *Encrypt*

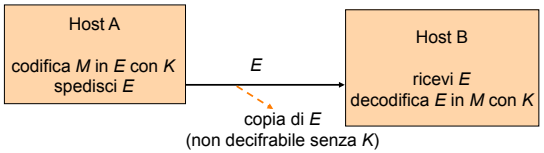
E = Encrypt (K, M)

B decodifica il messaggio *M* con la funzione (inversa) *Decrypt*

M= Decrypt(K, E)

ovvero: *M = Decrypt(K, Encrypt (K, M))*

Crittografia a chiave segreta



A e B condividono *K*

Problemi: comunicazione della chiave segreta, autenticità del mittente

Crittografia a chiave pubblica

chiave privata e chiave pubblica
chiave privata **segreta**
chiave pubblica, nota in associazione al nome dell'utente

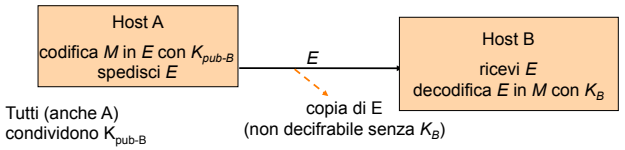
Un messaggio codificato con chiave pubblica non si decodifica facilmente senza la chiave privata
Un messaggio codificato con chiave privata si decodifica solo con chiave pubblica

One way property non si riesce a falsificare la codifica di un messaggio con chiave privata anche se è nota la chiave pubblica
Es. $K_{pub\text{-}A}$ sia la chiave pubblica e K_A la chiave privata di A

$$M = Decrypt(K_{pub\text{-}A}, Encrypt(K_A, M))$$

$$M = Decrypt(K_A, Encrypt(K_{pub\text{-}A}, M))$$

Crittografia a chiave pubblica



Tutti (anche A) condividono $K_{pub\text{-}B}$

Maggior costo rispetto alla cifratura simmetrica

Autenticazione e firma digitale

Firma digitale
meccanismo di autenticazione del mittente di un messaggio
il mittente codifica con chiave privata (ciò garantisce l'autenticazione)

Per evitare copia dei messaggi, si codifica il messaggio insieme alla marca temporale con data ed ora della creazione del messaggio

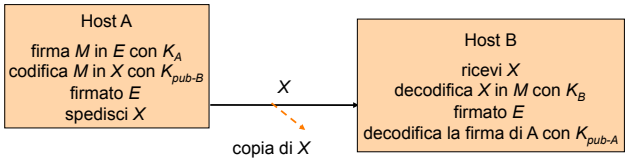
Per assicurare sia l'autenticazione del mittente che la privatezza del messaggio si possono usare **due livelli di codifica**:

per un messaggio M da A a B, siano K_A e K_B le chiavi private di A e B, $K_{pub\text{-}B}$ e $K_{pub\text{-}B}$ le chiavi pubbliche di A e B

A calcola X come $X = Encrypt(K_{pub\text{-}B}, Encrypt(K_A, M))$

B decodifica X come $M = Decript(K_{pub\text{-}A}, Decrypt(K_B, X))$

Autenticazione e firma digitale



Esempio di algoritmo

Prodotto di due interi (grandi, $>10^{100}$) p e q
N = p * q Z = (p-1) * (q-1)
d intero primo rispetto a Z
i i*d=1 mod Z (minimo intero di tipo nZ+1, n>0, divisibile per Z)

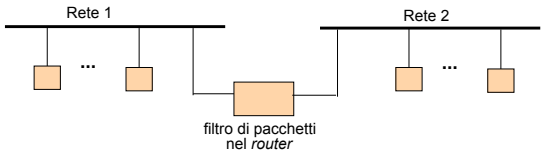
Messaggio
decomposto in blocchi di f bit, con $2^f<N$
calcola di ogni blocco
la codifica, con K_{pub} = (N, i), e con
$M= C^i \bmod N$
la decodifica K_{priv} = (N, d), e con
$C= M^d \bmod N$
le due chiavi sono inverse, K_{pub} pubblica, K_{priv} privata

Es. la fattorizzazione di un numero di
200 cifre richiede ca 4.000.000.000 di anni
500 cifre richiede $O(10^{25})$ anni

Filtro di pacchetti

Prevenzione dell'accesso libero a host o a servizi

filtro dei pacchetti
sw che evita che i pacchetti passino da una rete all'altra tramite il router
la configurazione del manager specifica a quali pacchetti si applica il filtro



Filtro di pacchetti

Analisi dell'intestazione del pacchetto nel campo *header*, identificando *sorgente e destinazione*

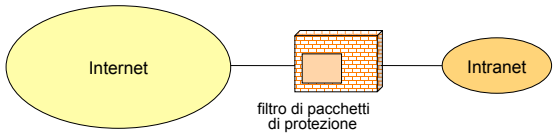
Per il filtro sui servizi:

analisi del protocollo o del servizio di alto livello corrispondente al pacchetto
Si seleziona così il traffico

Es. filtro su tutti i pacchetti che richiedono uso di servizi web
Anche filtri basati su espressioni booleane di sorgenti, destinazioni e servizi

Firewall

Filtro dei pacchetti utilizzato per proteggere una **rete** di una organizzazione da traffico Internet qualsiasi e non desiderato



Internet firewall
su ogni connessione della rete verso l'esterno
numero limitato di host interni che accedono all'esterno (minimo uno), che devono essere sicuri