Laboratorio di Amministratore di Sistema

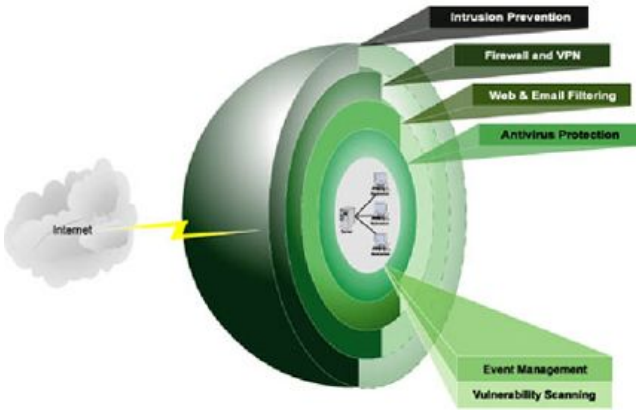
9. Sicurezza di rete

[Cisco ITESS II - Chapter 14]

Università di Venezia – Facoltà di Informatica
feb-mag 2013 - A. Memo

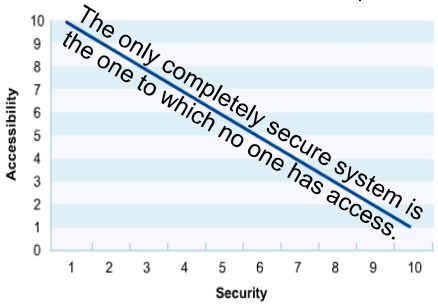
ver 2.1

Developing a Network Security Policy



Accessing Security Needs

- There must always be a delicate balance between **security** and **accessibility**.
- The more accessible a network is, the less secure it is.
- When it comes to a computer network, how much security is enough?



- There are several factors to consider:
  - The type of business in which the company engages
  - The type of data stored on the network
  - The management philosophy of the organization

Accessing Security Needs

- Type of business
  - Some business, sucha s law or medicine, by their nature generate confidential data. Law protects privacy of clients.
- Type of data
  - Certain types of data are considered to be private and should be protected (payroll records, employees' personal information, accounting and tax information, trade secret, …)
- Management philosophy
  - If data on the network is not subject to privacy law, the security level might depend on the personal philosophie of the management.

Acceptable Use Policy

- The first step in creating a security policy for a company network is to define an Acceptable Use Policy (AUP).
- An AUP tells the users what is acceptable and allowed on the company network.
- The AUP can include information on installing software or hardware.
- To view some examples of AUPs, visit these websites:

• http://www.freeservers.com/policies/acceptable_use.html

• http://www.rice.edu/armadillo/acceptable.html

Politica di utilizzo accettabile

- è un insieme di regole applicate dal proprietario / gestore di una rete , sito web o sistema di computer di grandi dimensioni che limitano i modi in cui la rete del sito o del sistema può essere utilizzato.
- L'AUP viene definita da società, imprese, università, scuole, fornitori di servizi Internet e proprietari di siti web spesso per ridurre il rischio di azioni legali che possono essere prese da un utente, e spesso con poche prospettive di applicazione.
- L'AUP è parte integrante del quadro generale di sicurezza, ed è spesso pratica comune chiedere ai nuovi membri di una organizzazione di firmare un AUP prima di avere accesso a sistemi informatici.

freeservers POLICIES

ACCEPTABLE USE POLICY

Freeservers has a **zero tolerance** policy for **spam, pornography, and warez**. Any sites found to contain, promote, or link to such content are subject to immediate removal from our service.

By typing in the 4-character security code and clicking the "submit" button at the bottom of the registration page, you accept these terms and conditions and acknowledge that your use of Freeservers is subject to certain limitations set forth below. If you do not accept this agreement, do not proceed with the sign-up process.

I. For the purposes of this agreement, Freeservers, Freeservers.com, and About Web Services may be used interchangeably.

II. Freeservers.com cannot be held responsible for the content of pages hosted under our service. Freeservers does not review pages for content before they are posted and does not verify, endorse, or otherwise take responsibility for the content of any user-created pages. However, we reserve the right to remove any page from our servers which we determine is violating our rules and guidelines. **Users are solely responsible for all files contained in their own directory,** and can be held legally liable for the contents of their Web site.

Penn Computing

Computing : Menu ...

Policy on Acceptable Use of Electronic I

Summary

This policy defines the boundaries of "acceptable use" of lim electronic information sources, as detailed below. It includes information environment evolves.

The policy is based on the principle that the electronic inform and service. Other uses are secondary. Uses that threaten t system; the privacy or actual or perceived safety of others; o

By using University electronic information systems you assume personal responsibility for their appropriate use and agree to comply with this policy and other applicable University policies, as well as City, State and Federal laws and regulations, as detailed below.

The policy defines penalties for infractions, up to and including loss of system access, employment termination or expulsion. In addition some activities may lead to risk of legal liability, both civil and criminal.

Users of electronic information systems are urged in their own interest to review and understand the contents of this policy.

Purposes

The University of Pennsylvania makes computing resources (including, but not limited to, computer facilities and services, computers, networks, electronic mail, electronic information and data, and video and voice services) available to faculty, students, staff, registered guests, and the general public to support the educational, research and service missions of the University.

Username and Password Standards

- Usually the system administrator will define the naming convention for the usernames on a network.
- A common example is the first initial of the person's first name and then the entire last name.

John Doe → jdoe
Kevin Smith → ksmith
Mary Smith → msmith

- A complex username naming convention is not as important as having a complex password standard.
- When assigning passwords, the level of password control should match the level of protection required.
  - Password should expire after a specific period of time
  - Passwords should contain a mixture of letters and numbers so they cannot easily be broken.

Virus Protection Standards

- Some standards requires that current virus protection software be installed on all the systems on the network.
- Place proper filters and access lists on all the incoming gateways to protect the network from unwanted access.
- To prevent viruses, e-mail policies also need to be developed that state what may be sent and received.

University of California Electronic Communications Policy

The University of California Electronic Communications Policy was originally issued November 17, 2000. A revision was issued in 2005.

Policy Issuance Letter
- August 18, 2005 (pdf)

Policy
- Electronic Communications Policy (pdf) (html)
- Attachment 1, ECP User Advisories (pdf) (html)
- Attachment 2, ECP Implementation Guidelines(pd

Major Changes in August 2005 Revision

Key Points on the Use of E-mail at UCOP

Getting the Message, Highlights of the ECP (pdf)

Campus and UCOP ECP Coordinators

Nonconsensual Access Requires Formal Authori
- **Contact your campus ECP coordinator fo**
- **Campuses may adapt and use the UCOP (pdf)**
- **Annual Reports on Nonconsensual Acce**

University Electronic Mail Student Notification Policy (Use of E-mail for Official Correspondence to Students)

Last revised: May 24, 2004 **Last reviewed:** May 24, 2004

Answers to Frequently Asked Questions

A. Policy Statement

Electronic mail (e-mail), like postal mail, is a mechanism for official University communication to students. The University will exercise the right to send e-mail communications to all students, and the University will expect that e-mail communications will be received and read in a timely manner.

B. Scope

This policy applies to all admitted and enrolled students of The University of Texas at Austin. Official communications using e-mail can include e-mail to a group, such as all admitted students, or an e-mail message to only one student.

- http://www.utexas.edu/policies/email/#policy
- http://www.ucop.edu/ucophome/policies/email/email.html
- http://www.onet.on.ca/onetspam.html

Security Policy

SANS (SysAdmin, Audit, Network, Security) Institute

1. **Acquisition Assessment Policy**
   defines responsibilities regarding corporate acquisitions, and defines the minimum requirements of an acquisition assessment to be completed by the information security group.
2. **Bluetooth Device Security Policy**
   this policy provides for more secure Bluetooth Device operations. It protects the company from loss of Personally Identifiable Information (PII) and proprietary company data.
3. **Dial-in Access Policy**
   defines appropriate dial-in access and its use by authorized personnel.
4. **Ethics Policy**
   defines the means to establish a culture of openness, trust and integrity in business practices.

Security Policy

SANS (SysAdmin, Audit, Network, Security) Institute

5. **Information Sensitivity Policy**
   defines the requirements for classifying and securing the organization's information in a manner appropriate to its sensitivity level.
6. **Internal Lab Security Policy**
   defines requirements for internal labs to ensure that confidential information and technologies are not compromised, and that production services and interests of the organization are protected from lab activities.
7. **Personal Communication Devices Policy**
   describes Information Security's requirements for Personal Communication Devices and Voicemail.

Security Policy

SANS (SysAdmin, Audit, Network, Security) Institute

8. **Risk Assessment Policy**
   defines the requirements and provides the authority for the information security team to identify, assess, and remediate risks to the organization's information infrastructure associated with conducting business.
9. **Technology Equipment Disposal**
   defines the rules for disposing of obsolete technology assets and computer equipment recycling
9. **Web Application Security Assessment Policy**
   defines the assessment of Web applications to identify potential or realized weaknesses as a result of inadvertent mis-configuration, weak authentication, insufficient error handling, sensitive information leakage, etc.

