Sicurezza Delle Reti

Vulnerabilità

Sono delle particolari **debolezze** del sistema che possono essere sfruttate da un aggressore per portare un attacco

Bug = errore commesso in una delle fasi di sviluppo, può essere generato da persone, procedure e strumenti

Flaw = caratteristica non primaria derivata da un'errata concezione del progettista

Backdoor = vulnerabilità inserita di proposito, con lo scopo di garantire un accesso futuro a chi ha progettato il sistema, eludendo la sicurezza

Vettori di attacco

- **Phishing**
- **Spear Phishing** (phishing mirato)
- **Whaling**
- **Vishing** (Phishing con chiamate telefoniche)
- **ClickJacking** (Phishing con collegamento ipertestuale)
- **Impersonation**
- **Baiting** (Adescamento)
- **Dumpster Diving** (Raccolta di info da oggetti dismessi dall'utente)
- **Piggybacking** (ottenere accesso illecito tramite vicinanza all'autorizzato)
- **Bufale**
- **Click Baiting**
- **Baiting** (prende di mira utenti inesperti, creando discussioni aggressive)
- **Scam** (truffa)

Le Principali Minacce

- **Software malevolo** (attenta alla sicurezza dei sistemi informatici)

Classificazione del **Malware**

In base a **come si propaga**

- **Virus**: infezione di programmi e successiva diffusione
- **Worm**: sfruttamento di vulnerabilità e diffusione mediante download
- Attacchi basati su **Ingegneria sociale**, si attacca la persona, psicologicamente

In base alle **azioni che compie** quando arriva al target

- Corruzione di file
- Furto di informazioni
- Furto di servizi

Logic Bomb = codice inserito in un programma comune che si attiva con alcune condizioni che innesca azione innocua (scherzo) o dannosa (attacco)

Zombie = programma in grado di acquisire segretamente il controllo di un sistema di elaborazione

Troiano = programma apparentemente innocuo che in realtà induce effetti collaterali

Cryptolocker Ransomware = un sistema di crittografia che si diffonde via mail, informa che i dati crittografati non sono + leggibili e per sbloccarli nuovamente si indica una cifra e una scadenza per pagare o andranno persi per sempre, spesso tramite metodi di pagamento non tracciabili come i bitcoin

Wannacry = attacco documentato nel 2017 che infettò in poco tempo i sistemi pubblici di + di 150 nazioni, era un worm che infettava i sistemi windows sfruttando alcune vulnerabilità e, dopo aver cifrato i file, chiedeva un riscatto per decifrarli

Virus = è un programma che si nasconde dietro un altro programma installato dall'utente

Le fasi del **ciclo di vita** di un virus sono:

- **Fase silente**, il virus attende un evento che si deve verificare
- **Propagazione**, il virus si replica, inserendo dei propri cloni in altri programmi
- **Attivazione**, il virus viene svegliato da un evento
- **Esecuzione**, esegue l'azione per cui è stato programmato

Uno **stealth virus** è capace di eludere i sistemi di controllo antivirus

Un **polimorfic virus** è in grado di cambiare una parte del proprio codice ad ogni infezione, in modo tale da essere difficilmente rintracciabile siccome compare in forme diverse

Il **worm** è invece un programma che non infetta altri programmi, ma il suo obiettivo è quello di installare troiani, zombie o aprire backdoor nel sistema

Il worm ricerca autonomamente le proprie vittime e si diffonde tramite mail, esecuzioni remote, login remoti o file sharing

Lo **spyware** è un software che raccoglie informazioni sull'attività online degli utenti, inviandole poi a terzi per trarne profitto

Malvertising consiste nel collocare delle pubblicità su siti web senza comprometterne l'attività, ma mediante queste ads malevole gli attaccanti possono infettare le vittime

Gli attacchi più frequenti in rete sono:

Packet Sniffing = uno sniffer riceve passivamente tutti i frame del livello datalink, è utile per un amministratore di rete che vuole controllare il traffico della propria LAN, ma è uno strumento che potrebbe essere usato per scopi malevoli dagli attaccanti

Per evitare il packet sniffing si può utilizzare una cifratura per le informazioni, in modo tale che gli altri non riescano a leggere il contenuto del pacchetto.

