Information security

- **2 kinds of machines**
  - Standalone
  - Networked
- **2 kinds of data**
  - Machine generated (sensors, DB) security=reliability
  - Human generated... The most dangerous ;-)
- **Different levels:**
  1. Data level (ex.: hidden data, phishing, eavedropping,etc...)
  2. Application level (ex.: virus, SQL injection)
  3. Network level (ex.: network intrusion, DoS attack, etc...)

Focus on human + network ...
Ethical hacking;-)

reliability/security

- Specification S, program P
- Question: does P satisfy S?
- A simple example: bank account
- Joined account a - add x and y

```
      P1              ||              P2
get a from db;                get a from db;
a=a+x;                        a=a+y;
update db with a;             update db with a;
```

What is the final value of a?

Reliability's properties

- **Soundness (the program does the job)**
- **Mutual exclusion (data consistency)**
- **Fairness (access for everybody)**
- **No infinite loop**
- **Etc...**
- **Solutions:**
  - Proof methods (Hoare, Milner)
  - Other programming languages (Prolog,CAML,…)
  - Specifications methods (Z, B, UML,LOTOS)

What do we need

- Data: (ex. : my CV)
  - Confidentiality
  - Authenticity
  - Continuity (backup)
  - Consistency (database)
- Applications: (ex.: Moodle)
  - Smoothness/crash
  - 24/24 availability
- Network: (ex. : wifi)
  - Confidentiality
  - Smoothness
  - 24/24 availability

Main threats

- Data destroyed/stolen (human, virus/spyware)
- Eavesdropping (keyloggers)
- Spamming: unsolicited emails (http://interstices.info/anti-spam)
- Phishing: getting private data just by asking ;-)
- Sniffing (network) -> see eavesdropping
- Spoofing (IP usurpation)
- DoS/DDoS: destroying a service
- Network intrusion
- Etc...

Virus, Worn, Trojan

- **Classical pb (see example of code)**
- **No « ideal » solutions**
- **Polymorphic/metamorphic virus ;-)**
- **Anti-virus (signature based)**
- **Commercial**
  - Mac Afee – Symantec – etc...
- **Free :**
  - F-prot – Avast
- **Conclusion:**
  - Option 1 : Set up an anti-virus + update
  - Option 2 : Switch to Linux;-)

Keyloggers

- **Client machine side (hardware or software)**
- **Initially to fight internal threat**
- **Capture the keys using IRQ (see the general algo)**
- **Capture passwords, email addresses, web addresses**
- **Difficult to detect (process table, key logger detection)**
- **How to detect/remove/avoid:**
  - Microsoft Antispyware, Ad-Aware, etc…
  - Avoid accessing your online accounts from public computers
  - Basic tips for entering the data (explain)
  - Home/workplace: log out when leaving ;-)
  - Server side: virtual keyboards (ex. Citibank UK), random sequence (Natwest)
  - Future: bio-metric authentication (fingerprint) !

SPAM

http://interstices.info/jcms/c_41867/spams-et-hams-et-comment-les-filtrer

- **Unsolicited commercial emails**
- **Mail functional diagram:**
  - Mail Transfer Agent (sendmail, postfix, etc...)
  - Mail Retrieval Agent (Outlook, Thunderbird, Eudora, etc...)
- **Solutions: filtering**
  - Bayesian
  - K-nn
  - Kolmogorov
- **Location:**
  - Server side (MTA) (Spamassassin, Mailfilter, etc...)
  - Client side (MRA) (check out with Outlook, Thunderbird, etc...)
- **Future : SPIT ;-))) (with VoIP)**

Hidden data

- **1) Entire files**
  - Ex.: dot files with UNIX/Linux
  - System files with Windows: boot.ini, etc…
  - Virus, Trojan, etc… (executables can only be seen in the process table)
- **2) Data hidden within a file**
  - Watermarking
  - Digital signature (PDF files)
  - MSWord hidden data (see .doc file size versus .sxw)

Very difficult to detect ! See next slide ;-)