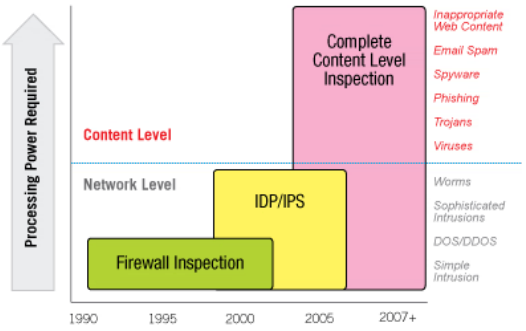
Online Security Resources

- In late 1988, a 23-year old graduate student at Cornell University released a self-replicating worm on the Internet. In a matter of hours, the rapidly spreading worm resulted in the shutdown of over 60,000 UNIX computers at universities and military facilities.
- Web-based resources offer critical information and powerful tools that can be used to protect a network. Some of the best online security resources are the NOS manufacturer websites

http://www.cert.org	http://www.redhat.com
http://www.microsoft.com	http://www.nipc.gov

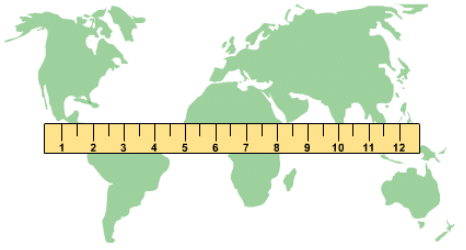
Threats of Network Security

Today's content-based threats, which bypass conventional firewalls, spread faster and do more damage.



Overview: Internal/External Security

- The Internet essentially works by following rules that are open to the public.
- If one studies the rules enough, one is bound to find loopholes and weaknesses that can be exploited.



Network security is essential because the Internet has made networked computers accessible and vulnerable.

- The number of individuals, organizations, and institutions connected to the Internet are growing.
- Connecting to the Internet opens the door to network intruders.
- In addition to outside threats from the Internet, corporate networks face numerous inside security concerns.
- Well-implemented security policies can minimize the risk posed by these scenarios

Security vulnerabilities within Linux services

The ten more critical internet security vulnerabilities in Linux

1. BIND Domain Name System
2. Remote Procedure Calls (RPC)
3. Apache Web Server
4. General UNIX Authentication Accounts with No Passwords or Weak Passwords
5. Clear Text Services
6. Sendmail
7. Simple Network Management Protocol (SNMP)
8. Secure Shell (SSH)
9. Misconfiguration of Enterprise Services NIS/NFS
10. Open Secure Sockets Layer (SSL)

Security vulnerabilities within Linux services

SANS Top-20 2007 Security Risks (2007 Annual Update)

Client-side Vulnerabilities in:

C1. Web Browsers
C2. Office Software
C3. Email Clients
C4. Media Players

Server-side Vulnerabilities in:

S1. Web Applications
S2. Windows Services
S3. Unix and Mac OS Services
S4. Backup Software
S5. Anti-virus Software
S6. Management Servers
S7. Database Software

Security Policy and Personnel:

H1. Excessive User Rights and Unauthorized Devices
H2. Phishing/Spear Phishing
H3. Unencrypted Laptops and Removable Media

Application Abuse:

A1. Instant Messaging
A2. Peer-to-Peer Programs

Network Devices:

N1. VoIP Servers and Phones

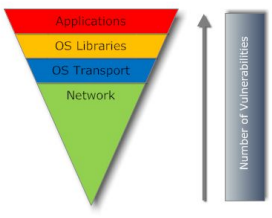
Zero Day Attacks:

Z1. Zero Day Attacks

http://www.sans.org/top-cyber-security-risks/

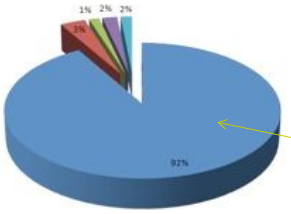
Top Cyber Security Risks

- Priority One: Client-side software that remains unpatched.
- Priority Two: Internet-facing web sites that are vulnerable.
- Operating systems continue to have fewer remotely-exploitable vulnerabilities that lead to massive Internet worms.
- Rising numbers of zero-day vulnerabilities



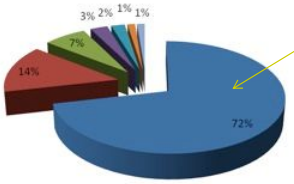
Application Vulnerabilities exceed OS Vulnerabilities

Microsoft OS Attack % For Vulnerabilities



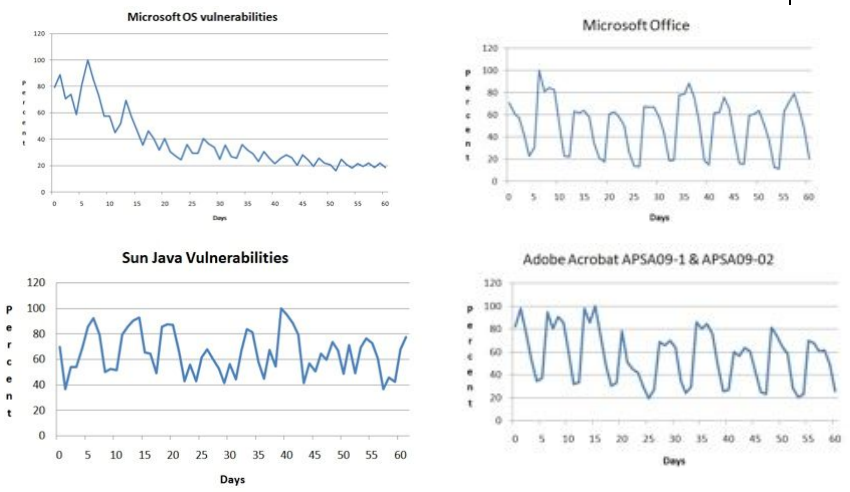
Conficker/ Downadup worm variants based on buffer overflow vulnerability

Apple Vulnerabilities Being Exploited



based on QuickTime

Application Patching is Much Slower than Operating System Patching



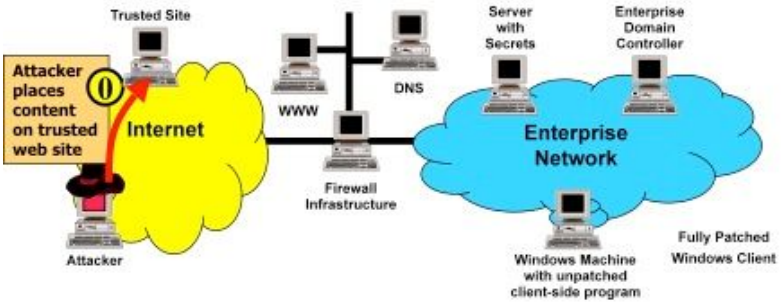
Real-Life HTTP Client-Side Exploitation Example

- Questa sezione illustra un esempio di attacco informatico reale
- In questo attacco, “Widget Acme Corporation” ha subito una grave violazione da pirati informatici che sono stati in grado di compromettere l'intera infrastruttura di rete interna utilizzando due dei più potenti vettori di attacco comunemente usati attualmente: sfruttamento delle lacune del software lato client e attacco *pass-the-hash* contro macchine Windows.

by http://www.sans.org/top-cyber-security-risks/

Step 0: Attacker Places Content on Trusted Site

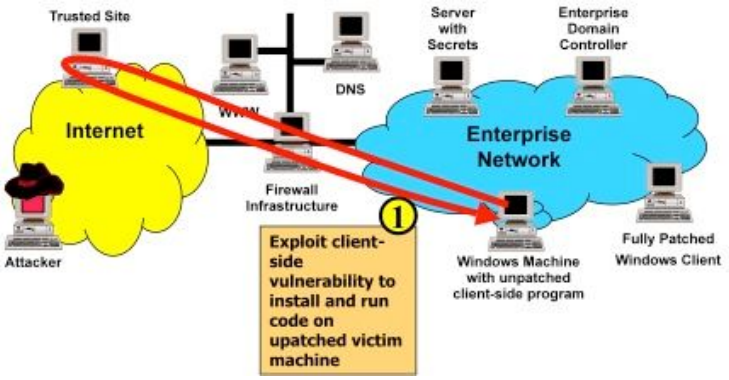
Nella Fase 0, l'attaccante comincia inserendo un contenuto malevolo in un sito web attendibile di terzi, come un social networking, blogging, o qualsiasi altro server web che ospiti contenuti postati da utenti pubblici. Il contenuto immesso dall'attaccante include codice per lo sfruttamento di software lato client non adeguatamente aggiornato.



Step 1: Client-Side Exploitation

Nella Fase 1, un utente sulla rete interna aziendale naviga in Internet da una macchina Windows con software lato client non adeguatamente aggiornato, come ad esempio un lettore multimediale (ad esempio, Real Player, Windows Media Player, iTunes, ecc) , un visualizzatore di documenti (ad esempio, Acrobat Reader), o un componente della suite di Office (ad esempio, Microsoft Word, Excel, Powerpoint, ecc.). Dopo aver caricato il contenuto malevolo dal sito, il browser dell'utente vittima invoca la vulnerabililtà del lato client del programma passandogli il codice exploit dell'attaccante. Questo codice permette all'attaccante di installare o eseguire programmi sulla macchina vittima, utilizzando i privilegi dell'utente che ha lanciato il browser. L'attacco è parzialmente limitato perché questo utente non dispone in generale di credenziali amministrative sul suo sistema. Tuttavia, l'attaccante può eseguire programmi con i privilegi utente.

