

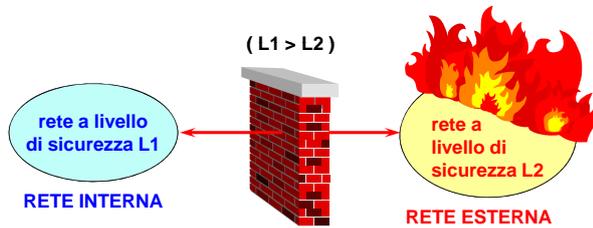
Firewall e IDS/IPS

Antonio Lioy
< lioy @ polito.it >

Politecnico di Torino
Dip. Automatica e Informatica

Che cos'è un firewall?

- firewall = muro tagliafuoco
- collegamento controllato tra reti a diverso livello di sicurezza = sicurezza del perimetro



Ingress vs. Egress firewall

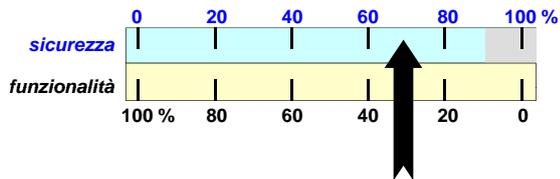
- **ingress firewall**
 - collegamenti incoming
 - tipicamente verso servizi offerti all'esterno
 - talvolta come parte di una comunicazione attivata dall'interno
- **egress firewall**
 - collegamenti outgoing
 - controllo dell'attività del personale
- **distinzione facile per servizi orientati al canale (es. applicazioni TCP), difficile per servizi basati su datagrammi (es. ICMP, applicazioni UDP)**

Progettazione di un firewall

Un firewall non si "compra", si progetta
(si comprano i suoi componenti)

- si tratta di trovare il compromesso ottimale ...
- ... tra sicurezza e funzionalità
- ... col minimo costo

L'indice della sicurezza



I TRE PRINCIPI INDEROGABILI DEI FIREWALL

- I. il FW deve essere l'unico punto di contatto della rete interna con quella esterna
- II. solo il traffico "autorizzato" può attraversare il FW
- III. il FW deve essere un sistema altamente sicuro esso stesso

D.Cheswick
S.Bellovin

Politiche di autorizzazione

“Tutto ciò che non è espressamente permesso, è vietato”

- maggior sicurezza
- più difficile da gestire

“Tutto ciò che non è espressamente vietato, è permesso”

- minor sicurezza (porte aperte)
- più facile da gestire

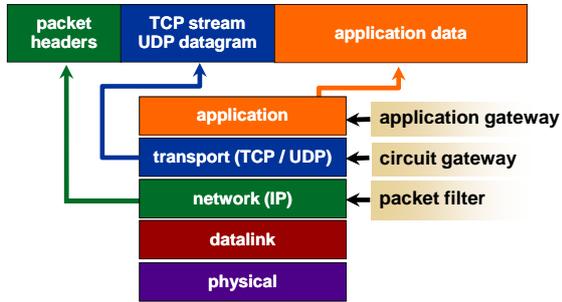
Considerazioni generali

- gli oggetti grossi sono più difficili da verificare
- se un processo non è stato attivato, i suoi banchi non ci riguardano
- “grande NON è bello” = configurazione minima
- un FW non è una macchina general-purpose (minimo del sw, no utenti)
- ognuno è colpevole finché non si dimostra innocente

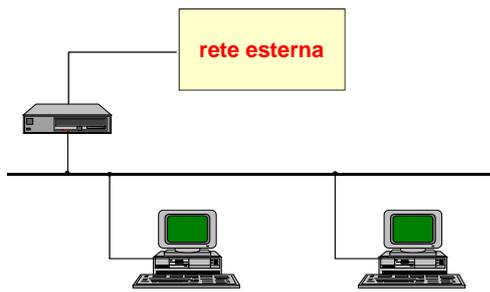
FW: elementi di base

- **screening router (choke)**
router che filtra il traffico a livello IP
- **bastion host**
sistema sicuro, con auditing
- **application gateway (proxy)**
servizio che svolge il lavoro per conto di un applicativo, con controllo di accesso
- **dual-homed gateway**
sistema con due connessioni di rete e routing disabilitato

A quale livello si fanno i controlli?



