Summary and Exam Preparation

Software Security

Steffen Helke
Chair of Software Engineering

30th January 2019

Brandenburgische
Technische Universität
Cottbus - Senftenberg

Objectives of today's lecture

- ➜ Understanding the *conditions* for the written examination
- ➜ Repetition of important content

General Remarks: Written Examination

When? Wednesday (13.2.), 11:30 - 13:00

Where? Large lecture hall ("Großer Hörsaal"), Note that this lecture hall has its own building, which is located between the main building and the ZHG.

How long? about 90 min

Testing aids? A non-programmable calculator is allowed

Prerequisite and Module Examination

Exercises

➜ Successful treatment of two exercise sheets

Presentation

➜ Successful presentation on one technical topic

Final Module Examination

➜ Written examination, 90 min

Repetition of Important Content

Foundations & Motivation

1. Overview and basic terms (mandatory)
2. Malware categorisation (mandatory)
3. Examples for software vulnerabilities, e.g. buffer overflow (mandatory)

Security Analysis, Design & Anonymity (Part I)

4. Security process, misuse cases & attack trees (mandatory)
5. Multi-Level security, e.g. BLP & BIBA (probably)
6. Information flow control, e.g. JiF (probably)
7. Anonymity & pseudonymity, e.g. TOR & DC (probably)

Repetition of Important Content

Encryption Techniques (Part II)

8. History of cryptology, monoalphabetic and polyalphabetic cipher, one-time pad, Pfitzmann's table (mandatory)
   Design and attacks for Enigma (probably not)
9. Pseudo-one-time pad using s^2-mod-n generator (probably)
10. GMR - strong cryptographic signature system (probably not)
11. RSA - encryption & signatures (probably)
12. DES/AES - symmetric encryption (probably)
13. Operation modes: block cipher vs. stream cipher (probably)
    Operation modes for full disc encryption (probably not)

Repetition of Important Content

Security Protocol Engineering (Part III)

14. Needham-Schroeder protocol (probably)

15. Kerberos protocol, v4 & v5 (probably)

16. Analysis using BAN logic (probably not)

17. Verification using CSP/FDR (probably not)

Introduction & Motivation

Introduction & Motivation (1, 2, 3)

- What is the difference between *safety* and *security*?
- What does *security* mean by C. Eckert? How is dependability defined by Laprie? What are the attributes of dependable systems?
- What are the three most important protection goals?
- Which types of *malware* do you know? What is an *universal* Trojan horse? What is a *transitive* Trojan horse?
- What are *worms* in contrast to viruses?
- How to protect against computer viruses? What is the *principle of least privilege*? Is it possible to detect known viruses all the time?
- How does an attacker manipulate the stack management to perform a buffer overflow?
- How works a *code injection* for an buffer overflow? What is a NOP command and how is it used?
- Do you know any countermeasures to prevent buffer overflow?

Security Analysis, Design & Anonymity (Part I)

Security Process, Misuse Cases & Attack Trees (4)

- Which process models in security engineering do you know? What are typical activities of a security analysis?
- Which optimizations are suggested by the BSI?
- How to classify protection goals based on the categories *communication content* and *communication circumstances*?
- Which correlations between protection goals do you know?
- What are the most important artifacts for security analyses?
- What is the basic notation of misuse cases?
- How to use attack trees to refine attacker goals?
- Please illustrate misuse cases and attack trees for a given example
- How to annotate an attack tree by costs and probabilities?

Multi-Level Security (5)

- Which access control strategies are distinguished?
- Which protection goal is implemented by the Bell-LaPadula model (BLP)?
- What are the most important rules of BLP?
- How are security classes represented and how are these classes ordered? What is a sensitivity level and what is a compartment set?
- How to make BLP more flexible? What do you know about the *high watermark principle*?
- What is the difference between a strong and a week *tranquility property*?
- How to bypass BLP with the help of covered channels?
- What are the differences between the BIBA and BLP model?
- What are the design principles of a *Trusted Computing Base* (TCB)?