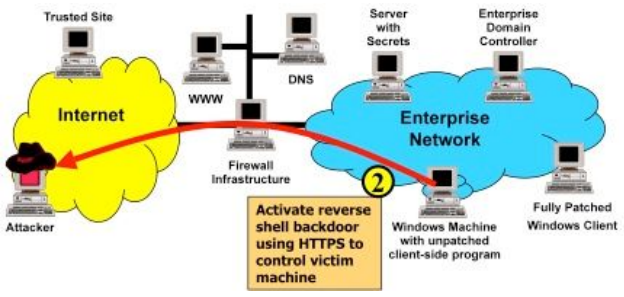
Step 1: Client-Side Exploitation



Step 2: Establish Reverse Shell Backdoor Using HTTPS

Nella Fase 2, il codice dell'attaccante installa un programma *backdoor back connection* sulla macchina vittima. Questo programma permette all'aggressore un accesso alla shell della macchina vittima, utilizzando l'accesso HTTPS dalla macchina attaccante a quella vittima attaccata.

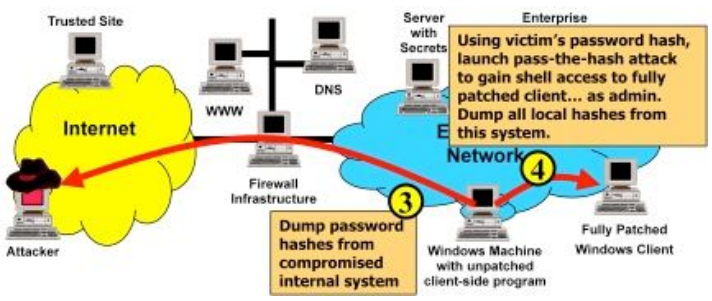
Il traffico *backdoor* sembra quindi essere regolare traffico cifrato web in uscita per quanto riguarda il firewall aziendale a cui la rete è collegata.



Steps 3 & 4: Dump Hashes and Use Pass-the-Hash Attack to Pivot

Nel passaggio 3, l'attaccante usa l'accesso alla shell del sistema vittima per caricare in locale un programma che sfrutti le debolezze della macchina per aumentare gradualmente i propri privilegi. Questo programma permette (o almeno cerca) al malintenzionato di passare su questa macchina dall'account utente con limitati privilegi a quello di totale controllo del sistema. Anche se i venditori spesso rilasciano patch per fermare questi attacchi di aumento di privilegi locali, molte organizzazioni non distribuiscono rapidamente tali patch, perché tendono a concentrarsi esclusivamente sui patch contro gli attacchi da remoto. L'attaccante ora può scaricare gli hash delle password per tutti gli account di questo computer locale, tra cui l'account di amministratore locale sul sistema.

Steps 3 & 4: Dump Hashes and Use Pass-the-Hash Attack to Pivot



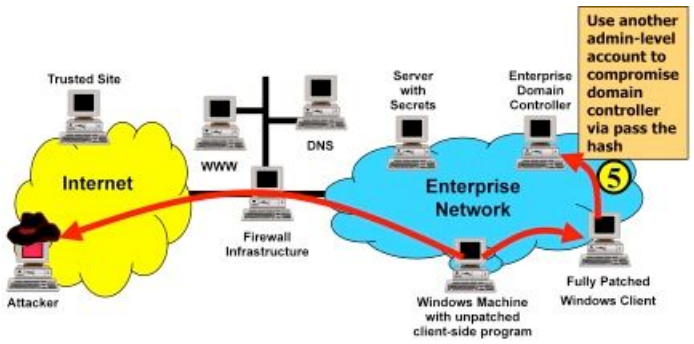
Steps 3 & 4: Dump Hashes and Use Pass-the-Hash Attack to Pivot

Nella fase 4, invece di violare ed utilizzare l'account di amministratore locale, l'attaccante utilizza un programma Windows *pass-the-hash* per scoprire l'autenticazione a un'altra macchina Windows sulla rete aziendale interna, un sistema client completamente aggiornato e protetto su cui il nuovo utente vittima ha privilegi amministrativi completi . Utilizzando NTLMv1 o NTLMv2, le macchine Windows si autenticano per l'accesso alla rete utilizzando Server Message Block (SMB), il protocollo basato sulle hash degli utenti e non sulle loro password, permettendo al malintenzionato di accedere al file system o eseguire programmi sul sistema completamente protetto con privilegi di amministratore locale. Utilizzando questi privilegi, l'attaccante ora scarica anche le hash delle password per tutti gli account locali di questa macchina Windows, che era regolarmente protetta.

Step 5: Pass the Hash to Compromise Domain Controller

Nel passaggio 5, l'attaccante utilizza una chiave hash della password di un account locale sul client Windows perfettamente protetto per accedere al sistema controller di dominio, di nuovo con un attacco pass-the-hash per ottenerne l'accesso alla shell. Poiché la password per l'account amministratore locale è identica alla password di un account amministratore di dominio, gli hash delle password per i due account sono identici. Pertanto, l'utente malintenzionato può accedere al controller di dominio con privilegi completi di amministratore di dominio, avendo il controllo completo su tutti gli altri account e le macchine presenti.

Step 5: Pass the Hash to Compromise Domain Controller



Steps 6 and 7: Exfiltration

Nella fase 6, con i privilegi completi di amministratore di dominio, l'attaccante compromette ora la macchina server che memorizza i dati riservati dell'organizzazione.

Nella fase 7, l'attaccante estrae ed analizza queste informazioni sensibili, composte generalmente da oltre 200 MB di dati. L'attaccante invia dal server questi dati attraverso Internet, sempre utilizzando HTTPS per criptare le informazioni, riducendo al minimo le probabilità di essere scoperto.

Outside Threats *(minacce esterne)*

- **Data theft** occurs when an unauthorized party or software program illegally obtains private information that is stored or transmitted on a network (packet sniffing and system break-ins).
- The **destruction of data** occurs when an unauthorized person or software program breaks into a system and deletes data.
- A **denial of service** (DoS) attack is designed to degrade server performance or remove it off the network completely.

Outside Threats

Several outside sources can cause attacks:

- **Hackers** - the true hacker desires to dissect systems and programs to see how they work.
- **Crackers** - those that break in to computer systems to tamper with, steal, or destroy data.
- **Virus** - it causes some unexpected and usually undesirable event.
- **Worms** - a self-replicating virus that does not alter files but resides in active memory and duplicates itself.
- **Trojan horse** - is a program that presents itself as another program to obtain information

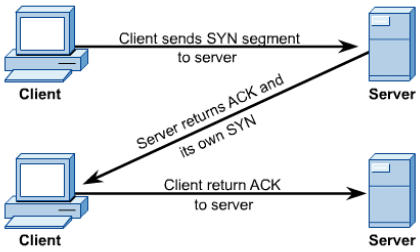
Denial of Service (DoS)

- A DoS attack occurs when the targeted system cannot service legitimate network requests effectively.
- As a result, the system has become overloaded by illegitimate messages.
- DoS attacks originate from one host or a group of hosts.
- When the attack comes from a coordinated group of hosts, such attacks are called **Distributed DoS** (DDoS).
- A common DoS attack is to overload a target system by sending more data than it can handle.

Denial of Service (DoS)

There are several specific types of DoS attacks:

- A **buffer overflow** attack is designed to overwhelm the software running on the target system (buffer overflow).
- The so-called **ping of death** is a well known buffer overflow. The attacker send an ICMP echo request that are illegally large to the target. (specific weaknesses in NOS software)
- The **TCP synchronization** (SYN) attack exploits the TCP protocol three-way handshake
  - the attacker sends a large volume of TCP synchronization SYN requests without replying to the ACKs (resulting high volume of half-open connections).



Denial of Service (DoS)

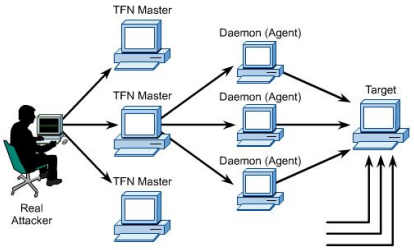
- In 1997, a variation of the TCP SYN attack, called **land**, was discovered. The land attack uses a program that alters the IP header of the SYN request (spoofing or forging), making the request to be sourced from the target itself.
- Network devices can be configured to block TCP SYN attacks from a single source, increasing the number of half-open connections or decreasing the amount of waiting time for the reply.
- **Teardrop** (or *fragment attack*) is the name of a program that takes advantage of the way IP handles fragmentation, sending fragments with overlapping reassembly information which confuse the target software.

Denial of Service (DoS)

- the **Smurf** attack relies on spoofing the IP packet header. Normally the external ping requests are denied and internal are allowed. Spoofing the IP header, external ping requests can reach and flood and overload the internal network.
- The examples listed here are all well known vulnerabilities. OS software is now written with these attacks in mind. For example, most systems are now immune to land and Teardrop.
- Known vulnerabilities in software represent holes in the system. These holes can be repaired, or patched, by installing software updates when they are made available by a vendor.

Distributed Denial of Service (DDoS)

Tribal flood network DDos attack



- Before the hacker can attack the ultimate target, a "fleet" of "zombies" (unsecure host with a permanent Internet connection) must be coordinated for the attack.
- The hacker takes advantage of the zombie's lack of security.
- The hacker breaks in to the system either directly or through an e-mail virus.
- The goal of the break in or virus is to install software on the zombie system.
- The hacker uses the zombies to launch a DDoS attack on the ultimate target.