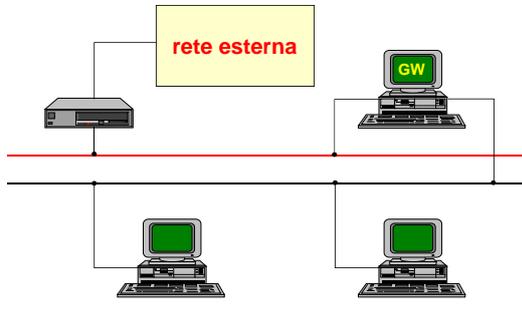
Architettura "screening router"



Architettura "screening router"

- usa il router per filtrare il traffico sia a livello IP che superiore
- non richiede hardware dedicato
- non necessita di proxy e quindi di modifiche agli applicativi
- facile, economico e ... insicuro!

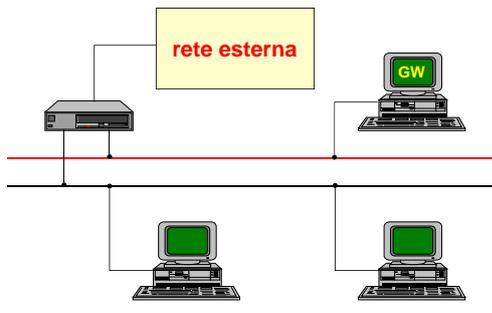
Architettura "dual-homed gateway"



Architettura "dual-homed gateway"

- facile da realizzare
- richiede poco hardware
- possibile mascherare la rete interna
- scarsamente flessibile
- grosso sovraccarico di lavoro

Architettura "screened host gateway"



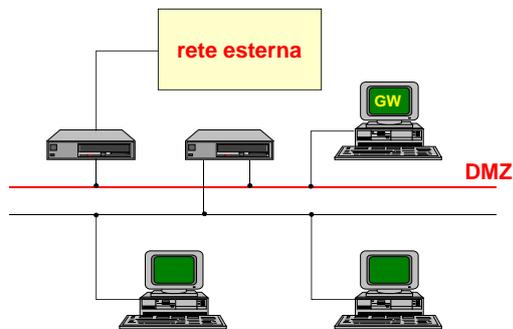
Architettura "screened host gateway"

- **router:**
 - blocca i pacchetti da INT a EXT a meno che arrivino dal bastion host
 - blocca i pacchetti da EXT a INT a meno che siano destinati al bastion host
 - eccezione: protocolli abilitati direttamente
- **bastion host:**
 - circuit/application level gateway per abilitare selettivamente dei servizi

Architettura "screened host gateway"

- più caro da realizzare
- più flessibilità
- complicato da gestire: due sistemi invece di uno
- si può selettivamente allentare il controllo su certi servizi / host
- si possono mascherare solo gli host/protocolli che passano dal bastion (a meno che il router abbia funzionalità NAT)

Architettura "screened subnet"

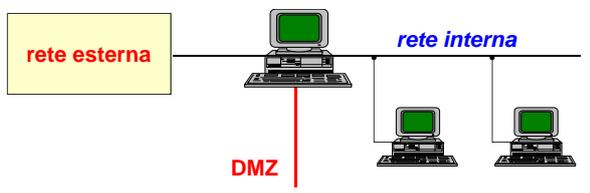


Architettura "screened subnet"

- DMZ (De-Militarized Zone)
- sulla DMZ - oltre al gateway - ci possono essere più host (tipicamente i server pubblici):
 - Web
 - accesso remoto
 - ...
- si può configurare il routing in modo che la rete interna sia sconosciuta
- soluzione costosa