Information Flow Control (6)

- Why is the generalization of the Bell-LaPadula model by Dorothy Denning useful for information flow control?
- What do you know about implicit information flows inside of a program and how can we analyse the code using Denning's operators (maximum and minimum)?
- Where in a Java program implicit information flows can arise?
- How to specify a security policy for confidentiality or integrity using JiF?
- How are implicit information flows handled in JiF?
- What is the meaning of an empty security label in Jif?
- What does the JIF compiler check for an assignment?
- What is problematic with a method call? How the JiF-Compiler is able to solve this problem?
- What is the meaning of begin and end-labels in JiF? How is it possible to support JiF-refactorings?

Anonymity and Pseudonymity (7)

- How to define *ordinary* and *perfect* anonymity?
- How to classify pseudonyms with regard to anonymity and linkability?
- What are the design differences between JAP and TOR?
- Which protocol steps are performed by TOR to support anonymous web browsing and which steps are needed for hidden services?
- What are the characteristics of probabilistic anonymity? Is there any alternative way to achieve anonymity?
- Why do you need the Diffie-Hellmann protocol for TOR? Why is this protocol not secure when used naively?
- How to construct a ticket in the shape of onion shells in such a way that the information contained remains as anonym as possible?
- What is the process flow of the DC-example for three actors? What is the information that is communicated anonymously?
- How to extend the simple DC protocol to larger networks?

Encryption Techniques (Part II)

Introduction Encryption & History (8)

- What are the differences between cryptography and cryptanalysis?
- How to decrypt a ciphertext encrypted by a monoalphabetic substitution using a frequency analysis?
- Do you know an example for a polyalphabetic substitution?
- How to decrypt messages encrypted by a Vigenére cipher
- What are the differences between *Vigenére cipher* and *Homophonic substitution*?
- How works the encryption using a Vernam cipher? Why is this cipher *information-theoretically* secure?
- How was the Enigma designed and what was the key for this electronic cipher machine? How was it possible to decrypt ciphers encrypted by the Enigma?
- How to categorize modern encryption systems using Pfitzmann's table? What means that a crypto system is *cryptographically strong*?

Encryption using s^2-mod-n (9)

- What means *prime factorization* and why is this operation so important for asymmetric encryption systems?
- There are two other operations that are based on factorization. Why these operations are important too?
- What cryptographic assumption is the basis for s^2-mod-n?
- Why is s^2-mod-n also called *Pseudo One-Time-Pad*?
- What are the differences between the symmetric and the asymmetric variant of s^2-mod-n?
- Which part of the key is public, which part is secret?
- How is it possible to compute a square root efficiently? Illustrate the procedure using an example.
- Why is s^2-mod-n cryptographically strong? Note you do not need to provide a formal proof for this property, however you should be able to explain the rough idea behind this proof.

GMR - Cryptographic Signature System (10)

- How does GMR differ from other signature systems you know?
- What means *collision resistant* for two given permutations?
- How to generate a signature using GMR? Give an example.
- Why is it necessary to restrict the definition range for the square functions of GMR? How is this restriction implemented?
- How to use the *Chinese Remainder Algorithm* (CRA) to generate GMR signatures?
- Which part of the key is public, which part is secret?
- How is it possible to attack the signature system GMR and how can this attack be prevented?
- Why is GMR cryptographically strong? Note you do not need to provide a formal proof for this property, however you should be able to explain the rough idea behind this proof.

RSA - Encryption & Signatures (11)

- How does RSA differ from other encryption systems you know? Why is RSA only classified as *well researched*?
- Which part of the key is public, which part is secret?
- How to generate a suitable RSA key pair?
- Why is the naive version of RSA not secure against attacks based on the multiplicative property? How can this attack be prevented?
- What are the technical challenges involved in implementing an RSA cryptosystem? Which algorithms must be provided?
- Why does the double exponentiation of a plaintext with secret and public keys result in the plaintext again? Note you do not need to provide a complete proof of correctness, however you should be able to explain the rough idea behind this proof.
- How works a total break of RSA by Fermat's factorization method? What is a suitable countermeasure?