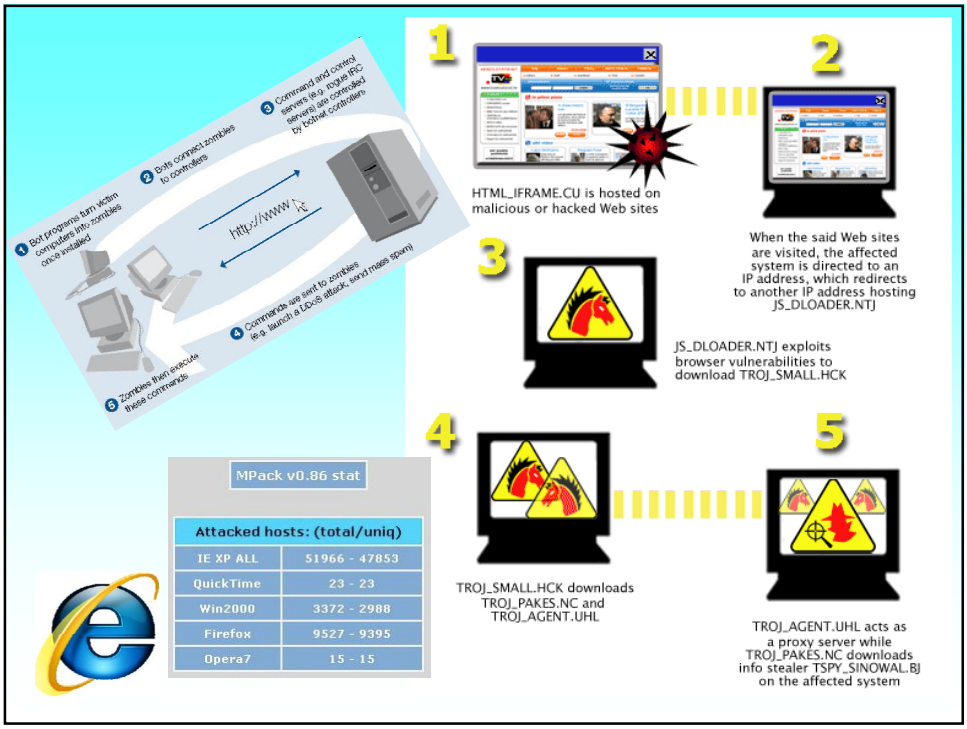
Well Known Exploits

- Each combination of NOS and application software contains it’s own unique set of vulnerabilities and weaknesses.
- Threats to network security comes from individuals with sophisticated tools (and with relatively weak technical abilities).
- Some of these individuals are often called "script kiddies".
- Script kiddy is a negative term used to describe immature individuals that use scripts, software programs, or techniques created by other, more skilled crackers.

Well Known Exploits

Well Known Exploits

- **Asmodeus (NetIQ security analyzer)** - Network security analyzer and port scanner for Windows that is capable of scanning ranges of hosts for remote security vulnerabilities.
- **SATAN (Security Administrator Tool for Analyzing Networks)** - An outdated network security analyzer for UNIX, similar in function to NetIQ.
- **Saint (The Security Administrator's Integrated Network Tool)** -An updated and enhanced version of SATAN.
- **Strobe (strobe-classb)** - A small but fast scanner, used to scan for open mail relays over class B networks.
- **Ogre** - Service and vulnerability scanner for Windows NT, including NetBIOS shares and some Microsoft Internet Information Services (IIS) vulnerabilities.
- **mscan (Multiscan)** - Scanner used to detect vulnerabilities in commonly used UNIX services, such as DNS, NFS, statd, X and finger.
- **Nmap** - A fast and powerful port scanner for UNIX, capable of scanning ranges of computers via IP address, domain, or randomly for open ports, operating system guess, and other information.
- **ncat (Network Config Audit Tool)** - Utility for scanning Cisco IOS Config files for user defined parameters, such as oversights or errors.
- **BackOffice** - A server that runs in the background of the installed computer, waiting for clientconnections to remotely administer the system, invisible to regular users.
- **NetBus** - Same thing as BackOffice but made by different people. Its not as powerful and is usually attached onto an unrelated executable.
- **SubSeven** - Same as BackOffice and NetBus, but similar to BackOffice in power.
- **trinoo, Stacheldraht, tribe flood network (TFN), mstream, carko, wormkit** - DDoS tools.
- **Ramen** - A collection of tools designed to attack systems by exploiting well-known vulnerabilities in three commonly installed software packages. A successful exploitation of any of the vulnerabilities results in a privileged root compromise of the victim host.

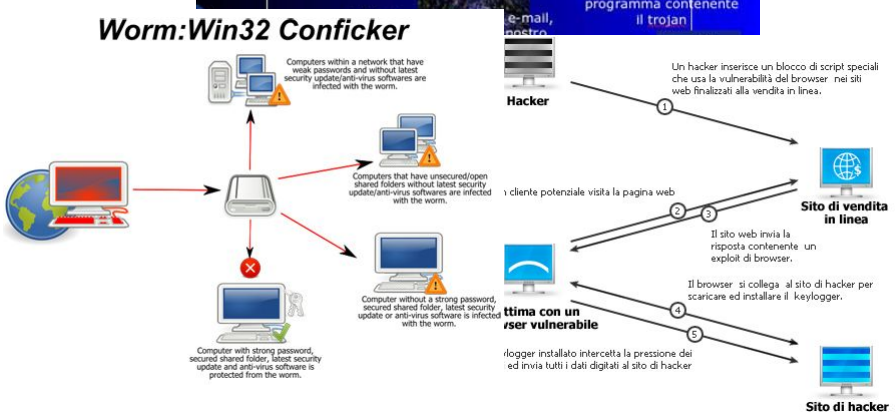


Grave vulnerabilità in Samba

Si consiglia l'aggiornamento alla versione successiva, o di applicare il workaround descritto in questo articolo.

Risale a martedì 10 aprile il bollettino di sicurezza del team di sviluppo di **Samba**, che **denuncia un grave bug** nella nota suite di interoperatibilità.

Le versioni 3.6.3 e precedenti sono tutte afflitte da



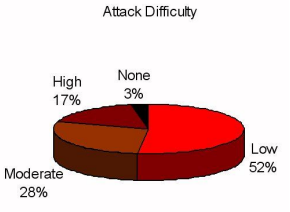
Inside Threats

- Security threats that originate from inside a network can be more harmful than outside threats.
- High profile inside threats include disloyal and disgruntled employees who use their inside access to destroy, steal, or tamper with data.
- Corporate espionage is the most sophisticated type of internal security threat. Employees can be approached by competing companies.
- Internal security breaches can also be the result of rebellious users who disagree with security policies.
- A growing problem for corporate networks is the widespread popularity of instant messaging and peer-to-peer file sharing. Chat and file sharing applications may be vulnerable to other forms of exploitation.

Inside Threats

In addition to a well-planned security policy, organizations should provide **training programs for all employees** who use the computer network. Since employees are often targeted as a way into the Intranet, it is important to instruct them on how to prevent viruses, DoS attacks, data theft, and so on. Damage is more likely to occur out of ignorance, not incompetence.

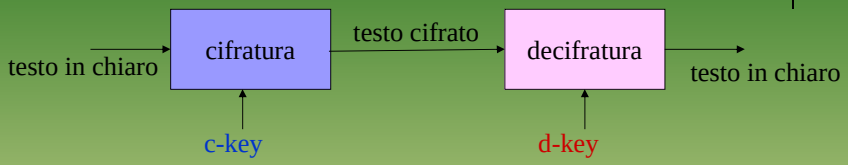
Implementing Security Measures



File Encryption, auditing, and authentication

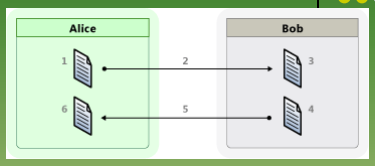
- File encryption is a way of encrypting data stored on a hard disk so that it is unreadable to anyone but the creator of the data.
- 3DES: is a cryptosystem which can encrypt and decrypt data using a single secret key
- DES is a block cipher, acts on a fixed-length block of plaintext (64 bits) and converts it into a block of ciphertext of the same size by using the secret key (also 64 bits but 8 bits for parity).
- The effective key length in DES is only 56 bits. In 3DES, 3 stages of DES with a separate key for each stage is applied. So the key length in 3DES is 168 bits.
- DES is known as a symmetric key cipher because the same key is used both in encryption and decryption.
- (DES, 3DES, AES = symmetric), (RSA, RC4, IDEA = asymm)

Crittografia

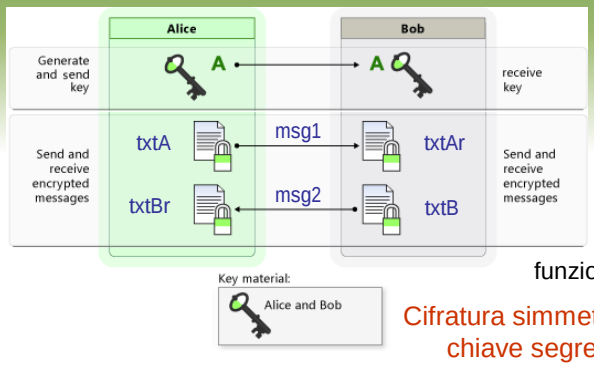


- Gli algoritmi di cifratura/decifratura sono pubblici
- Le chiavi di cifratura/decifratura sono segrete
- Cifratura simmetrica
  - c-key = d-key
  - *problemi di sicurezza nella distribuzione e condivisione*
- Cifratura asimmetrica o a chiave pubblica
  - c-key != d-key
  - *una chiave è pubblica, l'altra è segreta*