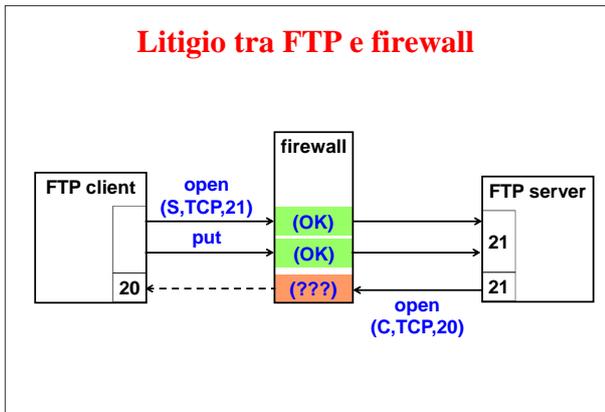
Architettura "screened subnet" (versione 2)

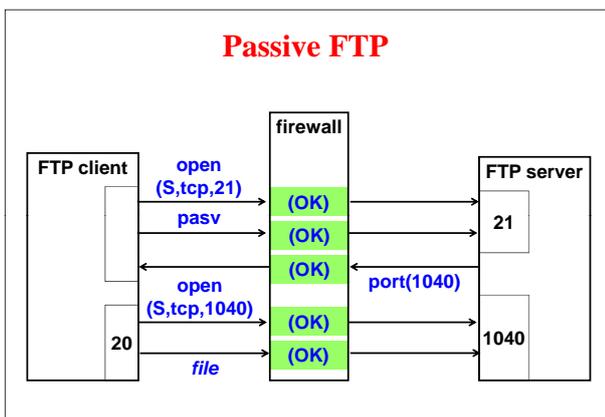
- per motivi di costo e di semplicità di gestione spesso si omettono i router (e le loro funzioni sono incorporate nel gateway)
- anche noto come "firewall a tre gambe"



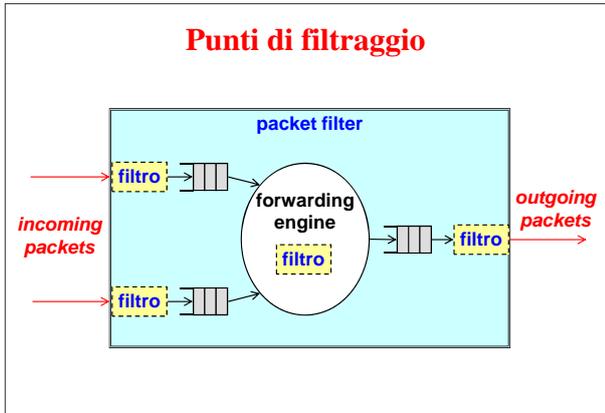
Filtri a livello rete (I)

- controllo degli indirizzi (ingress / egress filtering)
- disabilitare i servizi che si originano all'esterno
 - esempio: solo TELNET verso INET
 - esempio: solo HTTP verso web server su DMZ
 - problema: in FTP il trasferimento dati è iniziato dal server
- ICMP
 - è utile (ping, traceroute) quindi non disabilitarlo ma fare rate-limit
 - usato per denial-of-service
 - occhio a REDIRECT





- ### Filtri a livello rete (II)
- **UDP**
 - sono datagrammi, non circuito virtuale (quindi maggior carico per controllarli)
 - RPC usa porte a caso
 - meglio disabilitarlo tutto (tranne DNS)
 - **distinguere tra interfacce interne ed esterne**
 - **attenzione al numero di regole ed al loro ordine: possono cambiare drasticamente le prestazioni**



Filtri sui router: un esempio

- ipotesi: tutta la posta della rete 130.193 trattata solo da 130.193.2.1
- sintassi dei router CISCO:
 - access-list 100 permit tcp
0.0.0.0 255.255.255.255
130.193.2.1 0.0.0.0
eq 25
 - access-list 101 deny tcp
0.0.0.0 255.255.255.255
130.193.0.0 0.0.255.255
eq 25

Bastion host - configurazione

- ci devono girare solo i processi indispensabili
- deve fare il log di tutte le attività
- log in rete su un sistema sicuro all'interno
- disabilitare *source routing*
- disabilitare *IP forwarding*
- trappole per gli intrusi
(es. non usare mai 1s)

