UNIVERSITÀ DEGLI STUDI DI MILANO

Facoltà di Scienze Matematiche, Fisiche e Naturali

POLO DIDATTICO E DI RICERCA DI CREMA

SLIDE DI RIFERIMENTO PER :

SICUREZZA DEI SISTEMI E DELLE RETI

Mattia MONGA

anno accademico 2012 - 2013

SICUREZZA DELLE RETI E DEI SISTEMI NFORMATICI

Mattia Monga
https://mameli.docenti.di.unimi.it/sicureti

“l'informatica è una disciplina estremamente dinamica”

Obiettivi : interpretazione della letteratura scientifica

Parti del programma:

- TCP/IP sotto l'aspetto della sicurezza
“analisi del traffico (parte esame pratica)”

- Sicurezza perimetrale & rilevamento intrusioni
“parte tradizionale”

- Sicurezza nelle reti
“misure delle protezione importanti quando si usa una rete parzialmente non fidata”

Esame:

Parte scritta (domande ed esercizi) e parte di laboratorio (utilizzo dei tool trattati a lezione).
La prova è unica, durante lo stesso giorno, 1.30 h prima porva, pausa e 1h seconda prova.
75% del voto scritto, 25% prova laboratorio.

Appunti di MARCO BARATTA

N.B. I numeri tra parentesi indicano il numero della slide

1° lezione – 25/02/2013 (1)
Introduzione alla sicurezza delle reti

l'evento della diffusione del “worm”, correlato alla perdita di innocenza di internet, rese chiaro che i sistemi sono vulnerabili, ma che allo stesso tempo erano le reti ad essere violabili.In quel momento (2 Novembre 1988, rete in cui vi sono esclusivamente utenti competenti) la sicurezza o meglio la protezione delle rete, è divenuto un elemento fondamentale.
Il documento che analizzeremo è addirittura divenuto una RFC (1135). Il numero di macchine colpite si aggira attorno ai 500.

Nota. “Approfondire le teorie di Robert Tappan Morris sui sistemi operativi”

Il sopracitato Morris, è stato condannato nel 1986, da una legge principalmente percepita come una legge astratta, utile solo nel caso di casi militari. Ciò si è dimostrato falso, la sicurezza è un problema quindi generale, comune a tutti quanti, e non va percepito come un argomento astratto ma la contrario come qualcosa di concreto e applicabile / funzionale.

Capire i dettagli sui bug sfruttati è fondamentale per lo sviluppo e il miglioramenti, bisogna avere sempre un occhio attento sopratutto nel campo della sicurezza. La miglior tecnica di prevenzione e allo stesso tempo cura è la comunicazione tra i soggetti con qualsiasi infrastruttura, l'ambiente della sicurezza deve avere una rete per scambiare opinioni ed esperienze in modo da poter garantire la condivisione lo sviluppo.

-Il virus viene eseguito in un processo ospite.
-Il worm sarà un processo a se stante (quindi se un processo lo avvio io, non c'è il rischio che un worm entri nel mio sistema, a meno che il processo che lancio io non è un worm).
La differenza tra di due sta è rappresentata principalmente dalla diversa modalità di diffusione. I worm sono simili ai virus ma agiscono in modo indipendente senza bisogna di un processo ospite.
-Il trojan: è basato sull'idea di monitorare il sistema fino al momento in cui si trova un bug e in tal caso sfruttarlo.
-Rootkit, qualcosa che viene eseguito in kernel mode (garante che le nostre informazioni sono sicure) con i privilegi di root, in tal caso non si ha più la certezza che le informazioni nel sistema siano corrette.
-Keylogger (non richiedono particolare privilegi) e spyware, raccolgono info dei processi nel sistema (N.B. In allegato a questo argomento analizzeremo le OTP - One Time Password).
-Dialer moderni, presenti ad esempio negli smartphone (uniscono il mondo del TCP/IP al mondo telefonico, in sviluppo nel periodo più recente), ne sono esempio i messaggi premium (chiaramente involontari) simili a quelli di Theleton.

Le percentuali di segnalazioni vanno prese con le pinze, possono rappresentare tante cose diverse.
Può essere che vogliano far credere cose non vere, o molto imprecise prendendo in riferimento molti valori tra cui segnalazioni innocenti. E' importante verificare la complessità degli attacchi in quanto è stato constatato che circa il 96% sono degli attacchi possono essere evitati con dei semplici controlli preventivi e “runtime”.

La difesa è importante che sia proporzionata al valore delle risorse da proteggere.

La sicurezza informatica mette insieme il vasto mondo dell'informatica con le infrastrutture della sicurezza, è un ambiente in cui non esiste il bianco e il nero, le scelte vanno ponderate e sono (a volte) molto personali, sono scelte maturate con esperienza.

2° Lezione – 26/02/2013 (2)
Il modello di riferimento

Nello **Stack dei protocolli internet** (modello semplificato a 4 livelli) non consideriamo il livello fisico, il mezzo trasmissivo, partiamo subito dal livello "LINK", quindi i vari device driver che comunicano con il livello "NETWORK" ovvero il livello IP.
Il livello "TRANSPORT" che permette di stabilire (nella famiglia Tcp /Ip, connection-less "UDP", connection oriented "TCP") in modo molto generale la comunicazione, ed infine il livello "APPLICATION" ad esempio un protocollo applicativo come HTTP.

Il vantaggio di usare uno schema semplificato è che ciascuno dei livelli ha un mapping definito con le entità che pensiamo quando siamo abituati a pensare ai sistemi (il paradigma di confronto è UNIX, che è sempre stato aperto, o meglio conoscibile).
L'interfaccia di rete (eth) è una parte di silicio all'interno del PC (scheda di rete) funziona grazie grazie ai driver per tale interfaccia. Per farla entrare nel TCP/IP si deve assegnare un indirizzo IP al relativo driver che lo renderà riconoscibile. I vantaggi nel fare ciò sono molti come garantire comunicazione tra due noti in rete, ma sopratutto comunicazione fra due processi (in locale grazie al "pipe" ovvero strumenti per la comunicazione dei processi, in rete attraverso le "socket").

N.B. ifconfig = interface configuration

E' importante dividere lo stato del sistema operativo in "user mode" e "kernel mode". Fino al transport level il sistema è in kernel mode, ed è quindi in uno stato sicuro (Vedi schema di "stack dei protocolli internet"). Tale divisione è forzata dall'HW, non è possibile decidere preventivamente quale sarà lo stato. Vanno quindi suddivise le interfacce "ethernet" e "token ring", ciascuna con i propri protocolli di comunicazione che funzionano regolarmente a livello "FTP" e "TCP", e vanno intramezzati da un router per i "protocolli IP" e i "Driver".

Internet è una rete di nodi locali le quali sono collegati tramite router (o gateway).
Come si costruisce quindi internet?
Parliamo di:
-End to end principle: internet è una serie di tubi e l'intelligenza sta solo nei nodi, non nei tubi, quindi le parti di calcolo devono stare all'interno dei nodi. L'importanza di tale metodo è l'efficenza di trasmissione, non la sicurezza di tali trasmissioni. La rete è una struttura per massimizzare il flusso (tale principio è sempre più in discussione).