Crittografia



Trasmissione in chiaro



msg1 = Cif(txtA, A)
txtAr = Dec(msg1, A) =
= Dec(Cif(txtA,A), A)

msg2 = Cif(txtB, A)
txtBr = Dec(msg2, A) =
= Dec(Cif(txtA,A), A)

funziona se Dec(Cif(txt,k),k) = txt

Cifratura simmetrica (a chiave segreta)

Crittografia

Cifratura simmetrica

Molti sono i possibili schemi di cifratura utilizzati: DES, 3DES, AES, RC4, IDEA, …

In generale, la forza di uno schema di cifratura dipende dalla lunghezza della sua chiave (perchè la ricerca della chiave risulta più complessa)

- DES (data encryption standard, 1975) usa una chiave lunga 56 bit; divenuta vulnerabile per ricerca esaustiva della chiave
- Sostituito nel 2002 da AES (advanced encryption standard, 1998) che usa chiavi lunghe 128, 192, o 256 bit

$A \rightarrow B : \{ D \}K_{AB}$ $D = \{ \{ D \}K_{AB} \}K_{AB}$

Crittografia

Cifratura asimmetrica

- Ciascun utente ha una chiave pubblica K e una chiave privata K^{-1}
- K^{-1} deve essere mantenuta segreta, non va comunicata a nessuno, e non può essere dedotta da K
- K è pubblica e disponibile a tutti
- La cifratura e decifratura con le chiavi K e K^{-1} sono commutative:

$$\{ \{D\}K^{-1} \}K = \{ \{D\}K \}K^{-1} = D$$

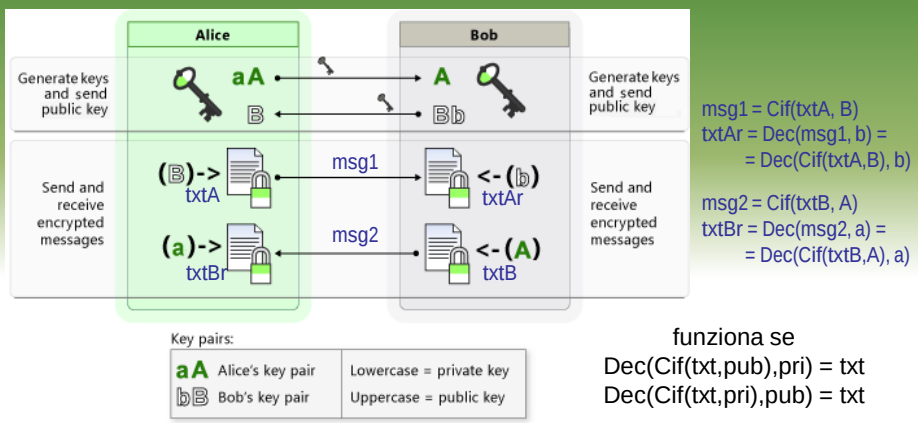
Crittografia

- La cifratura asimmetrica è 3-5 ordini di grandezza più lenta della cifratura simmetrica
- Si utilizza la cifratura asimmetrica per scambiare solo le chiavi simmetriche, mentre i dati vengono scambiati con schemi simmetrici:

$$A \rightarrow B: \{K_{AB}\}K_B, \{D\}K_{AB}$$

- Inoltre la cifratura asimmetrica ha altri importanti utilizzi nel campo dell'autenticazione

Crittografia



msg1 = Cif(txtA, B)
txtAr = Dec(msg1, b) =
= Dec(Cif(txtA,B), b)

msg2 = Cif(txtB, A)
txtBr = Dec(msg2, a) =
= Dec(Cif(txtB,A), a)

funziona se
Dec(Cif(txt,pub),pri) = txt
Dec(Cif(txt,pri),pub) = txt

Cifratura asimmetrica (chiave pubblica e privata) con algoritmo RSA

Crittografia

Come ricavare K e K^{-1}?

Si sfruttano le proprietà dei moduli: come $(x^a)^b = (x^b)^a$, nei moduli se $C=Y^A \pmod P$ e $D=Y^B \pmod P$ allora $C^B \pmod P = D^A \pmod P$.

Inoltre si usano funzioni suriettive con un numero di possibili inversi di complessità esponenziale, come ad esempio l'elevamento a potenza e l'esponenziale con i moduli.

Crittografia

L'algoritmo RSA garantisce l'impossibilità di ricavare la chiave privata da quella pubblica. Le chiavi sono lunghe da 256 a 2048.

Costruzione delle chiavi

- Si prendono due numeri primi, ***p*** e ***q***,
- Si calcolano ***n = p · q*** e ***z = (p-1) · (q-1)***
- Si sceglie un numero ***d*** che sia primo rispetto a ***z*** (cioè che non abbia divisori comuni con ***z)***.
- Si trova ***e*** tale che ***(e · d) (mod z) = 1***
- La chiave pubblica è data da ***pri = (e , n)***
- La chiave privata è data da ***pri = (d , n)***

Esempio elementare

p=3 q=11

n = (p)·(q) = 33

z = (p-1)·(q-1) = 20

d = 3

e = 7

Privata = (7, 33)

Pubblica = (3, 33)

Crittografia

Codifica del messaggio

- Suddividere il testo in chiaro in blocchi da **k** bit, con k il più grande intero tale che $2^k<n$,
- Per ogni blocco in chiaro B, il codice cifrato C vale

$$C = B^e \pmod n$$

Decodifica del messaggio

- Per ogni blocco cifrato C, il blocco in chiaro ricevuto R vale

$$R = C^d \pmod n$$

Esempio elementare

messaggio: 1308

k = 5 bit (da 0 a 31)

$B_1 = 13$ $B_2 = 08$

Privata = (7, 33)

Pubblica = (3, 33)

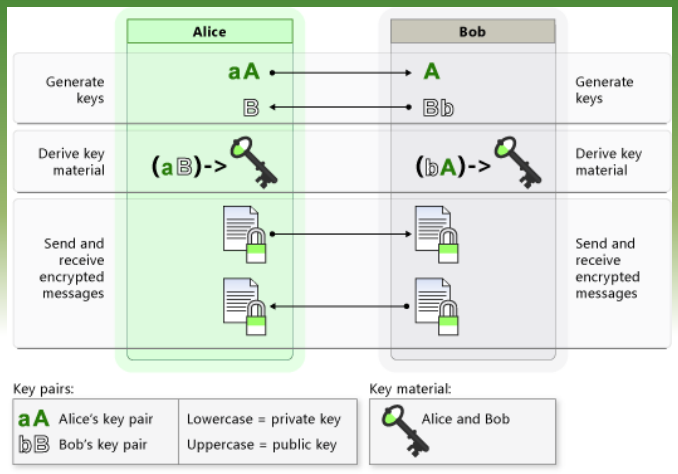
$C_1 = 13^3 \pmod{33} = 19$

$C_2 = 08^3 \pmod{33} = 17$

$R_1 = 19^7 \pmod{33} = 13$

$R_2 = 17^7 \pmod{33} = 8$

Crittografia

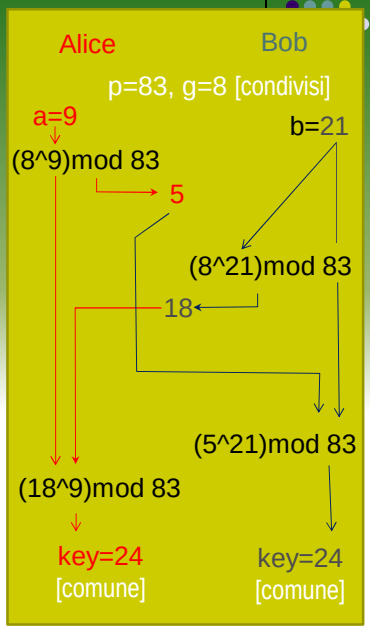


Trasmissione a chiave pubblica e privata con algoritmo ECDH
Elliptic Curve Diffie-Hellman

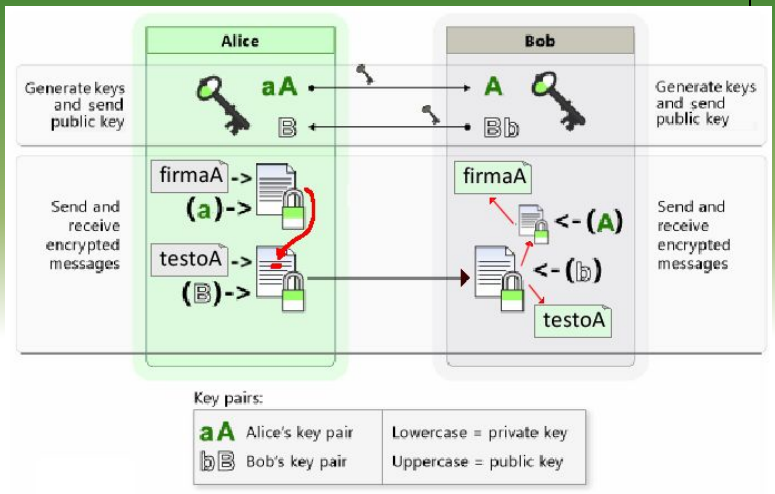
Crittografia

Algoritmo ECDH

- Alice e Bob concordano di utilizzare un numero primo **p** e un intero di base **g**
- Alice sceglie un intero segreto **a** ed invia a Bob **(g^a) mod p**
- Bob sceglie un intero segreto **b**, ed invia ad Alice **(g^b) mod p**
- Alice calcola la chiave
  (((g^b) mod p)^a) mod p
- Bob calcola la chiave
  (((g^a) mod p)^b) mod p
- Le due chiavi coincidono perché g^(ab) = g^(ba). Questo valore corrisponde alla chiave privata condivisa.