Ip Spoofing = tecnica che permette di cambiare l'indirizzo ip sorgente di un datagram, un utente può modificare un datagram ip in modo tale da far apparire come se i dati associati siano stati spediti da un altro host con ip arbitrario

Pharming = tecnica utilizzata per ottenere accesso a informazioni personali e riservate

Spamming = invio di elevato numero di messaggi, spesso via mail

Defacing = cambia illecitamente la home page di un sito web o modifica 1 o + pagine interne con lo scopo di truffa, propaganda, ricatto o burla

Denial of Service = vengono inviate numerose richieste al server dall'attaccante in modo da non rendere il servizio disponibile ai veri utenti

BotNet = insieme di computer collegati in rete e controllati da un'unica entità detta botmaster, usati per attacchi distribuiti e ricevono info dai bot

Possibili difese

Attività di **prevenzione**, prima degli attacchi

Attività di **detezione** e **filtro**, durante gli attacchi

Attività di **traceback** delle sorgenti e di identificazione, durante e dopo gli attacchi

Attività di **reazione**, dopo gli attacchi

L'Exploit è un attacco che sfrutta alcuni bug, vulnerabilità o malfunzionamenti del sistema. Una possibile vulnerabilità è il **Buffer Overflow** = ci sono delle applicazioni che non controllano la lunghezza dei dati in ingresso, ma si limitano a scrivere questi dati su un buffer con una data lunghezza, i dati in eccesso potrebbero quindi sovrascrivere alcune variabili interne dell'app o essere sfruttati per bloccare o prendere il controllo del sistema

Il **KeyLogger** è uno strumento in grado di intercettare tutto ciò ce viene digitato sulla tastiera del computer, può essere installato con troiani e viene usato per acquisire dati sensibili dell'utente come ad esempio credenziali di accesso

Crittografia

Permette ad un sender di "camuffare" i dati così che un intruso non possa trarre informazione dai dati intercettati

Il ricevente ovviamente deve essere in grado di recuperare i dati originali a partire da quelli camuffati

Gli obiettivi della crittografia sono:

-**Confidenzialità**, nessuno estraneo deve poter leggere i dati scambiati

-**Integrità**, i dati non devono essere modificati

-**Autenticazione**, l'identità delle parti deve poter essere verificata

-**Non ripudio**, se una persona invia un messaggio non può negare di averlo fatto

Cifrario: un algoritmo per rendere oscuro, cifrare, un messaggio o per ripristinare un messaggio cifrato

Chiave: segreto usato nel cifrario per cifrare o decifrare messaggi

Cifrario simmetrico: si usa stessa chiave per cifrare e decifrare

Cifrario asimmetrico : la chiave usata per cifrare è diversa da quella usata per decifrare

Criptoanalisi : l'arte di violare i cifrari

Principio di Kerckhoffs

La regola fondamentale della crittografia è che il criptoanalista consce il metodo di codifica ma non conosce la chiave

Per ogni algoritmo di cifratura si definisce il **fattore di lavoro**, ovvero la difficoltà computazionale per un criptoanalista nel dover effettuare una ricerca esaustiva nello spazio delle possibili chiavi ammesse per violare l'algoritmo

L'impiego di chiavi lunghe nella cifratura impedisce possibili attacchi di forza bruta, nonostante ciò però a volte si possono ricavare le chiavi conoscendo l'algoritmo casuale di generazione degli elaboratori

Negli anni 70 veniva usata soltanto crittografia simmetrica, presuppone che le due parti che stanno comunicando conoscano a priori la chiave

Cifrario a sostituzione -> Semplice, usato anche nell'antichità, la **tabella** che contiene ogni carattere con quale altro deve essere sostituito è la chiave del cifrario

Cifrario di Cesare -> ogni lettera subisce uno shift di n posizioni, la chiave è il n di posizioni di shift

Due tipi di cifrari a sostituzione

Cifrario monoalfabetico -> si sostituisce una lettera sempre con un'altra

Cifrario polialfabetici -> un carattere non viene crittografato sempre con lo stesso carattere, ma con caratteri diversi in base ad una regola

Cifrario di Vigenere -> combina diversi cifrari a sostituzione, il testo viene diviso in blocchi e per ognuno viene applicata una cifratura diversa, la chiave è una parola che contiene quindi le lunghezze degli shift e il n di caratteri in ogni blocco

Macchine a rotori -> un cifrario a sostituzione con chiave ruotata a ogni cifratura

Cifrari a trasposizione -> riordinano le lettere invece di trasformarle come quelli a sostituzione

Un cifrario è **sicuro** quando in maniera informale il testo sembra "casuale", ovvero non dà nessuna informazione sul testo in chiaro

Questa proprietà è chiamata **segretezza perfetta** ed i cifrari che li raggiungono sono detti cifrari perfetti

I **cifrari a sostituzione non sono perfetti** perché viene mantenuta la frequenza delle lettere e quindi il cifrario dà informazioni all'attaccante

Per un cifrario perfetto utilizziamo l'Or Esclusivo, che viene fatto in base alla chiave

One Time Pad usa l'Or Esclusivo e la chiave k deve essere generata in maniera casuale