-Robustness approach: oltre all'efficenza della rete un altro punto ha come obbiettivo quello di rendere tale servizio sempre più robusto all'interno dell'infrastruttura della comunicazione e quindi, renderla capace di individuare e recuperare rapidamente da errori di percorso.

N.B. Nel caso in cui sia necessario distinguere i bit (es. differenza di un utente flat, caso di scarsa importanza, e utente a quantità di dati fissa, utente di alta importanza) in tal caso non sarà più sufficiente scaricare l'intera capacità intellettuale ai soli nodi ma bisognerà garantire un minimo di intelligenza a livello rete in modo da distinguere (relativamente all'esempio) i bit di maggiore importanza dai bit di minore importanza.
Relativamente alla grande diffusione di internet va dato un gran merito al principio che tale rete è di tipo “end to end”, quindi stupida e senza distinzione del traffico.

L'unica funzione centralizzata è quindi l'assegnazione dell'IP, una volta ottenuto sarò l'utente a gestire node, e quindi a stabilire il “cosa e come”.

N.B. Le RFC per relative alle comunicazioni stabiliscono la semantica del linguaggio di comunicazione tra i processi, in modo che i processi riescano a comunicare ma sopratutto a comprendersi.

La scheda di rete sono identificate da un seriale di 48 bit, di cui i primi 24 vengono assegnati al produttore (e indicano quindi chi è il produttore)
N.B. 2^24 → circa 16 milioni (che tenendo conto il tutto non sono nemmeno così tanti)

Vi è una misura massima di trasmissione (maximum transmission unit MTU) che è di 1500 byte, e allo stesso modo vi è una dimensione minima, ovvero la trasmissione deve essere almeno di 46 byte, quindi anche nel caso in cui volessi trasmettere due byte, trasmetto si quei 2 byte ma assieme ai restanti 44byte (vuoti, serie di 0).

Per ampliare la rete si utilizzano gli “hub”, e gli “switch”, i primi non sono altro che dei semplici ripetitori di segnale, mentre gli “switch”, la differenza è che un hub è una vasca da bagno, lo switch è più sofisticato perchè permette di definire più vasche da bagno (collision domain) e definire chi appartiene a tali domini (la separazione è solo logica, e si ottiene tramite la CAM table (content addressable memory) che contengono le associazioni MAC – porta dello switch).

Il MAC è un identificativo nato per permette di selezionare quali sono i messaggi per se, questo è lo scopo principale, anche se non l'unico, in via generale viene anche usato per identificare un dispositivo (numero seriale, di 48bit in notazione esadecimale di cui i primi 24 sono di default per ogni produttore, utilizzato come indirizzo all'interno della LAN). Ad esempio all'accesso della scheda di rete viene richiesto il numero MAC.
In una rete è possibile impostare specifici permessi solo per determinati MAC address.
Il MAC non va bene come identificatore, in quanto con i privilegi adeguati è possibile impersonare una macchina non presente all'interno della rete, facendo credere a tutti i restanti nodi della rete stessa che quella macchina esista veramente.
Un ulteriore problema (MAC flooding) lo si può incontrare quando, generando dinamicamente la tabella di corrispondenza MAC – PortaSwitch (comodo in quanto si posono suddividere le porte in vari vari collision domain, come vere e proprie sottoreti differenti) questa viene saturata. Questo attacco, una volta saturata la CAM table, costringe lo switch ad entrare in una condizione detta di 'fail open', in poche parole che lo fa comportare come un hub, inviando così gli stessi dati a tutti gli apparati ad esso collegati,

l'IP (v4) è formato da 32bit, ovvero 4 ottetti in base 256 (l'instradamento avviene tramite i Gateway, che dirigono i pacchetti verso due o più LAN). Gli IP non sono tutti uguali, vi

sono classi dalla A alla E e all'interno della A,B e C vi sono dei determinati range di indirizzi utilizzati per le reti intranet, quando un router riceve pacchetti da tali indirizzi attuerà delle politiche speciali, che possono essere: scartare, manipolare…
In un numero ip ci sono due informazioni, assieme all'IP viene fornita una netmask, che viene utilizzata per definire il range di appartenenza di un host all'interno di una sottorete IP al fine di facilitare la ricerca e il raggiungimento di un determinato host con relativo indirizzo IP.
rappresenta un insieme di 1 dove identifica il nodo e 0 dove identifica la rete.

La notazione usata per esprimere indirizzi CIDR (in un indirizzo IP permette di definire quale parte indichi la sotto rete e quale gli host) è la seguente: a.b.c.d/x, dove x è il numero di bit (contati partendo da sinistra) che compongono la parte di indirizzo della rete. I rimanenti y=(32-x) bit consentono di calcolare il numero di host della sottorete.

3° Lezione – 04/03/2013 (3)
Dal livello link a quello di trasporto

In una rete locale l'IP è un dato superfluo, è sufficiente il numero MAC (calcolabile tramite ARP dall'indirizzo IP stesso). Il procedimento di ottenimento è piuttosto semplice, ogni nodo mantiene una ARP cache (tabella di corrispondenze IP-MAC già note), e nel caso in cui comparisse nella rete un IP non presente in tale tabella si procede all'interrogazione della rete stessa secondo il seguente procedimento:
-chi ha l'indirizzo IP xxxx.xxxx.xxxx.xxxx?
-sono io : MAC.MAC.MAC
-assegnazione dell'indirizzo IP xxxx.xxxx.xxxx.xxxx all'indirizzo MAC.MAC.MAC
Problema : bisogna portare molta attenzione all'ARP poisoning, la quale consiste nell'inviare intenzionalmente e in modo forzato risposte ARP contenenti dati non corrispondenti a quelli reali. In questo modo la tabella ARP (*ARP cache*) di un host conterrà dati alterati. Molto spesso lo scopo di questo tipo di attacco è quello di ridirezionare i pacchetti destinati ad un host verso un altro al fine di leggere il contenuto di questi per catturare le password che in alcuni protocolli viaggiano in chiaro.
QUINDI ATTENZIONE, NON SEMPRE TUTTI I NODI DI UNA RETE LOCALE SONO FIDATI.
Soluzione : (parziale) utilizzo di tabelle ARP statiche.

N.B. L'ARP poisoning ha anche usi legittimi, come per esempio convergere il primo collegamento di una connessione verso un server di autenticazione.

Come ormai abbiamo già specificato, il livello trasporto è al di sotto del livello applicativo, quindi è chiaro che in una comunicazione a livello applicativo, il livello di trasporto deve identificare sia i nodi che stanno comunicando che i processi.
Per tale motivo introduciamo il discorso delle porte.
Il segmento che due processi si scambiano necessita di 4 numeri, ovvero “ip1,n1 : ip2,n2” dove ip1 e ip2 sono rispettivamente l'indirizzo del primo e del secondo processo, ed n1 e n2 sono le porte sulla quali sono raggiungibili i servizi (devono essre note al client, in quanto sarà sempre lui, il client, a dover richiedere tali servizi).
A tali porte vengono assegnati dei numeri convenzionali inferiori a 1024.