Crittografia



Trasmissione **autenticata** a chiave pubblica e privata con algoritmo RSA

File Encryption, auditing, and authentication

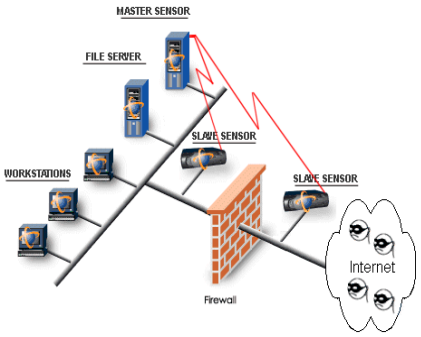
- Windows 2000 includes a file encryption function.
- Third party encryption programs are available for OSs:
  - PC Guardian, Deltacrypt, Winzap
- Authentication provides several methods of identifying users including the following:
  - Login and password dialog
  - Challenge and response
  - Messaging support
- A Linux server can use CHAP and PAP authentication for PPP connections.
  - PAP (Password Authentication Protocol)
  - CHAP (Challenge Handshake Authentication Protocol) uses pre-shared secret key (in /etc/ppp/chap-secrets) and periodical challenge requests

File Encryption, auditing, and authentication

- **Auditing** - relates to the computer and networking world is software that runs on a server and generates a report showing who has accessed the server and what operations the users have performed during a given period of time.
- A free in easy to install auditing software is LSAT (Linux Security Auditing Tool). It checks many system configurations and local network settings on the system for common security and configuration errors and for packages that are not needed. It is a post install security auditing tool.

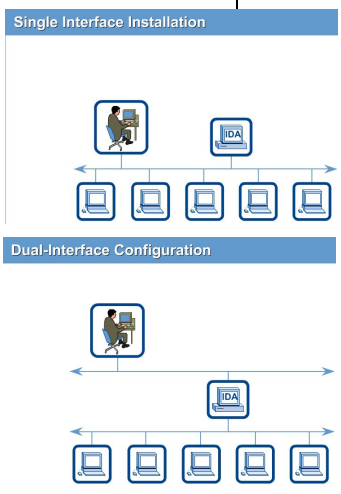
Intrusion Detection Systems

- An **Intrusion Detection System** (IDS) is hardware or software that is responsible for detecting inappropriate, unsuspected, or other data that may be considered unauthorized that is occurring on a network.
- An IDS is different than a firewall: the firewall limits the access based on a set on rules, IDS inspects all traffic and evaluates a suspected intrusion, generating an alarm.
- **Snort** - is a software-based real-time network IDS that can be used to notify an administrator of an intrusion attempt.



Intrusion Detection Systems

- Snort can be installed on a server with either a single or dual network interface.
- On a single interface installation the same interface listens to the network traffic and allows the management
- Snort detects intrusions according to a set of rules.
- The **rules.base** file contains the information for the INTERNAL and EXTERNAL networks and DNS servers from which tend to trigger the portscan detection will need to be entered.

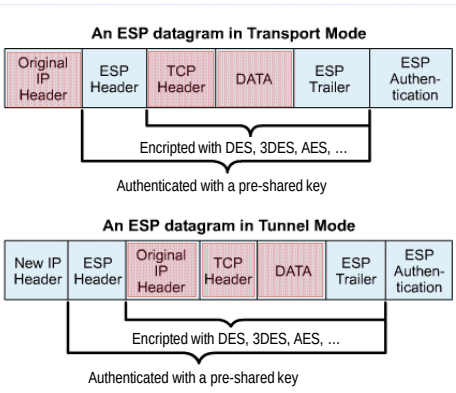


Intrusion Detection Systems

- **PortSentry** - is a port scan detector that can be configured to bind to ports you want monitored.
- When it finds one it can react in the following ways:
  - A log indicating the incident is made via syslog().
  - The target host is automatically dropped into /etc/hosts.deny for TCP Wrappers.
  - The local host is automatically re-configured to route all traffic to the target to a dead host to make the target system disappear.
  - The local host is automatically re-configured to drop all packets from the target via a local packet filter.
- The purpose of this is to give an admin a heads up that their host is being probed.

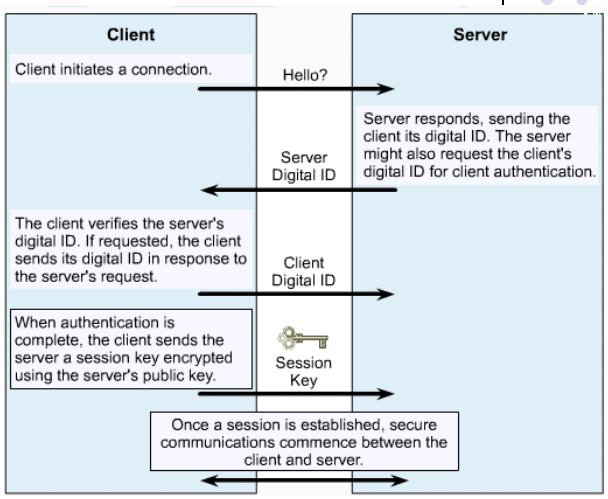
IP Security

- **IPSec** secures data at the packet level. So it works at the network layer of the OSI model and applications are not aware on it.
- The IPSec suite include the protocols AH and ESP. They can be used separately or together.
- The **Authentication Header** (AH) enables verification of the sender identity.
- **Encapsulating Security Payload** (ESP) ensures authentication and the confidentiality of the data itself.
- IPSec can operate in either the transport mode (host-to-host) or the tunnel mode (gateway-to-gateway).



Secure Sockets Layer (SSL)

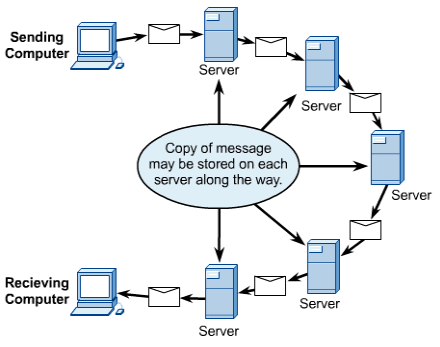
- SSL was developed by Netscape to provide security for its web browser.
- It uses public and private key encryption.
- SSL operates at the application layer and must be supported by the user application.



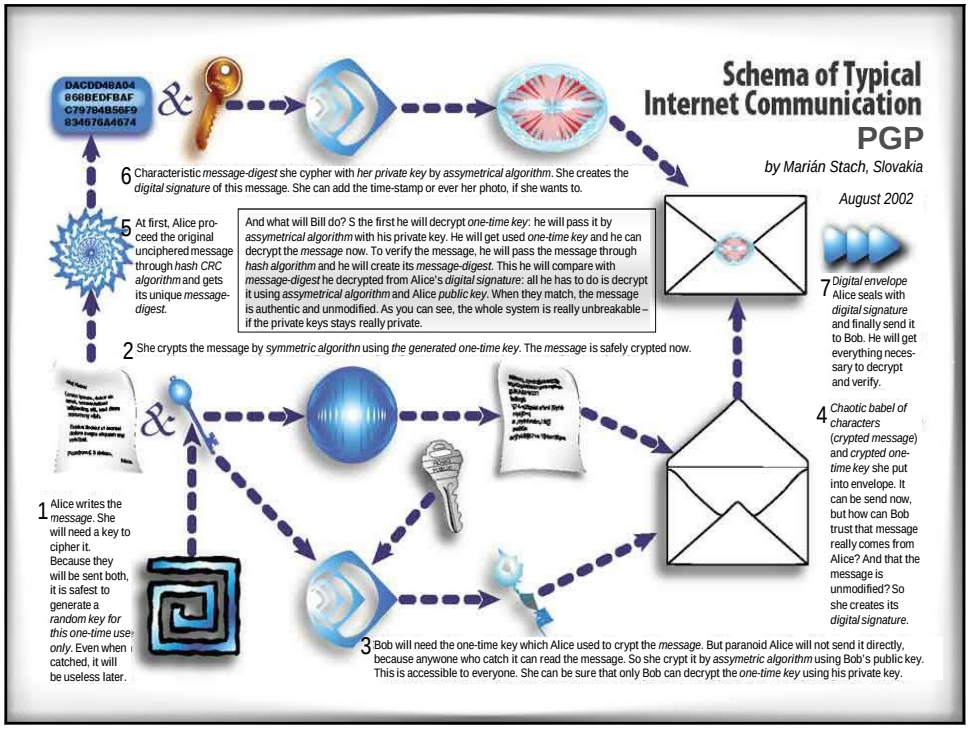
SSL (Secure Socket Layer) ≠ SSH (Secure Shell)