Il problema è che bisogna avere una chiave con lunghezza uguale almeno al testo e questa non dovrebbe essere trasmessa in condizioni di scarsa sicurezza ed inoltre il metodo di cifratura è sensibile alla perdita o inserzione di caratteri

La **crittografia moderna** impiega gli stessi principi di quella tradizionale con l'obiettivo di rendere l'algoritmo così complesso da non poter essere decifrato anche se in possesso di una grande mole di testo cifrato

Utilizza in genere cifrari a blocchi e può essere utilizzata cifratura che usa in maniera combinata blocchi di permutazione e di sostituzione

Blocco di permutazione cambia l'ordine di n bit in ingresso, mentre quello di **sostituzione** è formato da un decoder che prende in entrata 3 bit e alza una delle 8 uscite in base ai bit in entrata, poi abbiamo un blocco di permutazione che cambia l'ordine dei bit e poi un encoder che passa da 8 bit a 3 in uscita

Il **blocco prodotto** invece è costituito da una serie di blocchi di sostituzione e trasposizione in serie e parallelo

Il **DES** (**Data Encryption Standard**) è un algoritmo di cifratura a chiave simmetrica pubblicato nel 1977, usa chiavi a 56 bit per cifrare blocchi di testo da 64 bit concatenati

In ogni iterazione dell'algoritmo viene utilizzata una chiave diversa, siccome prima di essere utilizzata la chiave subisce una trasposizione a 56 bit, viene frazionata in 2 blocchi da 28 bit e ognuno di questi subisce uno shift circolare di 1 o 2 bit e dopo queste operazioni si ottiene una chiave da 48 bit differente a ogni iterazione

Per aumentare la sicurezza di questo algoritmo si può utilizzare il **3-DES**, ovvero si applica il DES 3 volte consecutive con 3 chiavi simmetriche diverse

Nel 1997 vennero richieste delle idee per realizzare un nuovo standard detto **AES** (**Advanced Encryption Standard**)

Le idee di base dovevano essere:

-Cifrario a blocchi simmetrico

-Progetto doveva essere pubblico

-Chiavi da 128, 192 e 256 bit

-Dovevano essere possibili sia implementazioni software che hardware

-Algoritmo pubblico o adottabile su licenza

Nel 2001 **Rijndael** è diventato uno standard del governo statunitense, supporta lunghezze di chiavi e dimensioni dei blocchi da 128 a 256 bit, in passi da 32

Con una chiave da 128 bit, per condurre un attacco di forza bruta ci vorrebbero 10^{10} anni avendo a disposizione un bilione di processori

Ogni passo dell'algoritmo è reversibile e per fare la decifratura viene utilizzato l'algoritmo inverso. L'algoritmo è stato progettato per offrire elevata sicurezza e velocità di esecuzione e utilizza delle **matrici 4X4** per cifrare i vari blocchi

I **block Ciphers** sono delle vere e proprie permutazioni, determinate dalla chiave scelta

Bisogna vedere come cifrare dati + lunghi di un blocco di n bit

Il **padding** viene usato quando il n di bit del messaggio non è multiplo della grandezza del blocco, il padding completa il blocco con dei byte con valore il num di byte mancanti

Il problema è che AES e DES si comportano come cifrari a sostituzione monoalfabetici con caratteri da 64 e 128 bit, quindi conoscendo come sono strutturati i dati, anche senza conoscere i dati contenuti stessi, si possono fare degli attacchi mirati

Per prevenire questi attacchi si è pensato alla concatenazione a blocchi, tecnica attraverso la quale la codifica di un blocco dipende anche da ciò che è stato cifrato in precedenza. Il primo blocco viene chiamato **Initialization Vector** e serve a far partire la codifica, poi per ogni blocco verrà utilizzato il blocco precedente

Il **block chaining** è una tecnica che ad ogni passaggio mette un blocco in xor con il blocco precedente cifrato

Il **feedback** invece consente di effettuare la cifratura byte a byte e utilizza uno shift register, che si riempie con gli ultimi byte codificati e cambia quindi ad ogni iterazione

Lo **Stream** è una modalità di concatenazione che può essere impiegata nei casi in cui non è possibile tollerare l'incapacità a decifrare 64 bit se si verifica un errore di trasmissione di un solo bit; cifra un IV con una chiave per ottenere un blocco di output, che viene cifrato poi per ottenere un secondo blocco di output e così via

Il problema delle modalità di concatenazione è che non è possibile effettuare accesso random ai dati.

Il **Counter** consente l'accesso random a blocchi cifrati

Il problema fondamentale è : **come condividere le chiavi**?