N.B. Un numero di porta non indica un preciso servizio ma la possibilità di stabilire una

connessione. Per esempio, vietare l'uso della porta 80 non pregiudica il fatto di non poter utilizzare il WWW ma semplicemente di impedire l'utilizzo da parte di client e server della porta 80, in quanto per qualche motivo hanno spostato il servizio WWW su una porta differente (attenzione tale limite è efficiente se imposta sul server, in quanto se impposto sulla rete il problema è raggirabile).

TCP
Il Transmission Control Protocol ha le seguenti caratteristiche:
-connection oriented: è necessario uno handshake preliminare per avviare una comunicazione
-full-duplex: supporta le comunicazioni in ambedue le direzioni.
-timer: I numeri di riscontro e i relativi timer di ritrasmissione permettono quindi di realizzare una consegna affidabile, ovvero di garantire che tutti i dati inviati siano comunque consegnati nel caso in cui qualche pacchetto possa essere perso nel transito attraverso la rete (controllo di errore in termini di riscontro di trasmissione).

Vediamo ora il flag appartenenti alle comunicazioni TCP:
SYN: richiesta di connessione (è sempre il primo pacchetto ad essere inviato)
FIN: intenzione da parte del mittente di voler terminare la sessione in maniera concordata
ACK: conferma della ricezione di uno dei due precedenti pacchetti (SIN o FIN)
RST: reset immediato della comunicazione
PSH: i dati in arrivo non devono venire bufferizzati ma passati subito ai livelli superiori dell'applicazione.

UDP
User Data Protocol, è il fratello di TCP, ma meno pesante del TCP. È una sorta di trasporto minimo, ovvero senza connessione e senza stato.
Ha le seguenti caratteristiche:
trasporto minimo : senza connessione, senza stato. Mentre in TCP qualsiasi informazione devo trasferire aggiungo 20 bit di informazioni, con UDP si usa solo 8 bit (12 in meno di TCP).

Sia TCP che UDP trasportano nel pacchetto Il checksum (funzionale nell'identificare errori casuali),non è nulla di più di un calcolo che permette di sapere lo stato di integrità del messaggio dopo la comunicazione :
- se i checksum dei messaggi sono diversi il messaggio è stato alterato
- se i checksum sono identici non si ha la certezza ma una buona probabilità che il messaggio sia corretto.

Passiamo ora ad analizzare i *problemi relativi a TCP/IP*:
-In TCP è difficile interrompere la comunicazione e perdere quindi il controllo, ma allo stesso tempo non c'è la protezione per la disponibilità di un determinato e specifico servizio (aspetto molto più ridotto rispetto al controllo)
-i controlli di integrità sono banali
-non vi è autenticazione fra le parti

Attenzione inoltre **all'IP spoofing** : tecnica tramite la quale si crea un pacchetto IP nel quale viene falsificato l'indirizzo IP del mittente. Le autenticazione tramite IP, sopratutto all'interno di reti locali, sono insicure e nel caso in cui in una rete vi siano indirizzi IP duplicati, può avvenire un DoS.

In qualsiasi caso, semplicemente spoofando l'IP, le risposte andranno comunque all'indirizzo legittimo (e non all'attaccante con l'indirizzo IP clonato) è quindi chiaro che non è una tecnica sufficiente per inserirsi interamente in una comunicazione IP, caso in cui bisogna servirsi dello handshaking completo.

Per far si che l'ISN (Initial Sequence Number) non sia facilmente riproducibile, alcune RFC vanno a specificare alcuni dettagli onde evitare confusioni e quindi connessioni duplicate, per la precisione:
-l'ISN va incrementato di 1 ogni 4 microsecondi
-Non può essere completamente casuale, ovvero bisogna utilizzare il contatore che si aggiorna ogni 4millisecondi sommato a una funzione hash crittografica.

In realtà i pacchetti del "sequence diagram" possono arrivare in ordine diverso da quello indicato, se si aspetta a sufficienza si raccolgono tutti i pacchetti e tramite i dati ISN (initial sequence number) si organizza lo stack dei dati con la sequenza corretta (rispettando appunto il numero di sequenza).
N.B. Guardare le RFC 793 e 1948, che ribadiscono e approfondiscono questi argomenti

Un'altro problema noto all'interno del sistema protocollare TCP/IP è senz'altro quello del **SYN flooding:** quando un host S riceve un richiesta SYN, tiene traccia per un certo periodo di tempo della connessione in una coda (di lunghezza finita), quando a tale SYN non segue un ACK la coda può riempirsi e portare al DoS.
Il sistema SYN cookies, è probabilmente l'unica difesa veramente efficace contro l'attacco SYN flooding. Questa difesa usa il sequence number del pacchetto SYN-ACK per mandare un cookie al client e quindi riconoscere se quel client ha già mandato dei SYN in quella sessione, senza dover memorizzare nessuna informazione sul server. In tal modo un singolo client non potrà inondare di SYN l' host S.
Stiamo parlando quindi di esame (non intrusivo) dei pacchetti di rete, ovvero il fingerprinting, il quale è in grado di identificare molti dettagli utili negli attacchi, come per esempio la topologia della rete (WindowsTTL=128, LinuxTTL=64).

4° Lezione – 05/03/2013 (4)
Scansioni

Monitoring della "cartografia di reti e servizi", quindi la conoscenza di canali di comunicazione disponibili, è fondamentale sia per gli attaccanti che per i difensori per progettare difese della rete.

Metodi di scansione – Strumenti di ricognizione
-Ping: Manda pacchetti ICMP (Internet Control Message Protocol) che richiedono una risposta, sono messaggi di controllo diagnostico. ICMP è un protocollo "over IP". Per evitare il Ping, spesso i messaggi ICMP vengono filtrati dai nodi e dai server per evitare sovraccarico. Esistono altri programmi per PING massiccio (non solo ICMP) come hping, nmap, fping.
-Traceroute: Essendo interessato alla topologia della rete, non voglio solo sapere se un determinato nodo è "vivo" ma anche quanti nodi attraversa per comunicare con un nodo in una rete, quanti "hop" (o cicli di vita) vengono eseguiti. Traceroute usa il TTL (Time To Live) per analizzare la topologia della rete, quando il TTL=0, si scarta il pacchetto e ci invia un messaggio ICMP per avvisare che il pacchetto ha concluso il suo ciclo. Si inizia

quindi inviando un TTL=1, e poi con TTL sempre crescenti fino a quando non si riceve una reply dalla destinazione finale, se con TTL=1 si riceve “TTL exceeded” allora il nodo sarà a 1 hop di distanza.
-Port scanning : l'obbiettivo è conoscere i possibili canali di comunicazione, quindi quali porte sono accessibili (TCP o UDP). Se sui canali non c'è nessuno attacco che sfrutti i protocolli TCP o UDP allora è impossibile (controllare le porte non vuol dire solo aprirle o chiuderle, poiché potrei comunque effettuare un “ping of death” oppure trasferire un eventuale malware su chiavetta sul nodo).