Alternate Data Streams

- Coming from Apple FS (Hierarchical FS)
- 1 file = 2 data streams
  - 1 stream for info
  - 1 stream for data
- Special name: `parentfile:filename`
- Create/delete/execute (to be done on the fly)
- Ex. With XP (file injection)
  - `cd /Windows/System32`
  - ‘`dir calc.exe`’ and check the info
  - Create a txt file: ‘`echo « welcome » > ads.txt`’
  - ‘`type ads.txt`’ to check the content
  - ‘`type ads.txt > calc.exe:ads.txt`’ … Done
  - Check with ‘`dir calc.exe`’ then ‘`type calc.exe:ads.txt`’

Execute ADS ;-)

- See http://support.microsoft.com/default.aspx?scid=kb;EN-US;q101353
- See http://www.cknow.com/cms/vtutor/ntfs-ads-viruses.html

```
C:  (go to the root of the disk)
Echo welcome > test.txt
Type notepad.exe > test.txt:ads.exe
Start c:\test.txt:ads.exe !!!!
```

- Using regedit, make it executable at boot time ;-)
  - HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run
  - Add new key with the full path!
- Undetectable with the process table (name of the parent file) – can be seen with dir /r (Win7)
- Cannot be deleted (parent file only)
- Tools: lads.exe (http://www.heysoft.de)
- No detection with classical anti-virus
- No real protection (except Unix FS ;-)

Data encryption

- Data encryption to ensure confidentiality
- D → E(D) → send(E(D))
- Receive(E(D)) → E^{-1}(E(D)) = D
- Main difficulty: E (encryption)
- E properties:
  - Fast
  - Not easy to get E^{-1} starting from $E(D_1)$, $E(D_2)$, $E(D_3)$,...
  - Legal issues (PGP – Phil Zimmerman)
- No need to decrypt: hash function to explain
  - http://pajhome.org.uk/crypt/md5/ to check online
  - www.functions-online.com

Using keys

- 2 types:
  - Symmetric (secret key)
  - Asymmetric
- Symmetric encryption = confidentiality
  - One shared key for encryption (a number or word)
  - The same one for decryption
  - Ex. DES (64bits) then AES (128,256, 512...)
  - +: simple, fast
  - -: to keep the key secret....
- Asymmetric encryption (Rivest,Shamir,Adleman = RSA) = confidentiality + authentication
  - A public key (a number) to encrypt
  - A secret key (a number) to decrypt
  - Sender A uses PK of B to encrypt – B uses SK of B to decrypt
  - Sender A can sign SK A – B uses PK of A to check the sender
  - +: safe
  - -: slow because complex, PKI needed, not suitable for VoIP

Authentication

- The problem: to be sure of the sender (man-in-the-middle attack)
- Electronic signature = certificates
- Trusted organisations delivering certificates (limited validity, can be expensive)
  - VeriSign, Thawte (commercial)
  - CAcert.org (free)

PKI

- PKI=Public Key Infrastructure
- Main purpose:
  - a place to store your public key
  - a way to ensure this is your’s (authentication)
- Pb: how to be sure this key is your public key?
  - Certificate needed from trusted third party: Certificate Authority (CA)
  - Public key
  - Name
  - Life time or validity period
- Ex.: Verisign, Thawtes, ...
- Self certification: ex. IRIT, BITE, (to check live;-)

Cryptographic protocols

- SSL: Secure Sockets Layer (Netscape)
- TLS coming from SSL (standard now)
- Main ideas:
  - Handshake procedure to
    - Agree on encryption algo (cypher + hash)
    - Get (check) server ident. + certif. + PK
    - Generation of the session keys
  - After successdful handshake, all data are encrypted
- TLS can be used with any appli. protocols (HTTP, FTP, SMTP, etc...)
  - Same protocol
  - Different ports ex. smtp -> 25 but secure-smtp ->465
- Support from Visa, AMEX, CB

A famous example: HTTPS

- **HTTPS=HTTP + SSL (see bottom right of the screen with BITE webmail, bank)**
- **Dedicated port: 443 (instead of 80)**
- **Ex.: Apache server**
  - – Configuration file: httpd.conf or apache.conf
  - – Secure server: ssl.conf (TBD)