Ovviamente non è possibile che due persone che devono comunicare devono scambiarsi una chiave fisicamente, anche perché molte volte si parla con persone che non si conoscono o sono geograficamente molto lontane tra loro

Si potrebbe pensare a una **Trusted Third Party**: un server centrale (Key Distribution Center) che ha una chiave condivisa con ogni persona

Questa soluzione ha senso in ambienti chiusi tipo università o aziende; se la TTP smette di funzionare nessuno può più comunicare

Si può gestire uno scambio di chiavi sicuro e “online”, impiegando tecniche basate sul concetto di “**Key Agreement**”, che consentono di costruire un segreto condiviso mediante uno scambio preliminare di informazioni

Merkle Puzzles = Alice e Bob si accordano su un cifrario a blocchi, Alice manda una serie di puzzle a Bob, Bob ne sceglie uno, lo risolve provando tutte le possibili chiavi e manda ad Alice una parte della soluzione, la parte di soluzione non inviata è la chiave che utilizzeranno per comunicare. Un attaccante per trovare la chiave dovrebbe provare a risolvere tutti i possibili puzzle con tutte le possibili chiavi, questo algoritmo risulta anche però oneroso per Alice e Bob

Scambio di chiavi Diffie Hellman = basato sul problema del logaritmo discreto, in grado di dare un gap esponenziale tra la complessità per gli utenti e per gli attaccanti, Alice e Bob scelgono due numeri primi molto grandi e attraverso uno scambio di numeri che sfruttano delle proprietà matematiche che rendono difficile la decodifica della chiave per l’attaccante, si raggiunge una sicurezza comparabile a un block cipher a 128 bit

I problemi sono che a parità di dimensione, alcuni numeri primi sono più deboli di altri. Inoltre l'algoritmo è vulnerabile ad attacchi attivi, come il **man-in-the-middle**

Crittografia a chiave pubblica o asimmetrica = la chiave pubblica e viene usata per l’algoritmo di cifratura, mentre la chiave privata d viene usata per la decodifica

Il legame tra d ed e permette di effettuare cifratura e decifratura con chiavi diverse
Il sistema **RSA** è il primo schema a chiave pubblica, pubblicato nel 1977, basato sul problema della fattorizzazione ed è caratterizzato da un fattore critico costituito dalla scelta delle chiavi pubblica e privata

Si devono scegliere 2 numeri primi, tipicamente di 1024 bit ciascuno e si calcolano n= p*q e z=(p-1)*(q-1) e si sceglie e che non abbia nessun fattore comune con z, d invece si sceglie in modo che ed-1 risulti divisibile per z

La chiave pubblica sarà costituita dalla coppia (n,e) e quella privata dalla coppia (n,d)

Il **fattore di sicurezza** è legato alla difficoltà di fattorizzare i grandi numeri

L’RSA è utilizzato per cifrare/decifrare messaggi, scambiare le chiavi per una cifratura simmetrica e implementare la firma digitale

La **Firma Digitale** indica il proprietario o il creatore di un documento o messaggio, deve essere verificabile, non falsificabile e non rifiutabile.

Rende possibile provare che un documento firmato da un individuo sia stato davvero firmato da quell’individuo in forma verificata

La crittografia a chiave pubblica è costosa da un punto di vista computazionale, si può usare un approccio più efficiente che fa uso di un **message digest** come sola parte del messaggio ad essere cifrata a chiave pubblica e a costituire la firma digitale, il message digest è una sorta di impronta digitale, calcolata come MD(P), implementata usando **funzioni di hash**, che produce una stringa di lunghezza fissa a seconda del messaggio

Bob invia un messaggio ad Alice e usa una funzione di hash per generare il digest e lo cifra con chiave privata, Alice riceve il messaggio e applica la chiave pubblica per recuperare il digest, poi applica la funzione di hash per ottenere il secondo digest, se i due digest sono uguali, l'autenticità e l'integrità del messaggio sono garantiti

Pretty Good Privacy (PDG) è uno schema di protezione delle e-mail, sfrutta il digest, una crittografia a chiave simmetrica e fa la compressione dei dati

Posta elettronica certificata è uno strumento che ti permette di attribuire ad una mail lo stesso valore di una raccomandata

Per lo scambio delle chiavi si utilizzano degli intermediari di fiducia di cui noi ci fidiamo, ad esempio la **Certification Authority** certifica che una chiave pubblica appartiene ad una determinata entità, mentre il **Key Distribution Center** è una singola entità di fiducia della rete con cui viene stabilita una chiave segreta condivisa da impiegare per ottenere singole chiavi di sessione

Una CA assegna chiavi pubbliche a chi le richiede, se Alice vuole la chiave pubblica di Bob deve recuperare il certificato di Bob, deve applicare la chiave pubblica della CA al certificato per autenticare la chiave pubblica di Bob unitamente alla sua integrità

Autenticazione è il processo che prova l'identità di qualcuno a qualcun altro, avviene mentre la comunicazione è effettivamente in atto, basata su un protocollo a chiave segreta simmetrica o a chiave pubblica

Alla base dell'autenticazione ci sono il concetto di **sfida**, che i due interlocutori si scambiano attraverso un numero casuale chiamato **nounce**.