Vi sono vari stati un cui può trovarsi una porta.
-Open : la porta è aperta, ci si può connettere al servizio collegato.
-Closed : la porta è disponibile ma nessun processo ha effettuato la bind.
-Filtred : viene scartato il traffico filtrato (tramite firewalling o utilizzo di un router), appare come “Closed”.
N.B. Lo stato riconoscibile è OPEN, in quanto closed come filtred non sono distinguibili.
Uno degli strumenti più potenti conosciuti a livello globale è Nmap.

3 Way Handshake – TCP
- Un SYN a porta chiusa viene rifiutato, si risponde con RST (reset).
- Se filtrato, Il SYN arriva ma non c'è il RST (quindi riconosce che la porta è filtered).
- Se invio un SYN-ACK il protocollo risponde con RST
- Un RST viene ignorato.

UDP scan
UDP è privo di handshake (perchè è connection-less), tuttavia rispetto a TCP e più lento in quanto e basato su timeout. Scelta spesso scomoda ed inaffidabile in quanto lo stato viene segnalato tramite ICMP e spesso proprio ICMP viene filtrato dagli host server (es. possono passare solo x ICMP al minuto).

Le tecniche di scanning
- Connect
Il TCP connect () scan prende il nome dalla chiamata connect () che viene utilizzata dal sistema operativo per avviare una connessione TCP a un dispositivo remoto. A differenza della scansione TCP SYN (-sS), la scansione di TCP connect () utilizza una connessione TCP normale per determinare se una porta è disponibile. Questo metodo di scansione utilizza la stessa connessione handshake TCP di ogni altra applicazione basata su TCP utilizza sulla rete.

- SYN scan (half open)
La scansione TCP SYN utilizza metodi comuni di identificazione della porta che permettono ad nmap di raccogliere informazioni sulle porte aperte senza completare il processo di handshake TCP. Quando una porta aperta è identificata, l'handshake TCP viene resettato prima che venga completata. Questa tecnica è spesso definito come "half open" scanning.
Presenti due vantaggi, ovvero: più veloce perchè ho un passaggio in meno, e siccome l'handshake non è finito non viene loggata la connessione, perchè non è stata di fatto stabilita nessuna connessione. Spesso richiede comunque i privilegi di root.

-TCP NULL - FIN - xMAS scan
Viene eseguita la scansione delle porte chiuse con l'invio di alcuni tipi di pacchetti

appositamente modificati (asincroni o non consentiti) e rilevato lo stato delle porte tramite pacchetti RST (nel caso in cui non ci sono regole di filtraggio) o ICMP / null (nel caso in cui le porte siano filtered).

-IDLE SCAN
L'attaccante interroga lo zombie per verificarne l'inattività e per sapere qual è il valore che sta usando per il campo identification. L'attaccante invia poi un pacchetto alla porta della vittima che intende sondare, specificando però un IP sorgente pari a quello dello zombie (tramite ip spoofing) che deve essere idle. Il risultato ottenuto può essere uno dei seguenti:

-la vittima ha la porta aperta: in questo caso la vittima reagisce inviando allo zombie un pacchetto con i flag SYN/ACK. Lo zombie lo riceve, ma trattandosi di un pacchetto fuori sequenza, e quindi inatteso, esso risponde alla vittima trasmettendole un pacchetto con il flag RST.

-la vittima ha la porta chiusa: in questo caso la vittima reagisce trasmettendo allo zombie un pacchetto ICMP di tipo 'Destination Unreachable'. Lo zombie lo riceve, ma non fa nulla perché si tratta di una risposta inattesa ad una richiesta di connessione che esso non aveva inviato.

-la vittima scarta il traffico in ingresso sulla porta (ad esempio tramite un firewall): il pacchetto viene ignorato, e non vi sono risposte ICMP verso lo zombie.

A questo punto l'attaccante interroga di nuovo lo zombie e può osservare uno di questi due comportamenti:

-il valore di identification dello zombie è variato, quindi deduce che la porta della vittima era aperta.

-il valore di identification dello zombie non è variato, e quindi deduce che la porta della vittima era chiusa oppure filtrata.

La tecnica è piuttosto imprecisa e richiede che ci sia un host zombie totalmente inattivo (idle), ma ha il vantaggio di essere completamente anonima alla vittima, impedendo quindi qualsiasi contromisura e facendo scattare un allarme in un eventuale IDS che però indica l'indirizzo dell'idle host.

5° Lezione – 05/03/2013 (4)
IPsec

IPsec
La suite TCP/IP non è stata progettata con particolari metodi di difesa e sicurezza (creata per uso accademico). IPsec è un protocollo che specifica come cifrare autenticare e scambiare chiavi con IP (diverso tra IPv4, dove trova un supporto facoltativa, e IPv6 dove invece è obbligatorio).

Vi sono vantaggi nell'utilizzo di IPsec, CIA
-Confidenzialità : autentcazione dell'origine dei dati (NO spoofing)

-Integrità : protezione da Reply attack (difficili da evitare senza meccanismi appositi es. time-out)
-Autenticazione (N.B. Non availability) : Controllo dell'accesso della comunicazione (controllo di chi comunica).

IPsec in realtà è composto da due protocolli:

-Autentication Header (AH)
si occupa di autenticazione e integrità, identifica replay di pacchetti con una tecnica "sliding window" * e un contatore che per esser inizializzato necessita una SA.
Un "security parameter index" identifica la SA**.

*La finestra scorrevole rappresenta il numero di byte che il destinatario della trasmissione si dichiara disposto a ricevere dal mittente oltre l'ultimo byte per il quale il mittente abbia già ricevuto il segnale di conferma ACK.

-Encapsulating Security Payload (ESP) –
si occupa di confidenzialità serve a cifrare il contenuto dei pacchetti, può essere implementato tramite due modalità:
transport: protocolli superiori cifrati end to end
tunnel: i pacchetti IPsec contengo i pacchetti IP (cifrandoli)
Un "security parameter index" identifica la SA**.

** Una Security Association (SA), che si occupa di creare un associazione "sicurista", è la creazione di attributi di sicurezza condivisi tra due entità di rete per supportare la comunicazione sicura di credenziali, in pratica una sorta di protocollo di certificati all'interno della crittografia.

Come vi sono dei vantaggi, sono presenti anche dei **problemi relativi a IPsec**:
- La configurazione dei firewall per permettere l'esecuzione di IPsec non è banale perchè sono protocolli che hanno pacchetti con campi strani, a volte cifrati a volte no, è difficile distinguere i vari casi.
- La complessità da gestire ha un costo per la comunicazione, come adattare misure speciali di security association (Overhead amministrativo e computazionale).

6° Lezione – 11/03/2013 (5)
TLS / SSL

TLS (Transport Layer Security) o **SSL** (Secure Socket Layer) vengono usati per lo stesso concetto, vengo usati sia uno che l'altro, iniziamo a toccare il livello di TRASPORTO (il vantaggio risiede nel fatto che non va riconfigurato il livello di rete, ma solo il livello di trasporto, che comunica direttamente con il livello applicativo).
Nasce negli anni 90 con l'introduzione dei browser (proposta creata con Netscape), qualche anno dopo viene inserito qualcosa che permetta livelli di sicurezza, che il browser appoggiato a TCP non avrebbe.
Vengono utilizzati per fornire sicurezza e garanzia che gli utenti che navigano si stiamo collegando a server certificati.
Il browser diventa l'interfaccia prevalente e fondamentale sia nella sicurezza sia nelle strategie commerciali degli attori di internet, ovvero come negli anni 80 (quando

l'innovazione di internet permetteva che io mi inventassi una nuova applicazione con un protocollo applicativo e iniziassi a distribuirla), ai giorni d'oggi esiste ancora la condivisione (vedi client bit torrent) ma prende sempre più piede il fatto che le info per passare sul web devono passare dal browser che viene visto come il canale principale.
N.B. I browser hanno una complessità paragonabile a quella di un SO, e la loro comprensione è rilevante nel mondo della sicurezza.
Con il passare degli anni SSL 3.0 viene standardizzato da IETF come TLS.