Tecnologia dei firewall

- **tecnologie diverse per controlli a vari livelli di rete:**
 - (static) packet filter
 - stateful (dynamic) packet filter
 - cutoff proxy
 - circuit-level gateway / proxy
 - application-level gateway / proxy
 - stateful inspection
- **differenze in termini di:**
 - prestazioni
 - protezione del S.O. del firewall
 - mantenimento o rottura del modello client-server

Packet filter

- **storicamente disponibile sui router**
- **effettua controlli sui singoli pacchetti IP**
 - IP header
 - transport header

Packet filter: pro e contro

- **indipendente dalle applicazioni**
 - ottima scalabilità
 - controlli poco precisi: più facile da "fregare" (es. IP spoofing, pacchetti frammentati)
- **ottime prestazioni**
- **basso costo (disponibile su router e molti SO)**
- **arduo supportare servizi con porte allocate dinamicamente (es. FTP)**
- **configurazione complessa**

Packet filter & FTP

- **due scelte possibili:**
 - lasciare aperte tutte le porte dinamiche (>1024)
 - chiudere tutte le porte dinamiche
- **difficile trade-off tra sicurezza e supporto a FTP!!**

Stateful (dynamic) packet filter

- **simile al packet filter ma "state-aware"**
 - informazioni di stato dal livello trasporto e/o da quello applicativo (es. comando PORT di FTP)
 - distingue le nuove connessioni da quelle già aperte
 - tabelle di stato per le connessioni aperte
 - pacchetti che corrispondono ad una riga della tabella sono accettati senza ulteriori controlli
- **prestazioni migliori rispetto a packet filter**
 - supporto per SMP
- **molte delle limitazioni proprie del packet filter**

Application-level gateway

- **composto da una serie di proxy che esaminano il contenuto dei pacchetti a livello applicativo**
- **spesso richiede modifica dell'applicativo client**
- **può opzionalmente effettuare il mascheramento / rinumerazione degli indirizzi IP interni**
- **nell'ambito dei firewall, normalmente ha anche funzioni di autenticazione**
- **massima sicurezza!! (es. contro buffer overflow dell'applicazione target)**

Application-level gateway (1)

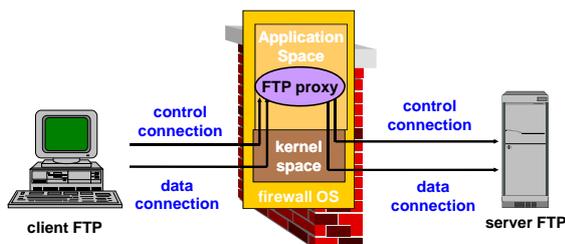
- regole più granulari e semplici rispetto a packet filter
- ogni applicazione richiede uno specifico proxy
 - ritardo nel supporto per nuove applicazioni
 - consumo risorse (molti processi)
 - basse prestazioni (processi user-mode)
- supporto SMP può migliorare prestazioni
- rompe completamente il modello client/server
 - server più protetti
 - può autenticare i client
 - mancanza di trasparenza per i client

Application-level gateway (2)

- può esporre il SO del firewall ad attacchi
- che fare in presenza di metodi di sicurezza a livello applicativo (es. SSL)?
- varianti:
 - transparent proxy
 - meno intrusivo per i client
 - strong application proxy
 - solo comandi/dati permessi sono trasmessi
 - è l'unica configurazione giusta per un serio proxy applicativo

Application-level gateway & FTP

- totale controllo della sessione applicativa



Circuit-level gateway

- è un proxy non “application-aware”
 - crea un circuito tra client e server a livello trasporto
 - ... ma non ha nessuna comprensione dei dati in transito

Circuit-level gateway

- rompe il modello client/server per la durata della connessione
 - server più protetti
 - isola da tutti gli attacchi che riguardano l'handshake TCP
 - isola da tutti gli attacchi che riguardano la frammentazione dei pacchetti IP
 - può autenticare i client
 - ma allora richiede modifiche alle applicazioni
- molte limitazioni proprie del packet filter rimangono

SOCKS

- è un proxy a livello trasporto (L4), ossia realizza un circuit-level gateway
- inventato dalla MIPS, v4 da NEC, v5 da IETF
- aka AFT (Authenticated Firewall Traversal)
- i client devono essere modificati:
 - standard: telnet, ftp, finger, whois
 - libreria per sviluppare propri client
- supporto anche commerciale:
 - nei browser (es. FX e IE)
 - nei firewall (es. IBM)