Spesso i protocolli di autenticazione consentono di negoziare una chiave di sessione mediante la quale poter cifrare i dati scambiati successivamente, bisogna usare una **chiave nuova per ogni sessione** per minimizzare il danno se la chiave dovesse essere scoperta e a ridurre il testo utile che l'attaccante può usare per decifrare i messaggi della conversazione

Vari tipi di **protocolli di autenticazione**

-**In chiaro** -> è il più semplice, consiste solo nell'affermare di essere qualcuno

-Basato su **indirizzo IP** -> si verifica l'indirizzo IP dei datagram che trasportano il messaggio di autenticazione

-Basato su **password in chiaro** -> viene inviata una password in chiaro per identificarsi

-Basato su **password cifrata** -> si usa l'approccio della crittografia a chiave simmetrica per cifrare la password

Attraverso un **attacco di riflessione**, una terza persona potrebbe avviare più sessioni e nella seconda sessione si fa cifrare il nounce Ra per restituirlo nella prima sessione

-Basato su **HMAC**, ovvero l'Hashed Message Authentication Code, ovvero un diggest calcolato su un blocco di dati che contiene la chiave simmetrica

Un approccio diverso alla condivisione di chiavi segrete è basato su KDC (Key Distribution Center), che gestisce una chiave segreta per ogni user

Ap Otway e Rees impiega sfide comuni ed individuali, il KDC autentica Alice e Bob valutando l'uguaglianza degli R (sfide) nei token

Kerberos coinvolge tre server, oltre ad Alice:

-**Server di autenticazione**

-**Server di gestione dei ticket** per la prova di identità

-Il **server** di Bob **a cui vuole accedere** Alice

Kerberos è idoneo a gestire l'autenticazione in ambito locale a causa dell'impiego di server centralizzati

Alice quando avvia una sessione, riceve la chiave di sessione e il ticket, successivamente Alice effettua il login attraverso la password, una volta loggata riceve una chiave locale usata per estrarre la chiave, poi viene distrutta la password di Alice, che tanto ormai è dotata di ticket e può chiedere al TGS le credenziali per accedere a Bob. Alice effettua la richiesta al TGS, usando il ticket ricevuto prima, poi il TGS restituisce la chiave di sessione in due versioni: una cifrata con Ks, in modo tale che Alice possa leggerla e l'altra codificata con la chiave di Bob così che Bob possa leggerla

Infine Alice richiede a Bob di accedere inviandogli il token, che permette a Bob di autentificare Alice e che contiene la chiave di sessione, poi Bob restituendo il timestamp aggiornato, si autentica a sua volta con Alice

Iterazioni sicure

Possono essere garantite implementando specifiche soluzioni a diversi livelli del modello ISO/OSI.

Secure Socket Layer/ Transport Layer Security

Protocollo progettato per implementare una connessione sicura tra due socket

Consente la negoziazione di parametri per la connessione sicura tra client e server, l'autenticazione mutua tra client e server, l'integrità dei dati e la comunicazione segreta

SSL/TLS giace tra il livello delle applicazioni e quello del trasporto

Dal lato sender SSL riceve i dati da un'applicazione, li cifra e li indirizza ad una socket TCP, dal lato receiver invece SSL legge i dati da una socket TCP, li decifra e li indirizza all'applicazione

Permette ad un utente, mediante il suo browser, di **confermare l'identità** di un server mediante la richiesta e ricezione della sua chiave pubblica tramite certificato

L'autenticazione del client permette ad un server di confermare l'identità di un utente, usa lo stesso meccanismo per riconoscere il server, ma l'autenticazione del client è opzionale.