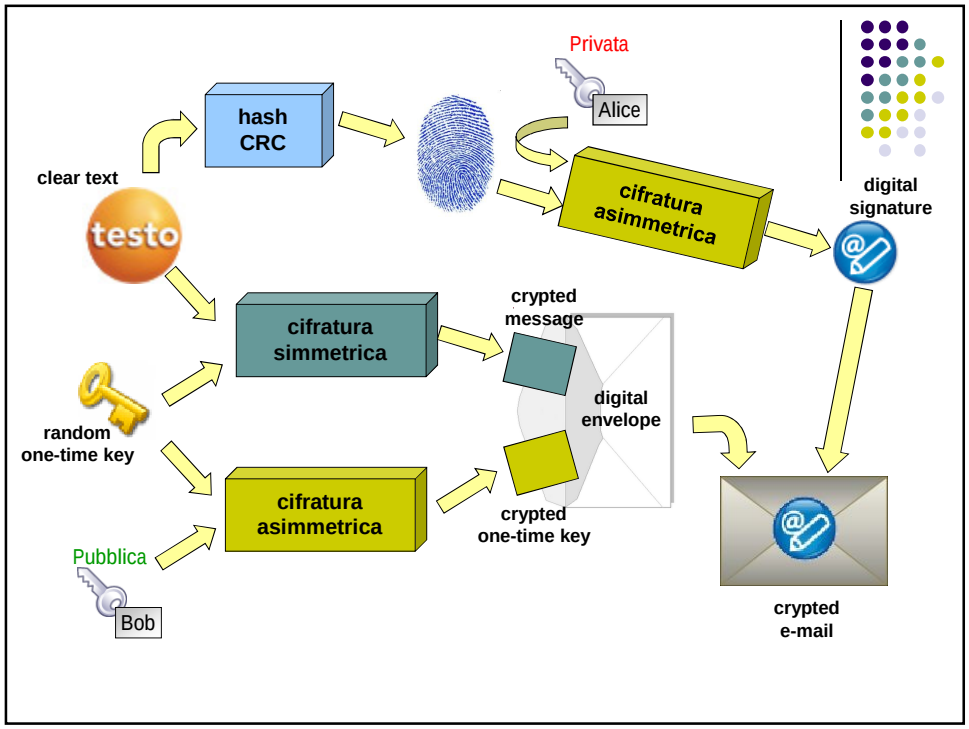
E-mail Security

- E-mail users think they have the same expectation of privacy when sending e-mail as they do when sending a letter through the postal service.
- A more accurate expectation would be to assume that the e-mail is like a postcard that can be read by anyone who handles it during its journey from sender to recipient.



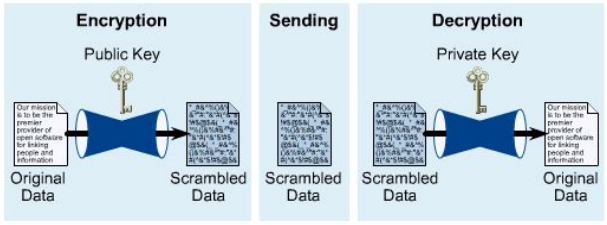
- They often travel through dozens of nodes or servers on their way from sender to recipient.
- The most popular e-mail protection programs are PGP (Pretty Good Privacy), Kerberos, Fire Trust and MailMarshal


Schema of Typical Internet Communication
PGP
by Marián Stach, Slovakia
August 2002
DACDD48A04
868BEDFBAF
C79784B56F9
834676A4674
6 Characteristic message-digest she cypher with her private key by assymetrical algorithm. She creates the digital signature of this message. She can add the time-stamp or ever her photo, if she wants to.
5 At first, Alice proceed the original unciphered message through hash CRC algorithm and gets its unique message-digest.
And what will Bill do? S the first he will decrypt one-time key: he will pass it by assymetrical algorithm with his private key. He will get used one-time key and he can decrypt the message now. To verify the message, he will pass the message through hash algorithm and he will create its message-digest. This he will compare with message-digest he decrypted from Alice's digital signature: all he has to do is decrypt it using assymetrical algorithm and Alice public key. When they match, the message is authentic and unmodified. As you can see, the whole system is really unbreakable – if the private keys stays really private.
7 Digital envelope Alice seals with digital signature and finally send it to Bob. He will get everything necessary to decrypt and verify.
2 She crypts the message by symmetric algorithm using the generated one-time key. The message is safely crypted now.
4 Chaotic babel of characters (crypted message) and crypted one-time key she put into envelope. It can be send now, but how can Bob trust that message really comes from Alice? And that the message is unmodified? So she creates its digital signature.
1 Alice writes the message. She will need a key to cipher it. Because they will be sent both, it is safest to generate a random key for this one-time use only. Even when catched, it will be useless later.
3 Bob will need the one-time key which Alice used to crypt the message. But paranoid Alice will not send it directly, because anywone who catch it can read the message. So she crypt it by assymetric algorithm using Bob's public key. This is accessible to everyone. She can be sure that only Bob can decrypt the one-time key using his private key.


Privata
Alice
hash
CRC
clear text
testo
cifratura
asimmetrica
digital
signature
cifratura
simmetrica
crypted
message
digital
envelope
random
one-time key
cifratura
asimmetrica
crypted
one-time key
Pubblica
Bob
crypted
e-mail

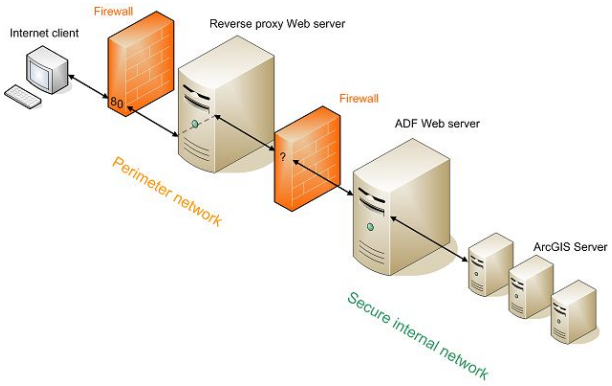
Public/Private Key Encryption

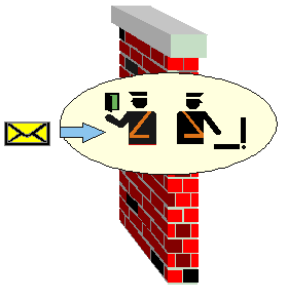
- One key is published and is widely available.
- The other key is private and known only to the user.
- Both keys are required to complete the secure communication.
- This type of encryption, is also referred to as asymmetric encryption.
- With this type of encryption, each user has both a public and a private key, called a key pair.



Arriva il CISPA, benvenuti nel regno del Grande Fratello

Firewalls



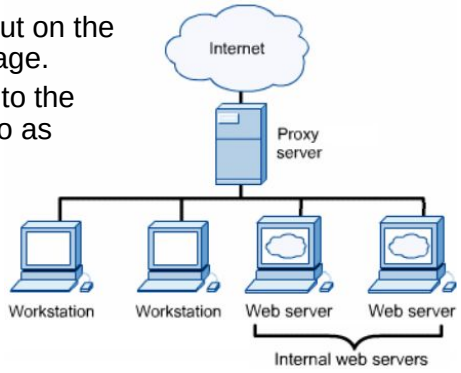


Introduction to Firewalls and Proxies

- A firewall is specialized software, hardware, or a combination of the two, whose purpose is to prevent unwanted or malicious IP packets from reaching a secure network.
- Typically, an Internet firewall is a host running IP packet filtering software, accordin to a specific set of rules.
- For example, a packet matching a particular source address can be dropped, forwarded, or processed in some special way.
- Early firewalls filtered packets based on addressing information. Today common matching criteria are:
  - IP address, both source and destination
  - TCP/UDP port number, both source and destination
  - Upper layer protocols, HTTP, FTP, and so on.
- This approach is also referred to as **rules-based forwarding**.

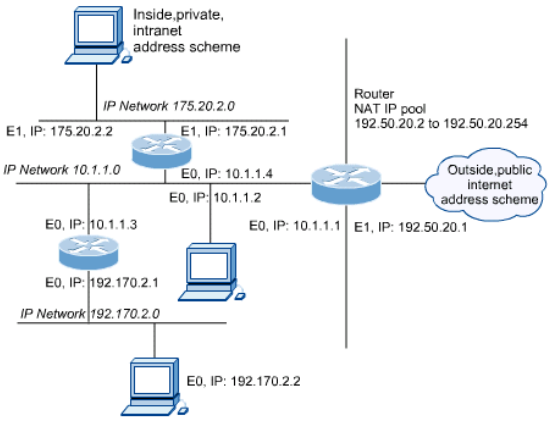
Introduction to Firewalls and Proxies

- A **proxy** is software that interacts with outside networks on behalf of a client host.
- Typically, client hosts on a secure LAN request a web page from a server running proxy services.
- The proxy server then goes out on the Internet to retrieve the web page.
- The web page is then copied to the proxy server, this is referred to as **caching**.
- Finally, the proxy server transmits the web page to the client.



Introduction to Firewalls and Proxies

- Administrators use Network Address Translation (NAT) to alter the source address of packets originating from a secure LAN.
- This allows secure LANs to be addressed using private IP addresses.
- Private IP addresses are not routed on the Internet.
- An outside hacker cannot directly reach a computer with a private address.
- Some experts make a distinction between NAT and a firewall. Others look at NAT as included on firewalls.