TLS ha degli *obbiettivi*:

-cifratura end to end, ovvero faccio una cifratura che impedisce la comprensione del messaggio al di fuori di mittente e destinatario, ovvero browser e web-server, (nemmeno i router riescono a comprendere i messaggi, ovvero non si percepisce che c'è crittografia, ma si vede semplicemente una sequenza di caratteri) e quindi protezione dell'integrità (evitare Man in the Middle).
P.S. La crittografia viene considerata arma strategica, quindi non esportabile fuori dagli USA, e quindi venivano creati due versioni differenti degli stessi software, la restrizione è passata poi ai soli software con chiave maggiore a 48 bit per infine infine divenire disponibile a tutti tranne che ai nemici.

-Autenticazione, viene permesso di autenticare il server (e quindi client anonimo) tramite SSL (non è un obiettivo principale). Basti pensare che l'unico briciolo di sicurezza che abbiamo sul web è l'indirizzo, ovvero il dominio alla quale ci stiamo per connettere, ma attenzione alla compromissione dei DNS (possibilità di dominio scritto in UTF-8 ovvero molti caratteri simili tra loro, o DNS poisoning).

-doveva esser adeguato all'uso applicativo che se ne voleva fare (HTTP) che in principio era dedicato a connessioni brevi e stateless (ovvero corte e senza memoria delle connessioni precedenti).

La soddisfazione di tutti questi obbiettivi pone davanti a una scelta, o creare una sessione per ogni connessione (può creare overhead, a causa dell'onerosità di ogni richiesta) oppure, scelta più adeguata, lo stato della sessione viene suddiviso per più connessioni.
L'handshake serve a costruire la sessione, scambiandosi il TLS record layer (pezzettini aggiuntivi per creare la sessione, che autentica il server e non il client (quest'ultimo viene garantito da un ulteriore autorità)).

P.S. La dimostrazione dei protocolli di crittografia viene testata utilizzando dei teoremi matematici, in pratica si cerca un'approssimazione della relativa complessità che però con il tempo subisce mutamenti frequenti, ergo, non fidarsi troppo e quindi non esporre / indicare il protocollo utilizzato.

TLS handshake
-Il client elenca le “chiper suite” (CS) ovvero quale meccanismo crittografico conosce.
-Il server sceglie una CS tra quelle indicate dal client e spedisce un “Digital Certificate” (DC) firmato da una “Certification Authority” (CA).
-Il client controlla il certificato digitale ricevuto, cifra ed invia una chiave di sessione random “k”.

N.B. richiamo a “How secure is HTTPS”

TLS può esser utilizzato oltre che con il browser, con altre applicazioni (come abbiamo detto risiede a livello di trasporto). Vi sono tre metodi di implementazione:
-Creo una nuova applicazione basata su TLS (es.SSH2)
-Aggiungere TLS a un servizio noto (es. HTTP con l'aggiunta di SSL, ovvero HTTPS), in pratica si aggiunge un controllo TLS ad ogni comando del precedente protocollo (es. GET over TLS).
-Aggiungere ulteriori comandi con la possibilità che tali comandi possano esser utilizzati con TLS (es. ESMTP)

N.B. richiamo a “shmat_ccs12.pdf”
abbiamo visto che:
-Evidentemente perfino ditte che producono servizi dove la sicurezza è molto importante fanno poco in ambito di “adversarial testing”, ovvero test contraddittori.
-La maggior parte delle librerie SSL sono “unsafe by default”, ovvero insicure di default (contrario del principio).
-Se non verifico il protocollo rischio di ricadere in “misuse”, ovvero uso improprio.
-La sicurezza ha un costo, e non tutti vogliono spendere sia tempo che materiale. In molti casi sviluppatori disabilitano i certificati.

IL problema rilevante, nell'uso della sicurezza SSL o TLS, risiede nelle prestazioni (un sito in HTTPS risulta circa 82 volte più lento dell'equivalente senza HTTP) inoltre, tale connessione va gestita per ogni richiesta di ogni client.
Per risolvere tale problema si sta ideando “tcpcrypt” (*) che arriverà ad essere 3 volte più lento di tcp (nel lontano 2020), rispetto alle 82 attuali.
Nonostante la connessione sia falsa, l'overhead persiste, ovvero la connessione tra me e il server non validato rimane comunque over TLS, quindi sicura (cifratura inutile).

(*)Un'ulteriore proposta di riduzione dell'overhead prevede di spostare l'onere computazionale della crittografia sulla macchina client, in modo da scaricare il server. Si tratta anche qui (simile al TCP handshake) di “opportunistic encryption”, si chiede al client se supporta tale procedimento e nel caso in cui la risposta sia positiva si procede. (N.B. vedi slide 120 per schema).

TCPcrypt HandShake
viene creato un session ID (SSID) che la caratteristica di esser “probabilisticamente unico”, ovvero non è garantito che non sia già stato generato, ma l'algoritmo prevede una probabilità di generare un doppione molto bassa. Tale SSID non dice che sto comunicando con ad esempio “Mattia Monga” ma da meno informazioni, non dice con chi comunico (di questo se ne occupano ulteriori autorità), garantisce che la comunicazione è protetta. Può esser utilizzato per la creazioni di protocolli più complicati, identifica quindi una sessione, e ciascuna sessione avrà un SSID “pseudo unico”.
Il messaggio non può esser aperto senza chiave, e l'attaccante non può riutilizzare il messaggio in un altra connessione in quanto l'SSID sarà diverso (N.B. Vedi slide 121 per schema).

N.B. richiamo a “tcpcrypt.pdf”
L'unica cosa che fornisce il protocollo è l'SSID, che può esser utilizzato in modi diversi per scopi diversi, uno dei principali (ma non sempre necessario) è l'abilitazione all'utilizzo del certificato, il server manda al client la chiave di sessione, il certificato, e l'SSID firmato

con la chiave privata (vedi pag. 7 Cap. 4.1).

7° Lezione – 18/03/2013 (6)
I confini di una rete

Dal 12 marzo 2013 si sconsiglia altamente l'utilizzo di RC4 (cifratura a flussi) in TLS.
L'argomento di oggi sarà la difesa perimetrale, ovvero definire quali sono i confini di una rete locale. Nella internet originaria (rete di reti locali) la rete era divisa in due parti, ovvero rete locale e rete non locale (ovvero tutta la rete che non è locale). La differenza tra le due viene identificata dalla netmask (che distingue solo numeri IP), ma con il passare del tempo si è arrivati a pensare che questa fosse un definizione un po' troppo grossolana.
La differenza è importante in quanto, secondo la regola generale: ciò che è locale è "trust". Ho bisogno quindi di stabilire cosa è locale (attenti a non esagerare con i confini, in quanto si potrebbe comprendere ciò che non lo è veramente). Tale regola, ai giorni d'oggi non è più sempre vera in quanto il problema è che molto spesso, se non si è una situazione "safe" in locale, il firewall serve a ben poco.