Il server specifica la catena del suo certificato, il browser ricerca all'interno del proprio elenco di CA al fine di ricavare la chiave pubblica del server; una volta ottenuta, il browser genera una chiave random a 384 bit che invia cifrata con la chiave pubblica del server

Dopo che client e server hanno calcolato la chiave di sessione, possono terminare il sub-protocollo di handshake, passando alla fase di cifratura della sessione

Una **VPN** è una virtual private network, ovvero una rete privata virtuale, mappata su rete pubblica e costituita da circuiti virtuali che consentono di restringere la comunicazione ad un dato insieme di host. Per collegare gli host vengono utilizzati i **tunnel**, ovvero connessioni punto-punto fra una coppia di host che permettono di restringere la comunicazione mediante IP fra coppie di host

Ogniqualvolta un router all'ingresso di un tunnel vuole inviare un datagram IP attraverso il tunnel, deve **incapsulare il datagram** in un nuovo datagram IP il cui indirizzo destinazione è quello del router all'altra estremità del tunnel, mentre quello sorgente coincide con il proprio indirizzo

L'**IP Security** è una suite di protocolli e servizi che fornisce sicurezza allo strato della rete, consente di selezionare i servizi di sicurezza da adottare, scegliere l'algoritmo di cifratura nelle comunicazioni e selezionare la granularità dei servizi di sicurezza

L'architettura di **IPsec** è costituita da:

-Una coppia di protocolli che implementano i servizi di sicurezza (**AH** e **ESP**)

-Un **protocollo specifico di gestione delle chiavi e dei servizi di sicurezza** (ISAKMP)

-Un'astrazione che collega i protocolli ai servizi di sicurezza (**SA security association**)

AH e ESP sono i 2 protocolli fondamentali di IPsec

AH (**Authentication Header**) fornisce i servizi di **autenticazione** della sorgente e di **integrità** dei dati

ESP (**Encapsulation Security Payload**), oltre a fornire i servizi di AH, fornisce la **segretezza** dei dati attraverso meccanismi di cifratura

SA (Security Association) è il **canale logico** che permette di far viaggiare un datagram da un host sorgente a un host destinazione, è una connessione unidirezionale

ISAKMP (Internet Security Association and Key Management Protocol)

Abbiamo due modalità operative:

-**Modalità di trasporto**: l'header IPSec (AH o ESP) è inserito subito dopo l'header Ip

-**Modalità tunnel**: il datagram IP originario è incapsulato in un nuovo datagram IP, il cui header è seguito da quello IPsec, si può controllare instradamento organizzandolo solo fra host sicuri

Secure HTTP aggiunge sicurezza al solo http, implementa un contenitore digitale sicuro per incapsulare i messaggi http e fornisce degli header addizionali di sicurezza in base a quanto negoziato tra client e server; supporta cifratura, autenticazione, integrità dei dati, non ripudiabilità dei dati

L'SHTTP può essere impiegato o direttamente sullo strato del trasporto o su un protocollo di trasporto sicuro

S/MIME (Secure / Multipurpose Internet Mail Extension) è un miglioramento di sicurezza applicato al formato di posta elettronica, fornisce la capacità di firmare e/o cifrare i messaggi di posta elettronica

La **sicurezza** in **LAN Wireless** è molto difficile da ottenere perché l'impiego di comunicazione wireless su canale broadcast rende più facile sniffare il traffico avendo un semplice notebook collocato a portata della capacità di connessione della LAN

Il primo protocollo di sicurezza a **livello Data Link** impiegato è stato il **WEP** (Wired Equivalent Privacy) :

-Ogni stazione ha una chiave segreta condivisa con la stazione base

-Viene impiegata una cifratura a stream

-Per ogni pacchetto è calcolato il checksum che viene accodato al pacchetto in modo da formare il testo in chiaro da cifrare

Il WEP presenta diverse falle che lo rendono vulnerabile, le chiavi degli utenti potrebbero essere uguali, ma anche se sono distinte sarebbe necessario utilizzare un IV per ogni pacchetto inviato, in modo da evitare attacchi

L'attacco può essere portato avanti analizzando due pacchetti cifrati con lo stesso IV e la stessa chiave, è possibile ricavare il testo in chiaro facendo lo XOR di 2 testi cifrati

WPA

Gestisce l'autenticazione, la cifratura RC4 con chiave a 128 bit e IV a 48 bit, **integrità** dei dati attraverso un **MIC (Message Integrity Code)** e contromisure extra

WPA 2

Gestisce l'autenticazione e integrità sviluppate intorno all'AES a 128 bit e nessun re-impiego dei IV

Sicurezza in Bluetooth

La portata ridotta limita decisamente i problemi di sicurezza rispetto all'802.1x, Bluetooth prevede 3 modalità per la sicurezza e la implementa a diversi livelli

Quando due dispositivi si vogliono connettere tramite bluetooth si scambiano una **passkey** che può essere stabilita dal produttore se i 2 dispositivi sono realizzati dallo stesso produttore o viene preinserita in un dispositivo e deve essere digitata dall'utente al momento della connessione

Il bluetooth **autentica** solo i **dispositivi**, non gli utenti siccome dà per scontato che il dispositivo sia di uso personale di chi lo possiede