SOCKS RFCs

- RFC-1928 “SOCKS protocol V5”
- RFC-1929 “Username/password authentication for SOCKS V5”
- RFC-1961 “GSS-API authentication method for SOCKS V5”
- RFC-3089 “A SOCKS-based IPv6/IPv4 gateway mechanism”

SOCKS: funzionamento

- la libreria rimpiazza le funzioni standard per maneggiare i socket `connect()`, `bind()`, `accept()`, ...
- ... con funzioni che:
 - aprono un canale col SOCKS server
 - inviano `version`, `IP:port`, `user`
- il server SOCKS:
 - controlla la ACL
 - apre il canale richiesto (col proprio IP) e lo "congiunge" con quello interno

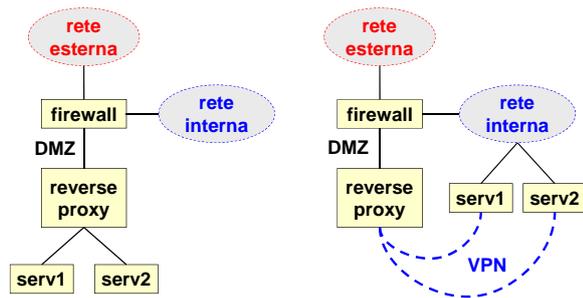
SOCKS: critiche iniziali

- SOCKS v4:
 - non distingue la rete interna da quella esterna
 - l'autenticazione degli utenti è molto debole (si basa su `identd` o configurazione locale del client)
 - supporta solo TCP
- soluzione = SOCKS v5:
 - supporta anche UDP
 - migliore autenticazione (user+pwd o GSS-API)
 - crittografia (tra client e server SOCKS)

Reverse proxy

- un server HTTP che fa solo da front-end e poi passa le richieste al vero server
- benefici:
 - obfuscation (non dichiara il vero tipo di server)
 - load balancer
 - acceleratore SSL (con back-end non protetto ...)
 - web accelerator (=cache di contenuti statici)
 - compressione
 - spoon feeding (riceve dal server tutta una pagina creata dinamicamente e la serve poco per volta al client, scaricando così il server applicativo)

Configurazioni di reverse proxy



Architetture di firewall: quale scegliere? (1)

- in teoria, più alto il livello a cui il firewall opera:
 - più alto sarà il consumo di cicli macchina
 - più alto sarà il livello di protezione che offre
- la realtà:

Firewall customers once had a vote, and voted in favor of transparency, performance and convenience instead of security; nobody should be surprised by the results. (Marcus J. Ranum, the "grandfather of firewalls", firewall wizard mailing list, oct 2000)

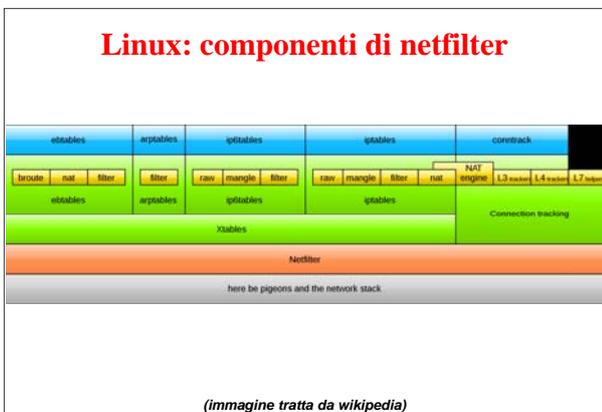
Architetture di firewall: quale scegliere? (2)

- **la scelta migliore:**
 - non un singolo prodotto, ma un'architettura di firewall robusta che supplisca alle carenze e eventuali vulnerabilità dei singoli dispositivi!!!
 - per il singolo elemento richiedere se possibile il supporto ad architetture multiple: meglio poter scegliere che lasciar scegliere ad un vendor!!
 - attenzione alle soluzioni che promettono di risolvere ogni vostro problema: forse si tratta di pubblicità ...