N.B. Non si parla di confine ma di perimetro

Il firewall (muro anti fuoco – taglia fiamme) vede tutto il traffico tra due reti, e il compito principle è quello di dividere ciò che è locale da ciò che non è locale (es. rete A – locale, e rete B – non locale), l'idea non è quella di fermare il "fuoco" (traffico non permesso) ma semplicemente di mantenerlo al di fuori della rete locale.
Come abbiamo detto, vede tutto il traffico, funzione essenziale in quanto deve poter scegliere cosa bloccare, cosa lasciar passare (deve filtrare il traffico secondo una precisa politica d'accesso, la "policy"). Al contrario di quel che si possa pensare il firewall non deve controllare che il traffico permesso non faccia danni (intrusion detection, tramite il "control", ovvero scoprire qualcosa che ha già passato il confine e che vado a definire "traffico illecito"). Il difficile risiede proprio nello stabilire il limite corretto tra traffico lecito e non.

N.B. *L'obiettivo dei firewall* (bloccare traffico illecito) è irrealistico, perchè qualcosa riuscirà sempre a raggirare il sistema di firewalling e passare "lecitamente" in rete locale.

I firewall, tipicamente vengono realizzati come:
-Forwarding gateway: macchina che quando trasferisce traffico lo fa solo se questo rispetta una certa policy (tra due reti A e B)
-Filter Routing: stessa funzione del forwarding gateway ma più in generale. Viene messa su un router, in pratica si differenzia solo dal livello di complessità (può supportare quindi più reti).
-Proxy: Fa finta di essere qualcos'altro per raggiungere un certo obiettivo. Nell'ambito delle reti è qualcosa di locale, ma nasconde qualcosa di remoto. Sta in mezzo tra locale e remoto e fa vedere una versione semplificata del tutto, come se fosse solo in locale.

È importante differenziare i vari tipi di firewall a seconda del livello in cui agiscono:
-Applicativo (application gateway - proxy)
-Trasporto (circuit gateway)
-Rete (packet filter)
-Ibrido (funzionano su più livelli, come ad esempio Dynamic Packet Filter, che agisce sia a

livello rete che trasporto e a volte anche applicativo.

Tecniche di filtraggio del traffico (tecniche firewalling):

Stateless filtering
Filtraggio che, in ogni preciso momento (senza history o memoria) stabilisce cosa bloccare e non. Quindi si guarda e si filtra ogni singolo elemento del protocollo applicativo (es. in HTTP non voglio che passino comandi POST).
La realizzazione avviene tramite la scrittura di una ACL (Access control list) che filtra, come già detto, ogni singolo elemento (nota bene la differenza tra "porta interna" e "porta esterna". Gli equivalenti degli "indirizzi interni" ed "indirizzi esterni" in IPtables).
Le azioni possibili sono :
-Reject (rifiuta e segnala che non si vuole ricevere quel determinato pacchetto)
-Drop (rifiuta senza dire niente)
-Accept (accetta il pacchetto)
-Log (accetta ma segna i log di tale pacchetto)

Se un determinato tipo di pacchetto che non è descritto nelle "policy" va stabilita una semantica, la scelta è tra:
-Default deny (vieto tutto ciò che non è esplicitamente permesso, quindi il pacchetto non passa. Dalla parte del mondo delle reti risulta conveniente in modo da evitare incidenti. È più sicuro anche se vi sono restrizioni per gli utenti).
-Default permit (se non indicato diversamente è permesso, quindi il pacchetto passa, corretto secondo un punto di vista giuridico).

Stateful filtering
Filtraggio che, scrive una policy che tiene conto di ciò che è già passato e ciò che non è ancora passato; Si ha traccia dello stato del sistema e il filtraggio avviene sulle storie dei pacchetti o delle richieste. In questo caso serve mantenere una tabella delle connessioni (es.in HTTP non voglio che passino più di 3 comandi POST, quindi quando arriva POST li metto in attesa in uno spazio apposito, se alla fine della comunicazione sono meno di 3 allora le richieste passano, viceversa vengono bloccate).

N.B. Le connessioni stateful oltre ad essere costose sono sconsigliate, se non per particolari casi.

I firewall stateful che operano filtraggio applicativo analizzando il contenuto dei pacchetti vengono definiti "Deep Packet Inspection" i quali sono caratterizzati da due caratteristiche quali analisi del traffico applicativo la cui liceità va valutata caso per caso, e generalmente basati su pattern matching di stringhe. Da non dimenticare che utilizzando Deep Packet Filtering si avranno sia problemi tecnici (scarsa, scarsissima efficienza) e problemi dal punto di vista giuridico, ovvero si è in bilico tra la semplice analisi del traffico e l'abuso del potere andando a ledere la privacy e il trattamento dei dati personali (ricorda di leggere le informative, orco cane).

Vediamo ora le *configurazioni ricorrenti*
DMZ: Demilitarized zone (segmento isolato di LAN raggiungibile sia da reti interne sia esterne ma caratterizzata dal fatto che gli host attestati sulla DMZ hanno possibilità limitate di connessione verso host specifici della rete interna)
SHBH: single-homed bastion host* (Nel caso in cui il firewall venga compromesso la rete

interna rimane isolata dal bastian host; vi è una sola scheda di rete)
DHBH: double-homed bastion host* (vi sono due schede di rete fisiche, risulta più costosa ma leggermente più efficace, in quanto in questo caso si hanno due sottoreti, una leggermente più protetta rispetto all'altra più esterna).
P.S. Per una completa e corretta comprensione dell'argomento visitare slide 149 – 150.

**bastion host:* non vi sono servizi sopra, ma è una macchina molto protetta che spendiamo appunto in protezione, l'attaccante è costretto ad attaccare quella macchina che si interpone tra noi (e quindi i nostri dati) e la rete esterna, serve da diversivo (viene utilizzato come ostacolo che rallenta gli attacchi) grazie al quale, mentre viene attaccato, abbiamo il tempo per staccare i nostri dispositivi della rete in modo da renderli irraggiungibili da remoto.
Quindi nel caso in cui il firewall venga compromesso la rete interna rimane isolata dagli attacchi esterni grazie al bastion host
Il passo processo di sicurezza si completata con il DMZ, dove oltre a tutto quanto detto, viene inserito un ulteriore firewall dopo il bastion host.

N.B. È possibile inserire vari DMZ in modo da avere un controllo granulare sulle tipologie di trust distribuiti nel sistema, quindi inserire più firewall e ottenere "regioni" di rete con diverso grado di sicurezza.

P.S. La lezione conclude con degli esempi di filtering sulle slide.