Packet Filtering

- The most basic firewall solution is an IP packet filter.
- To configure a packet filter, a network administrator must define the rules that describe how to handle specified packets.
- The first packet filters the filtered packets based on the addressing information contained in the packet header (**layer 3**)
- Later, packet filters were designed to base decisions on information contained in the TCP or UDP header at **layer 4**.

```
#Clear all rules
/sbin/ipfw -f flush

#Deny Routing Information Protocol on UDP port 520
/sbin/ipfw add deny udp from any 520 to any 520 via xl0

#Send all packets to the NAT Daemon for address translation
/sbin/ipfw add divert and all from any to any via xl0

#Allow specific hosts access
/sbin/ipfw add allow ip from 172.17.4.5 to any
/sbin/ipfw add allow ip from 172.17.87.52 to any

#Allow Web Requests
/sbin/ipfw add allow tcp from any to 192.168.54.198 80

#Deny everything else to 192.168.54.0/24 network
/sbin/ipfw add deny ip from any to 192.168.54.0/24

#Permit the rest
/sbin/ipfw add permit ip from any to any
```

Packet Filtering

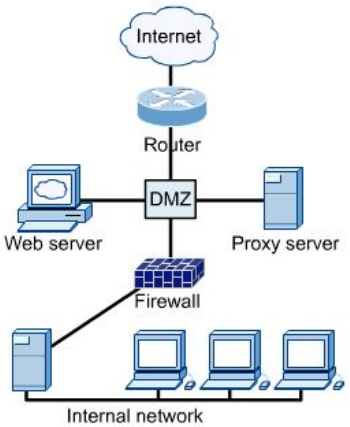
- **Layer 4 access lists** can be configured to permit or deny packets.
- The firewall can also be configured to examine the **TCP code bits**, analysing the three-way handshake and keeping uninvited traffic out (*connection traffic filtering*).
- Firewall must guess at what connectionless traffic (ex: IP, ICMP, UTP) is invited and what connectionless traffic is not.
- ICMP ping does not use a layer 4 header. There is no way to determine if a datagram is part of an established connection. This is because IP and UDP are both connectionless.
- Intelligent firewalls monitor the connectionless traffic noting the IP and the UDP source and destination ports in a table, and detecting connectionless traffic that looks like it is invited (**stateful packet filtering**). Such firewalls are dynamic.

Packet Filtering

- The most comprehensive form of packet filtering examines layer 3 and 4 headers and the layer 7 application data as well.
- Layer 7 firewalls look for patterns in the payload of the packet.
- This is done in an effort to determine what application is being used, such as HTTP, FTP, and so on (**layer 7 stateful packet filtering**).
- They work with software that is preprogrammed to recognize a given application. Therefore, not all applications will be supported.
- This type of packet filtering adds a significant amount of delay and overhead to the routing process.

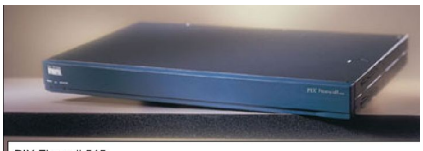
Firewall Placement

- A boundary router connects the enterprise LAN to its ISP or the Internet.
- The boundary router should only allow HTTP, FTP, mail, and DNS related traffic to the DMZ.
- The DMZ is designed to keep the inside network clean.
- The NOS servers in the DMZ should be tightly configured.



Common Firewall Solutions

- An appliance is a device that is stand-alone and easy to configure
- The most popular firewall appliance is the Cisco PIX. It include NAT and stateful packet filtering.
- The PIX Firewall 515 uses TFTP for image download and upgrade.
- It has a low profile design, 128,000 simultaneous sessions, and 170 Mbps throughput.
- Default settings allow all connections from the inside interface access to the outside interface, and block all the connection from outside to inside.



PIX Firewall 515
- Low profile design
- 128,000 simultaneous sessions
- 170 Mbps thru-put

Cisco ASA 5500 Series
- da 4.000 a 150.000 simultaneous sessions
- da 150 Mbps a 10 GBbps throughput
- IPSec VPN capability

Using a NOS as a Firewall

- In high-traffic environments, a specialized packet filtering and NAT solution is recommended.
- A device such as a router or firewall appliance is designed to switch packets and manipulate them quickly.
- A NOS running on ordinary hardware may be able to do the same job. However, it is not without **adding latency** and overhead on the server.
- In low traffic environments, such as small offices and home networks, a NOS firewall solution is a good choice.
- Linux can use **ipchains** and **iptables** to act as a gateway between a private network and the internet, thereby providing firewall capabilities.

Using a NOS as a Firewall

- **netfilter** è un componente del kernel di Linux che permette l'intercettazione e la manipolazione dei pacchetti che attraversano il computer (svolge funzioni di firewall).
- **iptables** è il programma che permette di configurare netfilter.
- iptables raggruppa tutti i controlli che può fare sul traffico in entrata nella cosiddetta *INPUT Chain*. I controlli sul traffico in uscita sono invece raggruppati nella *OUTPUT Chain*. La *FORWARD Chain* serve per il traffico non indirizzato a noi ma che comunque passa per il nostro computer.
- Ognuna di queste catene ha una *policy*, cioè un'azione predefinita da eseguire quando tutti gli altri controlli della catena hanno fallito nel riconoscere se il dato era buono o meno.

Using a NOS as a Firewall

- I valori di policy possono essere:

  ACCEPT lascia passare il pacchetto
  DROP scarta il pacchetto
  QUEUE dirotta il pacchetto nello spazio utente per un'analisi succ.
  RETURN viene eseguita la regola di default della catena
- Una catena (che ha la forma di una Access Control List) è composta da una serie di regole suddivise in

  match condizione da verificare
  target azione da intraprendere se il pacchetto soddisfa il match
- La policy, detta anche regola di **default**, viene eseguita quando nessun match precedente è verificato

Using a NOS as a Firewall

```
$ sudo iptables -L                    list the rules in the chains
```

```
Chain INPUT (policy ACCEPT)
target prot opt source       destination
Chain FORWARD (policy ACCEPT)
target prot opt source       destination
Chain OUTPUT (policy ACCEPT)
target prot opt source       destination
```

- Di default, iptables lascia passare tutto, quindi per prima cosa blocchiamo il traffico:

```
$ sudo iptables –P INPUT DROP
$ sudo iptables –P FORWARD DROP      change the policy in a chain
```

```
Chain INPUT (policy DROP)
target prot opt source       destination
Chain FORWARD (policy DROP)
target prot opt source       destination
```

Using a NOS as a Firewall

- consentiamo tutto il traffico interno al nostro computer, che passa per l'interfaccia di loopback (**lo**).

```
$ sudo iptables -A INPUT -i lo -j ACCEPT      append a rule to a chain
```

–A INPUT aggiunge una nuova regola alla catena INPUT
-i lo il nome dell'interfaccia da cui ricevere i pacchetti
-j ACCEPT l'obiettivo della regola

- consentiamo la navigazione e il traffico da noi richiesto

```
$ sudo iptables -m state --state ESTABLISHED,RELATED -j ACCEPT
```

–m state aggiunge una nuova regola alla catena INPUT
--state x,y se i pacchetti sono nello stato ESTABLISHED (associato ad una connessione) o RELATED (nuova connessione associata ad un'altra)
-j ACCEPT l'obiettivo della regola

Using a NOS as a Firewall

- consentiamo il traffico entrante di tipo SSH (porta 22)

```
$ sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT
```

`–p tcp` il protocollo da controllare

`--dport 22` il port di destinazione da controllare

- consentiamo l'accesso ad un server web interno

```
$ sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
```

- scartiamo il traffico ICMP proveniente dall'host 88.11.22.33

```
$ sudo iptables -A INPUT -p icmp –s 88.11.22.33 -j DROP
```

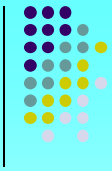
Using a NOS as a Firewall

- vediamo la situazione risultante:

```
$ sudo iptables –vv -L
```

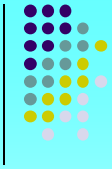
```
Chain INPUT (policy DROP)
target prot opt in  out source   destination
ACCEPT all  --- lo  any anywhere anywhere
ACCEPT all  --  any any anywhere anywhere state RELATED, ESTABLISHED
ACCEPT tcp  --  any any anywhere anywhere tcp dpt:ssh
ACCEPT tcp  --  any any anywhere anywhere tcp dpt:www
DROP   icmp --  any any 88.11.22.33  anywhere

Chain FORWARD (policy DROP)
target prot opt in  out source   destination

Chain OUTPUT (policy ACCEPT)
target prot opt in  out source   destination
```

Using a NOS as a Firewall



- altre opzioni del comando iptables
  - -N crea una nuova catena
  - -X rimuove una catena vuota
  - -P cambia la policy di una catena
  - -L elenca le regole di una catena (o di tutte)
  - -Z azzera i contatori di pacchetti e di byte di tute le regole
  - -A aggiunge una nuova regola in coda alla catena
  - -I aggiunge una nuova regola in un dato posto della catena
  - -R sostituisce la regola posizionata in un dato posto della catena
  - -vv output di tipo verbose

ufw: iptables and Ubuntu



- abilitiamo l'uso di ufw

```
$ sudo ufw enable
```

- per aprire il servizio in una porta (SSH)

```
$ sudo ufw allow 22
```

- per consentire l'accesso ad un server web interno

```
$ sudo ufw allow 80
```

- per scartare il traffico ICMP proveniente da 88.11.22.33

```
$ sudo ufw drop proto icmp from 88.11.22.33 to any port
```