Firewall: prodotti commerciali

- tutti i maggiori produttori offrono un firewall
- tipicamente su UNIX, talvolta su Windows (ma in questo caso gli cambiano lo stack di rete!)
- esiste il Firewall Toolkit (FWTK)
 - gratis da TIS (www.tis.com)
 - mattoncini base di tipo application-gateway
- oppure IPchains / IPfilter / IPTables sotto Linux
 - packet-filter

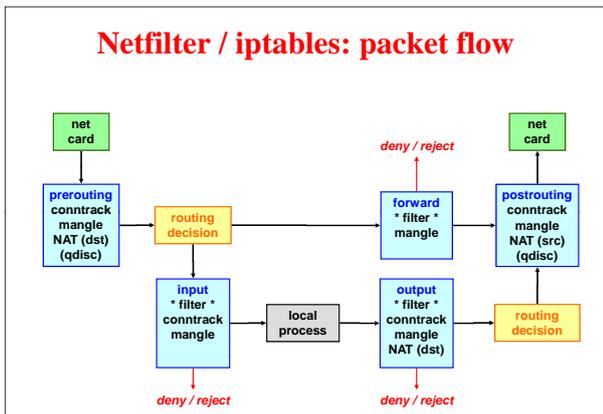
Linux: componenti di netfilter



Catene di default di netfilter

- **PREROUTING**
 - usata prima di una decisione di routing
- **INPUT**
 - pacchetti destinati al nodo stesso (la "local-delivery" routing table: "ip route show table local")
- **FORWARD**
 - per i pacchetti che hanno subito routing
- **OUTPUT**
 - pacchetti inviati da processi del nodo
- **POSTROUTING**
 - pacchetti pronti per l'invio in rete

Netfilter / iptables: packet flow



Stealth firewall

- firewall privo di un indirizzo di rete, così da non essere attaccabile direttamente
- intercetta i pacchetti fisicamente, mettendo la propria interfaccia di rete in modo promiscuo

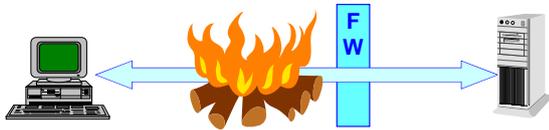


Local / personal firewall

- firewall installato direttamente sul nodo da difendere
- tipicamente un packet filter
- rispetto ad un normale firewall in rete può controllare i programmi a cui è permesso:
 - aprire collegamenti in rete verso altri nodi (ossia agire come client)
 - ricevere richieste di collegamento / servizio (ossia agire da server)
- importante per limitare la diffusione di malware o trojan, o semplici errori di installazione
- gestione firewall distinta da gestione sistemistica

Protezione offerta da un firewall

- i firewall sono efficaci al 100% solo relativamente agli attacchi sui canali che sono bloccati
- per gli altri canali occorrono altre difese:
 - VPN
 - firewall "semantici" / IDS
 - sicurezza applicativa



Intrusion Detection System (IDS)

- definizione:
 - sistema per identificare individui che usano un computer o una rete senza autorizzazione
 - esteso anche all'identificazione di utenti autorizzati, ma che violano i loro privilegi
- ipotesi:
 - il "pattern" di comportamento degli utenti non autorizzati si differenzia da quello degli utenti autorizzati

IDS: caratteristiche funzionali

- **IDS passivi:**
 - uso di checksum crittografiche (es. tripwire)
 - riconoscimento di pattern ("attack signature")
- **IDS attivi:**
 - "learning" = analisi statistica del funzionamento del sistema
 - "monitoring" = analisi attiva di traffico dati, sequenze, azioni
 - "reaction" = confronto con parametri statistici (reazione scatta al superamento di una soglia)

IDS: caratteristiche topologiche

- **HIDS (host-based IDS)**
 - analisi dei log (del S.O. o delle applicazioni)
 - attivazione di strumenti di monitoraggio interni al S.O.
- **NIDS (network-based IDS)**
 - attivazione di strumenti di monitoraggio del traffico di rete