Possibili problemi in rete (Slide 14 4.01)

Se Alice vuole visitare il sito di Bob, scrive l'URL di Bob e recupera la pag web, ma è davvero la pag di Bob? Trudy potrebbe intercettare i pacchetti e modificarli all'insaputa di Alice

Quando Alice interroga il suo per ottenere indirizzo IP di Bob, il server propaga l'interrogazione alla ricerca di una risposta autoritativa

Trudy può restituire al DNS di Alice un record DNS di Bob modificato che contiene l'indirizzo IP di Trudy, Trudy deve fare ciò però molto velocemente, prima che arrivi la vera risposta in modo che questa quando arriverà sarà scartata

Adesso tutto il traffico destinato a Bob raggiunge in realtà Trudy, anche perché anche la cache del DNS di Alice salverà come indirizzo di Bob quello mandato da Trudy

Per associare le richieste alle risposte il server DNS impiega dei **numeri di sequenza**, quindi per fare **spoofing** Trudy deve conoscere il n di sequenza della richiesta di Alice

Per sapere il n di sequenza Trudy registra un suo dominio e un suo server DNS, poi richiede l'indirizzo del proprio sito per scoprire il n di sequenza, poi velocemente visita il sito di Bob e legittima la query con numero di sequenza n + 1, rispondendo con il proprio indirizzo IP e inserendo quindi in cache l'indirizzo sbagliato che sarà poi usato da Alice per andare sul sito di Bob, che in realtà sarà quello di Trudy, facendo poi scartare la vera risposta

Il problema è che DNS fu progettato quando internet aveva pochi utenti e non era importante la sicurezza, nel 1994 però l'IETF ha dato vita al progetto **DNSSec**, basato su crittografia a chiave pubblica e gestisce sicurezza nella provenienza dei dati, distribuzione delle chiavi pubbliche e autenticazione della transazione e della richiesta

Tutte le info inviate da un server DNS di zona vengono cifrate con la **chiave privata di zona** e i DNS che richiedono una risoluzione attraverso la chiave pubblica di zona possono verificare autenticità di provenienza e integrità dei dati

Gli **RRSets** sono insiemi di **record raggruppati**, vengono firmati attraverso una funzione hash il cui digest è cifrato con la chiave privata di zona, per verificarne l'autenticità e l'integrità basta decifrarlo con la chiave pubblica di zona

Firewall è uno strumento che serve a **proteggere** un determinato dominio, rete privata o computer **da accessi indesiderati**, si interpone tra la rete esterna e quella di interesse in modo da diventare l'unica via di comunicazione tra loro

Esistono diverse tipologie di firewall:

-Il **packet filtering** è implementato mediante 1 o più router posti tra rete interna e esterna, che consentono il passaggio solo al traffico autorizzato, il router verifica indirizzi IP e n di porte dei pacchetti che transitano e in base ad alcune **regole di filtro** preimpostate filtrano il traffico

-**Application Gateway** intercetta il traffico e **autentica gli utenti** a livello delle applicazioni, un utente che desidera eseguire un'applicazione accede al proxy che lo autentica, dopo di che ha accesso al server remoto che si trova in internet, questo controllo viene effettuato anche per tutte le comunicazioni provenienti da internet; operando a questo livello c'è bisogno di un **proxy** per ogni applicazione

-**Packet Inspection**, in questo caso il firewall effettua l'analisi anche del contenuto dei pacchetti ed offre quindi il massimo grado di affidabilità

Gli **Antispyware** sono dei moduli che controllano che non ci siano spyware nei file

I **NIDS (Network Intrusion Detectiom System)** individuano possibili tentativi di intrusione nei sistemi connessi in rete e forniscono report delle azioni di attacco ricevute, si occupano anche di rilevare dati anomali o inappropriati, che vengono considerati non autorizzati, di scoprire i pattern d'attacco e di avvisare l'amministratore di rete quando necessario

I NIDS, utilizzati insieme ai firewall, costituiscono un importante mezzo di difesa dei sistemi di rete

Un **NIDS** può essere posto tra

-**Firewall e rete esterna**, in questo caso è possibile rilevare tutti gli attacchi, anche quelli che possono essere bloccati dai firewall

-**Firewall e rete interna**, in questo caso è possibile monitorare solo il traffico già filtrato dal firewall, che non è detto sia sempre benigno

Si può anche utilizzare un **approccio distribuito** per i NIDS, si installano una serie di sensori, uno per ogni segmento di rete, che fanno tutti capo a un sistema centrale

I NIDS possono essere attaccati in diversi modi:

-Si tenta di contrastare l'analisi dei protocolli

-Si fa leva sulla starvasion delle risorse per disabilitare il sistema

L'**insertion** è un attacco condotto che cerca di riempire l'IDS di pacchetti subdolamente validi

Un IDS può infatti ritenere valido un pacchetto che un end-system scarta, l'attaccante sfrutta questa condizione inviando pacchetti che vengono scartati dall'end-system ma che vengono ritenuti validi dall'IDS, ad esempio vengono inviati caratteri uno alla volta che singolarmente non vengono ritenuti dannosi dall'IDS ma che in realtà una volta uniti lo diventano per l'end system

L'**evasion** è invece un attacco che sfrutta le inconsistenze fra analizzatore e end system, un end system può accettare un pacchetto che un IDS invece scarta perché malformato, questa condizione viene sfruttata per far scivolare delle informazioni cruciali in pacchetti che l'IDS non intercetta perché dotato di regole stringenti che non gli permettono di esaminare questi pacchetti, che quindi non vengono supervisionati

Se un attaccante riesce a mandare in crash il NIDS o a bloccare le sue risorse, può attaccare la rete come se il NIDS non ci fosse proprio

Il **DoS** infatti cerca di confondere il NIDS mandando molti pacchetti che vengono interpretati come tentativo di intrusione, mandando in **crash il sistema** o a causa di un esaurimento di risorse a disposizione del NIDS o a causa di un esaurimento di memoria

Gli **antivirus** consentono di individuare un'infezione, identificare il virus che l'ha provocata, rimuovere il virus e tutte le sue tracce

Gli antivirus di **prima generazione** operano cercando la **signature** dei virus o osservando modifiche nella lunghezza dei programmi

Quelli di **seconda generazione** operano in base a **regole euristiche**, ricercando particolari frammenti di codice usato dai virus

Quelli di **terza generazione** sono residenti in memoria e cercano di scovare i virus sulla base delle **azioni** che questi tentano di eseguire

Infine quelli di **quarta generazione** impiegano **tecniche congiunte**, di tipo euristico, basate sullo scanning delle signature e sull'analisi delle attività

I **generic decryption Antivirus** sono in grado di trovare anche **virus polimorfici** grazie all'impiego di un simulatore sicuro della CPU: i programmi non vengono eseguiti direttamente sulla CPU ma su un suo **emulatore** in modo tale da non andarla a intaccare

I **behaviour blocking antivirus** operano in maniera integrata al SO ed effettuano un **monitoring real time** dell'esecuzione dei programmi

Politica per la sicurezza

Richiede alcune fasi fondamentali:

-**Analisi del contesto** (tipologia di ente o azienda, struttura degli uffici, personale coinvolto, vincoli sui servizi o sui processi)

-**Analisi del sistema informatico** (ricognizione delle risorse a disposizione)

-**Classificazione di utenti e dei diritti di accesso** (chi fa cosa, analisi delle competenze)

-**Analisi di vulnerabilità e rischi** (Individuare le vulnerabilità e i rischi che si corrono in base alla priorità di intervento)

-**Individuazione e pianificazione delle contromisure**

Classificazione dei rischi

Avviene in base ai livelli di **severità** e **probabilità**

Per quanto riguarda la severità il livello 1 è quello **catastrofico**, che comporta un blocco totale del sistema

Il livello 2 è **critico**, dove abbiamo un blocco del sistema che richiede delle reinstallazioni e procedure di recupero dei dati

Il livello 3 è **marginale**, malfunzionamenti occasionali

Il livello 4 è **irrilevante**

Per quanto riguarda la probabilità invece il livello A è **frequente**, B **probabile**, C **occasionale**, D **remota** ed E **improbabile**

In base a severità e probabilità viene costruita una **matrice dei rischi** in cui i rischi in base a questi due livelli, si classificano in inaccettabile, indesiderabile, accettabile e sempre accettabile

Le **contromisure** da adottare devono essere valutate in base alla gravità dei rischi e a quanto costa adottarle o non adottarle

Praticamente bisogna sia calcolare la perdita che si avrebbe senza contromisure, sia il costo delle contromisure e quanto si limitino i danni adottandole, siccome a volte potrebbe capitare che le contromisure costano più della possibile perdita che si avrebbe ed è per questo che in quel caso non conviene adottarle

Il **ciclo fondamentale** che si segue nell'analisi dei rischi e delle possibili contromisure è un **modello operativo continuo** composto da 4 fasi cicliche: Plan, Do, Check, Act

L'amministratore di un sistema dovrebbe infatti essere **periodicamente informato** riguardo alle falle e ai problemi che si presentano, in modo tale da analizzarle e da poter ideare una strategia da mettere in atto