8° Lezione – 19/03/2013 (7)
Complessità del filtering

Principio least privilege (LPP), ovvero ciascuno attore dispone del minimo dei privilegi necessari per raggiungere gli obbiettivi assegnatigli dalle specifiche del sistema.
E' fondamentale capire il perchè di tale scelta: nel caso in cui si verificasse un accesso abusivo con l'account di un attore autorizzato al sistema, si avrebbero dei privilegi non sufficienti (si spera) per effettuare operazioni dannose. Il problema risiede come al solito nell'applicazione di tale regola, ovvero bisogna trovare un compromesso tra flessibilità e sicurezza del sistema.

Vediamo ora cosa rende l'operazione di firewalling più complicata, vi sono protocolli (come SSH, nati dopo l'introduzione dei firewall) il quale ruolo del client e del server sono fissi nel tempo (client sempre cliente e server sempre server) e la comunicazione è molto semplice, si tratta di request/reply. Tali protocolli vengono riconosciuti come firewall-friendly, in quanto la configurazione del firewall risulta davvero semplice.

-SMTP: In ogni rete aziendale un solo server di questo tipo (SMTP) è autorizzato alla gestione della posta elettronica con l'esterno. È un protocollo semplice (firewall-friendly), il ruolo client / server è abbastanza ben definito. Porta di default 25, a meno che il firewall e il protocollo non si mettano d'accordo in modo diverso.
P.S. Vedi slide 166 per esempi (confronto con quanto fatto con SSH).
N.B. L'SMTP server non è solo un server (riceve le richieste di altre send-mail in giro), è anche un client per gli altri SMTP server (ad esempio se voglio connettermi al mio account Gmail, per inviare o scambiare messaggi).
P.S. Slide 168, la configurazione funziona ma non rispetta il principio del minimo privilegio

in quanto qualsiasi porta può sia inviare che ricevere, non soltanto la 25 dell'SMTP. Una possibile soluzione viene presentata nella slide 169, dove viene ristabilita la correttezza inserendo limiti sulla porta 25.

L'obbiettivo del sicurista è il LEAST-PRVILEGE, non sempre di semplice implementazione (bisogna conoscere i protocolli).

-FTP: (slide 174) Non è un protocollo firewall-friendly. L'FTP server non è conoscibile a priori,quindi difficilmente riuscirò a metterlo nel firewall. Di recente viene utilizzato sempre più di frequente FTP passivo (comando pasv), nella quale viene richiesto al server FTP di entrare in modalità passiva, il server indica se è possibilie e nel caso lo fosse su quale porta, dopo di chè si farà il TCP handshake sulla porta indicata, vi sarà lo scambio di comandi e ACK (sulla porta 21, per scambio di comandi) e successivamente il download / upload e l'ACK sulla porta precedentemente selezionata dal server FTP.

-RPC : (slide 176) è un protocollo molto complesso (alcune delle cose di P2P funzionano tramite questo protocollo). Questo protocollo è costruito con il principio di non fissare permanentemente quali sono le operazioni consentite, ma far si che queste possano variare, e fornire una sorta di manuale sulla quale sono indicati tutti questi servizi (protocollo di servizi dinamici). L'infrastruttura è fatta in modo che io richiedo se un dato servizio è disponibile (tramite una directory sempre accessibile), in quel momento mi risponderà se il servizio è disponibile, e in tal caso su quale porta è raggiungibile. (es. c'è un protocollo SMTP disponibile? No / si è raggiungibile tramite la porta 5050). si crea quindi un infrastruttura che si auto adatta, contraria al principio della sicurezza, dove tutto è prestabilito e monotonamente ripetitivo (regole precise) nella quale le porte vengono assegnate dinamicamente (per ciascun servizio) e non si conosce quindi a priori a quale porta sarà raggiungibile ogni processo. Ne è esempio *PortMapper* in Unix.

Spenderemo ora qualche parola per il firewalling a livello applicativo, parliamo quindi di **PROXY** (che funge da intermediario, sia client che server), con questa soluzione non esiste più il server che risponde alle nostre richieste, ma vi è il proxy che media quindi la comunicazione tra client e server, viene disaccoppiata la comunicazione tra deu componenti.
P.S. Slide 180 per spiegazione grafica.
L'idea è che il proxy può decidere (tramite white / black list) quale pagine siano accessibili all'esterno e quali no.
Analizzeremo diversi tipi di proxy,
-anonymizing proxy: servono per anonimizzare le connessioni nel web.
-web proxy: vengono salvate le cache delle pagine visitate in modo da aumentare le prestazioni dei sistemi collegati al proxy (dove vengono appunto salvate le pagine).
-reverse proxy: Gestiscono accesso da utenti esterni a risorse interne alla rete, praticamente il vero accesso alle risorse lo fa il proxy, noi (client) ci colleghiamo al proxy, non alla risorsa (basato su un'idea simile a quella dei bastian host). P.S. Slide 182
-proxy firewall: Può esser utilizzato per analizzare i dati delle applicazioni in quanto opera a livello applicativo (simile a un firewall statefull ma appunto a livello applicativo), le performance sono davvero scarse.

L'FTPbounce è un exploit del protocollo FTP con cui un aggressore è in grado di usare il comando PORT per richiedere l'accesso alle porte indirettamente attraverso l'uso del computer della vittima come un inermediario per la richiesta.

NAT (Network address translation)
La NAT è una tecnica che consiste nel modificare gli indirizzi IP dei pacchetti in transito su un sistema che agisce da router all'interno di una comunicazione tra due o più host (di solito vengono utilizzati i range di indirizzi riservati, Vedi IPv4 pagina 4) in modo da mascherare gli indirizzi effettivamente utilizzati all'interno della rete.
L'assegnazione dell'IP può essere.
-statica (IP interni mappati staticamente in IP pubblici)
-dinamica (l'associazione tra IP interno e IP pubblico avviene a run-time)
N.B. Slide 190 per riferimento immagine DynamicNAT

IPmasquerading
E' un caso particolare di source NAT, in cui le connessioni generate da un insieme di computer vengono "presentate" verso l'esterno con un solo indirizzo IP. La tecnica è detta anche *Port Address translation* (PAT), in quanto vengono modificati non solo gli indirizzi IP ma anche le porte TCP e UDP delle connessioni in transito. Questo metodo prevede di individuare una rete "interna" (che tipicamente utilizza indirizzi IP privati) ed una "esterna" (che tipicamente utilizza indirizzi IP pubblici), e permette di gestire solo connessioni che siano originate da host della rete "interna".

IDS
Gli Intrusion Detection System (IDS) sono dispositivi HW e SW (a volte la combinazinoe di entrambi, sotto forma di sistemi stand-alone) che generano allarmi, utilizzati per l'identificazione degli accessi non autorizzati a dispositivi e alla rete locale.
Vi sono tre step nel loro funzionamento:
-Raccolta dei dati: quindi verifica dei log di sistema o di specifici applicativi per individuare eventuali anomalie
-Analisi dei dati: controllo dell'integrità dei fIe locali, come ad esempio modifiche sospette
-Generazione degli allarmi: reagire a pattern di attacco noti e a tentavi di intrusione.
I vantaggi, sono insiti negli step, ovvero sono utili per prevenire eventi indesiderati, per avere un meccanismo di segnalazione che permetta di attivare determinate procedure, ed infine per avere una conoscenza statistica sull'utilizzo della proprio infrastruttura.

