CSE484 Final Study Guide

Winter 2013

NOTE: This study guide presents a list of ideas and topics that the TAs find useful to know, and may not represent all the topics that could appear on the final exam.

Security Mindset

• Security reviews
  – Assets
  – Adversaries
  – Threats
  – Vulnerabilities
  – Risks
  – Defenses
• Attack trees

Buffer Overflows and Software Security

• Off-by-one errors
• Overflow function pointers
• Format string vulnerabilities
• TOCTOU
• Overflow implicit casts
• Timing attacks
• Canaries
• Fuzzing
• Difference between /dev/random and /dev/urandom
• Why not use the rand() function for security purposes?

Cryptography

• Secrecy guarantees of one-time pad vs. block ciphers
• Pros and cons of OTP
• What’s the purpose of a Feistel network?
• What is a meet-in-the-middle attack, and why does it work on double DES (but not triple DES)?
• Why use a PBKDF (password-based key derivation function)?
• CBC vs ECB vs CTR mode
• What happens if an IV isn’t random for CBC mode?
• Why should counters not overlap in CTR mode?
• Is integrity checked with these modes?
• Why have multiple rounds in a Feistel network? How to break it with one or two rounds?
• CBC-MAC
• Properties of cryptographic hash functions
  – One-wayness
  – Collision resistance
  – Weak collision resistance
• HMAC
• Encrypt-and-Mac vs. Mac-then-Encrypt vs. Encrypt-then-Mac
  – Why is Encrypt-and-Mac insecure?
  – Why is Mac-then-Encrypt weaker than Encrypt-then-Mac?
• Birthday attacks
• How many random input attempts would it take to find a collision for a particular 128-bit hash value?
• And for the same hash function how many attempts to find any single collision?
• How can Alice and Bob flip a coin over the phone?
• Password strength and entropy
• Is CBC or CTR mode preferred for parallelization, and why?
• Why is Diffie-Hellman secure?
  – DHP and DDH and Discrete Logarithm Problem
• Privacy and integrity in RSA
• RSA key generation. Or given p, q, n, phi, e, d, what are the PK and SK?
• Why is repeated squaring method of optimizing exponentiation insecure?
• Why is RSA secure?
  – Factoring is hard
  – Modular inverses are hard
  – Taking the \( e^{th} \) root mod is hard
• TLS version rollback attack
• What alternative to CAs, used with PGP, enables trust of public keys?

Cryptography Attack scenarios

• Ciphertext only
• Known plaintext
• Chosen plaintext
• Chosen ciphertext

Cryptography Summary

• Privacy
• OTP
• Block ciphers: AES, DES, ...
• Modes: ECB, CBC, CTR
• Public Key: DH, RSA
• Integrity
• MAC
• Hashes: MD5, SHA, ...
• Privacy and Integrity
• Enc-then-MAC
• Authenticity (and integrity)
• Digital signatures: RSA, DSS, ...

Certificates
• Certificate Authorities sign certificates
• Roots authorize intermediates
• What’s a certificate chain, and why do we have them?
• Risks of trusting certificate authorities?
• Alternate solutions? Convergence, CRLs, ...

ROP
• What is return oriented programming?
• What defense drives attackers to use it?
• Equivalents

General Principles to Achieve Software Security
• Input validation
• Least privilege
• Always check return values
• Securely clear memory (passwords and keys)
• Failsafe defaults
• Defense in depth
• Reduce size of TCB
• Minimize attack surface
• Use vetted components (and standard crypto libs!)
• Security by design

Web
• Why is encoding state in URL a bad idea?
• Web authentication via cookies
  – Why is adding a MAC just to “value (or price)” still insecure?
  – Attack scenario
  – Better cookie authenticator (what’s included in the cookie, what’s MAC-ed, etc.)
• Javascript security model
• Cross-site scripting attack
• Stealing cookies
• Login with CSRF – why would an attacker do this?
• Input validation / escaping
• Better to MAC all data at the same time rather than separately to prevent them being reassembled differently.
• Cookies
  – set-cookie (HTML) or document.cookie (JS)
  – Can narrow to subdomain or path
- Secure cookie (only send HTTPS)
- Set to HTTPOnly for to prevent JS access

* SQL Injections
* Web tracking
  - Storage
    * Cookies
    * HTML5LocalStorage
    * Flash cookies (LSO)
  - Popups and redirects can make 1st party cookies
  - Evercookie / zombie cookie

**Authentication**

- How should you store passwords on a server and what threats are you worried about?
- Dictionary attack, and why would we want to use salt/pepper?
- How can we improve the security of passwords?
- What are the possible alternatives or enhancements?
  - Pros and cons for each? (graphical password, biometrics, password managers, two-factor authentication)
- Biometrics
  - Types of biometrics
  - Any issues associated with using biometrics? Advantages and disadvantages?
  - Fraud rate and insult rate
- Why not store encrypted passwords on server?
- Something you know, something you have, something you are.

**Security and HCI**

- Why is usability important to achieving security?
- What are the key issues and challenges? How to cope with these challenges?
- CAPTCHA
- Phishing and social engineering
- Clickjacking

**Android Security**

- How are Android apps isolated? How/why does this differ from desktops?
- Sandboxes
- Permissions and manifests
- Memory management
- App signing
Anonymity

- You cannot be anonymous by yourself
- Anonymous emails
  - How does a mix network work?
- Onion routing
  - How does it work?
  - Tor circuits
- Attacks on anonymity
  - Eg. What is a sybil attack? How does it affect Tor?

Social Engineering

- Elicitation
  - How and why does it work?
  - Strategies

Physical Security

- Why is it necessary?
- Locks
  - Pin tumbler lock design and lock-picking basics