FSEM 1111 Computer Security – from a Free Software Perspective

Christian Grothoff
christian@grothoff.org

http://grothoff.org/christian/

Security Overview

Computer Security Overview

- Computer Security $\equiv$ protecting information
- **Protecting:** Integrity, confidentiality, authenticity, availability

  Information: Randomness, entropy, correlation, storage, transmission

Topics

- Cryptography and Protocols (theory)
- System Administration (practice)
- Privacy, Policies and Legal Aspects (politics)

Terminology (1/5)

- An adversary is a subject trying to break the security of a system
- A threat is a mechanism that the adversary can employ to achieve his goals
- A risk is a loss that would occur if the adversary succeeds
- A vulnerability is a flaw creating a threat
- A threat model describes the mechanisms available to the adversary
- A trust model describes subjects that are trusted not to have vulnerabilities
- A security model specifies functional and security goals together with threat and trust models

Terminology (2/5)

- Plaintext: P
- Ciphertext: C
- Encryption: $E_K(P) = C$
- Decryption: $D_K(C) = P$
- Cryptography + Cryptanalysis = Cryptology
- Steganography

Terminology (3/5)

- Authentication: receiver ascertains origin of message
- Integrity: verify message was not modified in transit
- Nonrepudiation: sender cannot deny sending message

Terminology (4/5)

- Cipher $= (E, D)$
- restricted algorithm $\equiv$ security based on secrecy of algorithm
- modern algorithm $\equiv$ security based on secrecy of key K

Terminology (5/5)

Attacker limitations:

- Data complexity (how much data required as input to the attack)
- Processing complexity (how much processing is needed)
- Storage requirements (how much memory is needed)

Kerckhoff's principle (1883)

The only thing the adversary does not know is the secret key.

The design of encyrption and decryption algorithms and the protocol is public:

- Allows public scrutiny of the design
- No need to replace system if design is exposed
- Same design can be used for multiple applications
- Focus on security the key!

Secure Voting, US-style

KEYS TO THE KINGDOM
Photo taken from Diebold's online store. The keys that open every Diebold touch-screen voting machine. Working copies have been made from the photo.

Substitution Ciphers

- Monoalphabetic ciphers $\equiv$ 1:1
- Homophonic substitution ciphers $\equiv$ 1:n
- Polygram substitution ciphers $\equiv$ n:m
- Polyalphabetic substitution ciphers

Famous examples: Caesar Cipher, ROT13, Vigenere, Enigma

Transposition Ciphers

- change the order of the characters, not the characters
- frequency distribution unchanged
- requires buffers in memory

XOR with key

- Vigenere polyalphabetic cipher
- Generally easy to break,
- except: key length = ciphertext length

Question

Why are one-time-pads almost never used in practice?

Questions

?

Defeating the Evildoers

CERT:

1. Install and Use Anti-Virus Programs
2. Keep Your System Patched
3. Use Care When Reading Email with Attachments
4. Install and Use a Firewall Program
5. Make Backups of Important Files and Folders
6. Use Strong Passwords
7. Use Care When Downloading and Installing Programs
8. Install and Use a Hardware Firewall
9. Install and Use a File Encryption Program and Access Controls

CRISP:

1. Use UNIX-based systems and avoid being `root`
2. Frequently update your software, it is free
3. Refuse to use Microsoft products and document formats
4. Be aware what services you run (`netstat -ntpl`)
5. Use version control for important files
6. Use strong passwords where necessary
7. Avoid using non-free software
8. Do not buy random security equipment
9. Use cryptography appropriately
10. Think. Sometimes, wear black hats.

Question

Why is UNIX important?

Review: UNIX File Permissions

- Standard permissions: Read (4), Write (2), eXecute (1)
- Differentiation by: User, Group, Others
- `man chmod`, `man chown`
- Default permissions are $arg\& \ mask$ where `arg` is specified by the application. For `mask`, see `man umask`

Process User Identifiers

- Each process is associated with multiple user IDs: real, effective, saved and possibly others
- Real UID is the UID of the process that created this process. Can only be changed if effective UID is root (0).
- Effective UID is used for permission checks; EUID can be changed to real UID or to saved UID. If EUID is 0, anything goes.
- New files are created using the effective UID

SUID, SGID

- If permissions of executable file are set to SUID, SUID of executed process will be set to UID of the file's owner.
- This allows the program to switch to those permissions using `seteuid(SUID)`
- Processes also have multiple group IDs, the same rules apply.
- Binaries with SUID and SGID can be used to elevate permissions

Groups

- Each user can be in any number of groups
- `newgrp` can be used to change the current group ID
- `/etc/group` specifies group memberships
- `groups` lists current memberships

Question

Why is UNIX important?

Question

Why is UNIX important?

Answer:

UNIX symbolizes simple, standardized and open solutions that work extremely well.

IP – the Internet Protocol

<table>
<tr><td>Version</td><td>HDL</td><td>ToS</td><td colspan="2">Length</td></tr>
<tr><td colspan="3">Identification</td><td>Flags</td><td>Fragment offset</td></tr>
<tr><td colspan="2">TTL</td><td>T. Protocol</td><td colspan="2">Checksum</td></tr>
<tr><td colspan="5">Source IP address</td></tr>
<tr><td colspan="5">Destination IP address</td></tr>
<tr><td colspan="5">Options (optional)</td></tr>
<tr><td colspan="5">Data (Length–HDL bytes)</td></tr>
</table>

TCP – the Transmission Control Protocol

<table>
<tr><td colspan="3">Source port</td><td>Destination Port</td></tr>
<tr><td colspan="4">Sequence Number</td></tr>
<tr><td colspan="4">Acknowledgment Number</td></tr>
<tr><td>Data offset</td><td>Reserved</td><td>Flags</td><td>Window</td></tr>
<tr><td colspan="3">Checksum</td><td>Urgent Pointer</td></tr>
<tr><td>Options</td><td colspan="3">Padding</td></tr>
<tr><td colspan="4">Data</td></tr>
</table>

Homework

Before the next lecture:

- Learn about `nmap`.
- Figure out how to determine the IP of your notebook.
- Use `nmap` (from the lab) to scan your notebook.
- Which network-facing services are running on your notebook?
- Which network-facing services are running in the lab?

Questions

?