E' bene distinguere tra:
HIDS: Sistemi che analizzano informazioni relative all'attività locale di un singolo host.
NIDS: Sistemi che utilizzano le informazioni raccolte da analizzatori di traffico di rete.

La rilevazione può avvenire secondo due modalità differenti:
-Misuse Detection: si caratterizza l'abuso, ovvero Erge il rilevamento delle anomalie, che definendo il comportamento normale del sistema allerta ogni qual volta incontrasse ciò che non rientra nella sfera del “normale”, ovvero situazioni che ricadono nella descrizione di attacco.
In pratica: L'amministratore definisce pattern predefiniti di usi non conformi e il sistema ca a cerca dei matching di tali pattern tra gli eventi monitorati (tecnica più diffusa).
Problemi: vengono rilevati soltanto attacchi noti, e quindi piccoli varianti e problemi sconosciuti passano inosservati dall'IDS.
-Anomaly Detection: si caratterizza l'imprevedibilità, ovvero si rilevano le situazioni che si scostano dal “normale” funzionamento in modo da poter rilevare ance attacchi ancora sconosciuti.si cercano quindi attività impreviste.
In pratica: vengono rilevate azioni anomale di un utente rispetto all'uso predefinito, e

deviazioni rispetto a profili statistici. Non dipende in pratica dalla conoscenza di tutte le modalità di intrusione. Spesso viene utilizzato senza che ce ne rendiamo conto, l'esempio classico è il controllo dell'integrità di un file per mezzo dell'hash.
Problemi: E' molto complesso da organizzare e oneroso da gestire.

9° Lezione – 08/04/2013 (8)
Falsi allarmi

E' importante fare una distinzione fra “falsi negativi” (attacchi non rilevati) e “falsi positivi” (attacchi falsi rilevati corrispondenti a situazioni normali). Negli IDS occore bilanciare il rapporto tra i due indicatori, il difficile nella creazione degli IDS risiede in questa scelta.

N.B. In alcuni casi è più grave la presenza di falsi positivi, in altri la presenza di falsi negativi.

Vediamo ora alcuni **dettagli riguardanti i falsi allarmi**:
- più la rilevazione è specifica (abbiamo definito molto dettagliatamente cosa è lecito o no secondo l'IDS) più aumenterà il carico computazionale e la rilevazione diventa sensibile alle variazione dell'evento analizzato. Il problema si presenta nel momento in cui ci si imbatte in qualcosa di leggermente diverso dalle “precise” politiche definite, caso in cui l'IDS segnala un allarme.
- quanto più la rilevazione si fa lasca (ovvero dettagli generici, NON specifici) il problema è un altro, ovvero capita che a volte, anche qualcosa di dannoso, rischia di passare.

La proporzione dei due falsi allarmi non è solamente una funzione opinionistica, in quanto è matematicamente misurabile e verificabile. Abbiamo FP (falsi Positivi) e FN (Falsi Negativi)

	Positivo (allarme)	**Negativo (non allarme)**	**TOT**
Attacco	TP	FN	TotAttacchi
Non attacco	FP	TN	TotNonAttacchi
TOT	TotAllarmi	TotNonAllarmi	TotSegnalazioni

TP + TN + FP + FN = totale

Vediamo la t*erminologia*:

FP: Type 1 error, “falso allarme”
FN: Type 2 erroe, “allarme non segnalato”
TotAllarmi : numero di volte che scatta l'allarme
TotAllarmi : numero di volte che l'allarme non scatta
TotAttacchi : numero di attacchi
TotNonAttacchi : numero di non attacchi (operazioni normali))
TotSegnalazioni : numero di segnalazioni vere/false, positive/negative

Sensibilità del test, recall, hit rate, TPR “true positive rate” ---> (TP) / (TP+FN)

Indica quanti attacchi vengono segnalati (TP) sul numero totale di attacchi.
N.B. L'unico modo per far risultare 100% di attacchi rilevati (TP) è che FN = 0, ovvero IDS molto sensibile che analizza molto scrupolosamente il traffico.
PROBLEMI RELATIVI AGLI IDS: FALSI ALLARMI E MANCATE SEGNALAZIONI.

Specificità del test ---> (TN) / (TN+FN)
Stesso significato ma riferito al negativo, ovvero quanti allarmi negativi "TN" vi sono sul numero totale di attacchi.

Accuratezza del test ---> (TP+TN) / tot
Indica quanta allarmi corretti ci sono in tutto

Precisione del test ---> TP / (TP+FP)
Indica quanti allarmi veri (TRUE) vengono segnalati su tutti i positivi (ovvero su tutti gli allarmi)

FPR ---> FP / (TN+FP) ---> 1 - (specificità)
Indica quanti falsi alarmi ho su il totale dei non attacchi

N.B. Prendere la tabella precedente come riferimento per la terminologia.

Esempio

	Allarme	**Non Allarme**	**TOT**
Attacco	63	37	100
Non attacco	28	72	100
TOT	91	109	200

TPR = TP / (TP+FN) = TruePositive / TotAttacchi = 63/100 =
63%
Specificità = (TN) / (TN+FP) = TrueNegative / TotNonAttacchi = 72/100 =
72%
FPR = 1 – specificità = 1 – 0,72 =
28%
Accuratezza = (TP+TN) / tot = TotTrue/ TotSegnalazioni (63+72) / 200 = 135 / 200 =
67.5%
Precisione = TP / (TP+FP) =TruePositive / TotAllarmi = 63/91 =
69.3%

Per l'analisi degli IDS si utilizza ROC (Receiver Operating Characteristic), che confronta la sensibilità al tasso di falsi positivi (noi vorremmo altissimo TPR e bassissimo FPR) e l'AUC (Area Under Cover).

Questi strumenti misurano l'efficacia dello strumento utilizzato solo "ex-post" ovvero dopo il fatto, a posteriori. Per la stima della probabilità nel momento preciso in cui si riceve un attacco, con gli strumenti analizzati, non è possibile. Per poter fare i conti serve sapere quanto è probabile a *priori* un attacco in assoluto (vedi "Teorema di Bayes").

Un NIDS complementa altre soluzioni, con un'architettura a diversi livelli
(defense-in-depth) dove bisogna stabilire quanti sensori installare, dove installarli e come gestire i dati rilevati.
Un sensore può essere:
-Esterno: rileva più dati e più allarmi (in quanto rileva l'intero traffico diretto alla rete)
-Tra router e firewall: rileva tutto il traffico meno quello filtrato.
-Sulla rete dei servizi pubblici, dietro il firewall: tutto il traffico autorizzato dal firewall iretto ai servizi pubblici, vi è la possibilità di un filtraggio mirato.
-Sulla intranet: rileva sia il traffico da reti più esposte (DMZ) che interno alla intranet, eventuali usi illeciti interni, causa molti falsi allarmi.
-Segmento critico della rete aziendale: monitora le connessioni dirette ad alcune risorse particolarmente critiche.
-Interno: rileva solo il traffico che entra effettivamente nella rete verificando l'efficacia del firewall.