Another example: PGP

- **Paul Zimmermann - 1991**
- **Pretty good privacy**
- **No known vulnerability**
- **Freely available to:**
  - Create your keys
  - Create your certificate
  - Encrypt your emails
- **Now zfone for VoIP**

Network intrusion

- **Examples**
  - DoS
  - Port scanning
  - Network monitoring
- **Objectives**
  - Malicious activities
  - Computer crack
- **Solutions**
  - Ethical hacking
  - Firewalling
  - Network intrusion system

A simple example

- **Avoid search engine indexing**
  - **Meta-tag:**

    ```
    <meta name="robots" content="noindex,nofollow" />
    ```

  - **Robots.txt file: robots exclusion protocol (show example)**
- **Target : web crawlers (search engines)**
- **Gentleman agreement only**
  - Rely on cooperation between server/robot
  - No guarantee for privacy
  - Publicly available files
- **A simple example with nmap**
  - nmap -T Aggressive -P0 -A -v www.irit.fr
  - nmap -v -T4 -PN -A www.irit.fr
  - we get the robots.txt file
  - we try one;-)

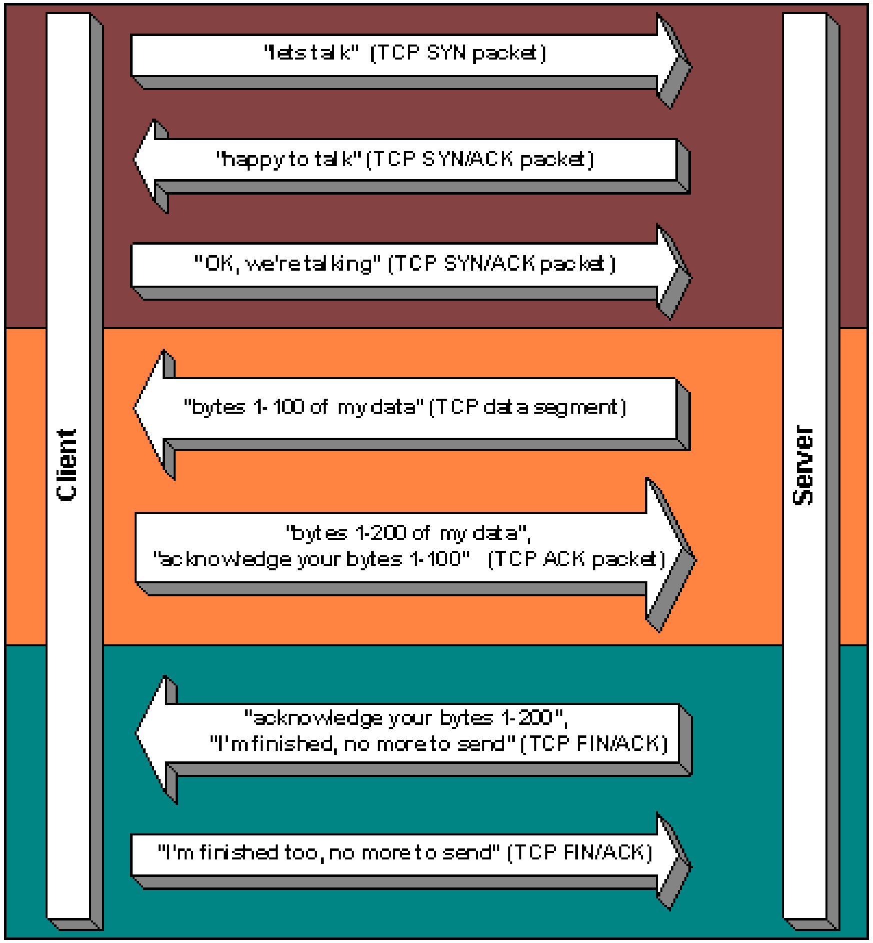
Network sniffing

- **Free available tools on the web (tbc)**
  - nmap, nessus, ethereal, wireshark, others...
- **Sniffing howto:**
  - Get all the IP packets
  - Analyse these IP packets
  - Do « bad » things
- **How to get all the packets:**
  - Passive method: promiscuous mode for NIC !
  - Active method: program !