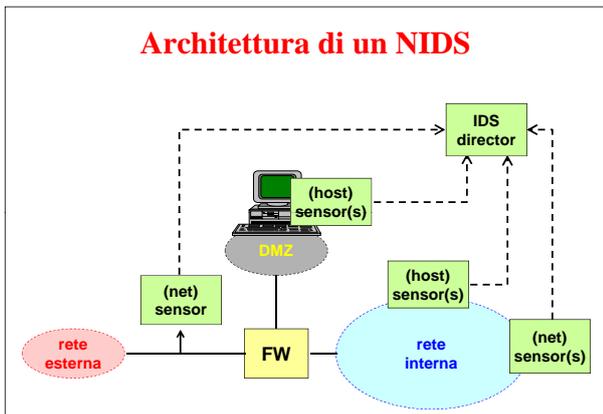
SIV e LFM

- **System Integrity Verifier**
 - controlla i file / filesystem di un nodo per rilevarne cambiamenti
 - es. rileva modifiche ai registri di Windows o alla configurazione di cron, cambio privilegi di un utente
 - es. tripwire
- **Log File Monitor**
 - controlla i file di log (S.O. e applicazioni)
 - rileva pattern conosciuti derivanti da attacchi o da tentativi di attacco
 - es. swatch

Componenti di un NIDS

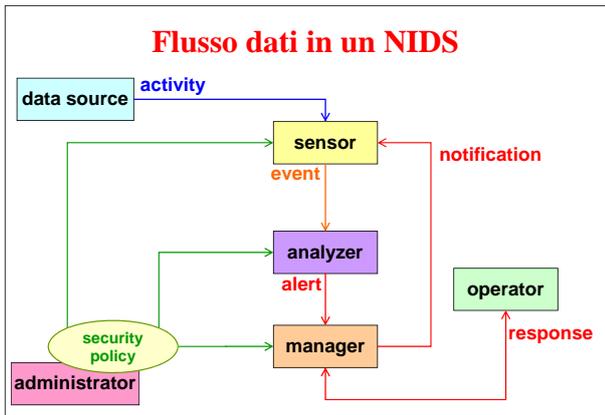
- **sensor**
 - controlla traffico e log individuando pattern sospetti
 - attiva i security event rilevanti
 - interagisce con il sistema (ACLs, TCP reset, ...)
- **director**
 - coordina i sensor
 - gestisce il security database
- **IDS message system**
 - consente la comunicazione sicura ed affidabile tra i componenti dell'IDS

Architettura di un NIDS



Interoperabilità di IDS/NIDS

- necessaria perché attacchi coinvolgono differenti organizzazioni e/o sono rilevate da diversi strumenti
- formato di signature:
 - nessuno standard, ma molto diffuso il formato di Snort
- formato degli allarmi e protocollo per la loro trasmissione:
 - IDMEF + IDXP + IODEF (IETF)
 - SDEE (Cisco, ISS, SourceFire)



- ### IDMEF + IDXP
- sviluppati da IETF
 - **Intrusion Detection Message Exchange Format**
 - indipendente dal protocollo (IPv4, IPv6)
 - supporta internazionalizzazione e localizzazione
 - supporta aggregazione e filtraggio dei dati (sul manager)
 - **Intrusion Detection eXchange Protocol**
 - basato su BEEP (RFC-3080)
 - scambia profili (di sicurezza end-to-end, di ID)
 - profilo di sicurezza base è TLS

- ### SDEE
- **Secure Device Event Exchange**
 - **basato sul paradigma webservice:**
 - messaggi in formato XML
 - scambio messaggi in HTTP o HTTPS
 - **standard (?) chiuso, gestito da ICSAlabs**

IODEF

- Incident Object Description and Exchange Format
- è un soprainsieme di IDMEF
- serve per scambiare informazioni tra enti diversi, tenere statistiche, valutare rischi, ...

IPS

- Intrusion Prevention System
- per velocizzare ed automatizzare la risposta alle intrusioni = IDS + firewall dinamico distribuito
- non un prodotto ma una tecnologia, con grosso impatto su tanti elementi del sistema di protezione
- pericolo di prendere la decisione sbagliata o di bloccare traffico innocuo

