ANNOUNCEMENTS

- Check class announcements daily
- You must implement programming projects by yourself

FINAL EXAM INFORMATION

The final exam for the course will be a common final exam (all sections in one room, taking the exam at the same time).

FINAL EXAM INFORMATION
DATE: Friday Dec 16
TIME: 4pm-6pm
LOCATION: CSIC 1115

We have verified this information with the school. Ignore any other time/date you may see in the university web site.

SECURITY (EMAIL)

- Least secure of internet protocols
- Avoid sending sensitive information (e.g., passwords) over e-mail
- Provide e-mail addresses in web sites in a way is not easily recognized by spam programs
  - Use at rather than @
  - Put an image with the e-mail
  - Avoid mailto
  - Use JavaScript to generate e-mail address
- Encrypt the message using PGP (Pretty Good Privacy) or GPG (GNU Privacy Guard)
- http://www.wbwip.com/wbw/emailencoder.html

SECURITY (PASSWORD-PROTECTED SITES)

- **Approach not recommended**
  - Store encrypted password
  - Decrypt password and compare against user provided password
- **Better approach**
  - Store encrypted password
  - Encrypt provided password and compare against stored password

SECURITY (ENCRYPTION)

- **Encryption** → process of converting plaintext into ciphertext
- **Decryption** → process of converting ciphertext into plaintext
- **Symmetric cryptography** → sender and receiver share the same key
- **Asymmetric (Public Key) cryptography** → sender and receiver have different, complementary keys
- **Symmetric cryptography**
  - Relatively fast compared to asymmetric
  - Drawbacks
    - Keys must be change frequently
    - How to distribute the key safely

SECURITY (ENCRYPTION)

- **Branches of public key cryptography**
  - Public key encryption
  - Digital signatures
- **Public key Encryption**
  - Example algorithm: RSA
  - Relatively slowed compared to symmetric
  - How it works?
    - Each user has a public/private key pair.
    - Public key is widely known
    - Private key only known by user that generated it
    - If user A wants to send user B a message, user A encrypts message with B's public key. B will decrypt the message with B's private key. The only way to decrypt the message is by using B's private key
- **Digital signature**
  - Message signed with sender's private key can be verified by anyone with sender's public key thereby proving message authenticity

DIGITAL CERTIFICATES (CERTIFICATES)

- **Digital Certificates** → electronic documents that contain information about a public key and the owner (name, address, etc.)
- Employed to verify a public key corresponds to a particular organization
- Certificates must be issue by a trusted third party known as certificate authority (CA) which guarantees the information is correct
- **About certificates**
  - Have a validity period and can expire
  - They can be revoke
  - Browsers have a collection of root certificates
    - In Firefox – Tools→Options→Advanced→Encryption→View Certificates

NEED FOR SECURITY

- **SSL (Secure Sockets Layer) Protocol** → Protocol that enable us to satisfy the need for security in client-web server transactions
- The algorithm provides support for confidentiality, integrity and authentication
- **SSL connection is established as follows:**
  - User connects to web server through the browser
  - Browser and server exchange public keys and certificate information
  - Browser checks server certificate validity (certificate not expired, issued by CA, etc.)
  - Optional: server can request a valid certificate from the client
  - Using public keys server and client determine a symmetric key to use
  - Communication from this point on is through symmetric cryptography

HTTPS

- **https** → http where
  - A different default port (443) is used
  - An extra layer of encryption/authentication exists between HTTP and TCP
- **https** → is not a separate protocol but a combination of HTTP over encrypted SSL or TLS transport mechanism

SOCIETY /IMPACT OF SOFTWARE FAILURES

- Software becoming part of basic infrastructure
  - Software in cars, appliances
  - Business transactions moving online
- Computers becoming increasingly connected
  - Failures can propagate through internet
    - Internet worms
  - Failures can be exploited by others
    - Viruses
    - Spyware

SOFTWARE CONTRIBUTES TO REAL FAILURES

- Bugs in software may cause real-world failures
- Example – Air Force F-22A Raptor
  - Stealth fighter costing $300 million each
  - 1.7 millions lines of code for plane's avionics

SOFTWARE CONTRIBUTES TO REAL FAILURES

- Air Force F-22A Raptor software fails midair
  - DefenseNews.com (March 5, 2007)
  - "When a dozen Raptors en route from Hawaii to Japan crossed the International Date Line for the first time, the jets' Global Positioning System navigation avionics went haywire, forcing the pilots to turn around."
  - **GPS software unable to handle change in longitude from W179.99^{o} to E180^{o}**
  - **Raptor pilots visually followed refueling tankers back to Hawaii**

SOFTWARE CONTRIBUTES TO REAL FAILURES

- Happy ending for Raptor?
  - Lockheed-Martin provided software fix in 48 hours
  - For “operational security reasons” the USAF declined to elaborate, saying only that the F-22A “experienced a software problem involving the navigation system”
- Tough being a Raptor test pilot
  - DefenseNews.com (March 5, 2007)
  - “When the plane was in developmental stages … pilots flying the Raptor would often have to reboot the onboard computers that controlled the jet’s high-end functions”

OTHER FAMOUS SOFTWARE FAILURES

- 1990 AT&T long distance calls fail for 9 hours
  - Wrong location for C break statement
- 1996 Ariane rocket explodes on launch
  - Overflow converting 64-bit float to 16-bit integer
- 1999 Mars Climate Orbiter crashes on Mars
  - Missing conversion of English units to metric units
- Other Failures available at:
  - http://news.zdnet.com/2424-9595_22-177729.html

WHY IS SOFTWARE SO DIFFICULT?

- Complexity
  - Software becoming much larger
    - Millions of line of code
    - Hundreds of developers
  - Many more interacting pieces

- Length of use
  - Software stays in use longer
    - Features & requirements change
    - Data sets increase
    - Can outlast its creators

SECURITY SITES

- www.securityfocus.com/
- www.cert.org/

IN-CLASS LAB