Promiscuous mode

- **Ethernet/IP:**
  - Send your unique MAC+IP to network (media)
  - **Normal** mode/unicast: NIC gets only the relevant packets
  - **Multicast** mode: get the packets of my group
  - **Promiscuous** mode : get all the packets !
- **How to get/modify the mode**
  - Depending of the NIC
  - Windows: control panel-network-choose the card-pick up properties-advanced-etc...
  - Linux: `ifconfig eth0 -promisc`
- **Harm the network traffic**
- **Very difficult to detect !**

TCP/IP review



Client

Server

"lets talk" (TCP SYN packet)

"happy to talk" (TCP SYN/ACK packet)

"OK, we're talking" (TCP SYN/ACK packet)

"bytes 1-100 of my data" (TCP data segment)

"bytes 1-200 of my data",
"acknowledge your bytes 1-100" (TCP ACK packet)

"acknowledge your bytes 1-200",
"I'm finished, no more to send" (TCP FIN/ACK)

"I'm finished too, no more to send" (TCP FIN/ACK)

Wifi

- WEP (wireless equivalent privacy)
  - 1key to encode the data
  - Can be easily decrypted
- WPA (WPA2) (wifi protected access)
  - 1key first access
  - New generated key (1 per second)
- Best solution for now
  - WPA2 (support AES algo better than TKIP)
  - Maximum length for the key
  - dhcp filtering

Cracking tools

- airsnort (http://airsnort.shmoo.com/)
- aircrack (http://www.aircrack-ng.org/)
- Main idea:
  - Collect data
  - For wep:specific algo
  - For WPA: brute force algo
  - For WPA2: not possible ...for now?
- DHCP filtering →
  - Interrupt the session
  - Get the MAC (media access control)

Firewall

- **Functional diagram (TBD)**
- **2 kinds:**
  - Hardware based (ex.: CISCO)
  - Software based (ex.: Linux IPTABLE, etc...)
- **Packet filtering:**
  - Static
  - Dynamic
- **Policy = set of rules**
  - Protect from tcp/ip attack, scan, probe
  - Protect from DoS
- **Basic policy: here**
- **VoIP aware firewalls**

Examples

- Pix (CISCO)
- IPTABLE (service iptables status)
  - iptables -P INPUT DROP
  - iptables -P OUTPUT DROP
  - iptables -A INPUT -p tcp --sport 22 –j(ump) ACCEPT (for ssh)
  - iptables -A INPUT -p udp --sport 22 -j ACCEPT
- Block MSN Messenger ;-)
  - iptables -A FORWARD -p tcp --dport 1863 -j DROP
  - iptables -A FORWARD -d 65.54.239.142/25 -j DROP
- Very difficult to get a secure firewall

LAMP/WAMP

- **Linux/Windows, Apache, MySQL, PHP**
- **2006: 43% of frauds with PHP (NIST)**
- **3 important files:**
  - httpd.conf for Apache
  - my.ini for MySQL
  - php.ini for PHP
- **+ the config file for Linux servers;-)**
- **Examples:**

Linux security

- **No virus...**
- **File system security**
  - **A file/directory: u g o**
  - **Permissions: r w x**
- **Process permissions**
  - Owner permission +
  - Setuid/setgid bit
- **Example**: passwd command (explain)
  - Setuid: `chmod u+s toto.exe`
  - Setgid: `chmod g+s toto.exe`
- **Huge security hole;-)**

Conclusion

- Security: hot topic
- Main holes: human errors!
- WIFI network... worse !
- Convergence (GPRS/GSM, etc...)
- SPIT ;-)
- Etc...

The never ending story!

Do not forget!
The basic;-)

- At least 2 external backup
- Backup : USB, DVD, etc...
- Partition main drive into A and B
- Backup on B
- Antivirus + antimalware up-to-date
- Check USB before plug in
- Non admin account login
- Wifi: WPA2 + DHCP filtering