DES/AES - Symmetric Encryption (12)

- What is the meaning of confusion and diffusion and how these concepts are usually implemented?
- What are the advantages of a Feistel network and which encryption system uses this technology?
- Why is DES considered insecure today and should not be used anymore?
- How works a *brute-force attack*? What is an important assumption of this attack?
- Why is the complementarity property of DES useful for an attacker?
- How much better is Triple-DES compared to DES? How works a Meet-in-the-Middle attack?
- Which four operations need to be implemented for AES? Which of these operations are based on polynomial arithmetic?
- How to implement polynomial arithmetic efficiently for AES?
- How many rounds are required for the AES algorithm?

Operation Modes: Block Cipher vs. Stream Cipher (13)

- What are the problems with the practical use of encryption methods? How can operation modes help to solve these problems?
- What is the difference between a synchronous and a self-synchronizing mode?
- Why is the Electronic Codebook Mode (ECB) considered insecure and should not be used?
- Why could be Cipher Block Chaining (CBC) a better alternative? What are the disadvantages of this operation mode?
- What is the problem with error propagation? Which operation mode is not sensitive to this problem?
- How can random access implemented?
- Which operation modes are suitable for hard disk encryption? Why could be a tweaked-codebook mode (e.g. XTS) a good option? How many keys are required for this mode?

Security Protocol Engineering (Part III)

Needham-Schroeder Protocol (14)

- Why is it so important to use secure protocols in addition to secure encryption algorithms?
- What principles (in addition to the security aspect) are considered when designing a security protocol?
- Which notation is usually used to specify a security protocol?
- Specify the steps of the traditional Needham-Schroeder protocol (symmetric variant)? Why is this protocol vulnerable?
- Specify the asymmetric variant of the Needham-Schroeder protocol. What attack for NSP has not been detected for many years?
- Specify a man-in-the-middle and a replay attack for the NSP example.
- Which countermeasures exist to prevent these attacks?

Kerberos Protocol (15)

- What are the differences between Kerberos and Needham-Schroeder protocol? What are the common features?
- What was the motivation for introducing the Kerberos protocol? Which concepts are used to avoid the frequent request for passwords during operation?
- What are the main differences between Kerberos v4 and v5?
- Which protocol step is particularly vulnerable to replay attacks on Kerberos? What would be a successful scenario from an attacker's point of view?
- What other vulnerabilities of the Kerberos protocol could an attacker use?
- How was it possible to avoid an explicit storing of passwords?
- Why can't timestamps provide absolute protection against replay attacks?

Protocol Verification using BAN logic (16)

General Remarks

- Why does it make sense to formalize protocols using BAN logic?
- How much can you rely on a protocol that has been proven to be correct?

How to apply BAN logic?

- What are the most important syntactical elements of this logic?
- What is critical about using the BAN logic? Why is the idealization step often a source of errors?
- How to derive a new proposition with the BAN logic?
- Can you specify two or three deduction rules of your own choice?
- What exactly is to be proven? What are the differences between first-order and a second-order goals?

Protocol Verification using CSP/FDR (17)

Foundations

- What is the modeling idea of CSP? Which important CSP operators do you know? What are processes, channels and events?
- Which CSP semantics did we use in the course? Illustrate the semantics using a small example

Protocol Specifications

- How modeled Gavin Loewe the NSP protocol using CSP? Describe only the rough idea behind this model.
- Which processes communicate with each other? Which events are used to synchronize?
- How is the attacker modeled? What knowledge does he have and how can he learn new information?
- What refinement proof was used to verify the Needham-Schroeder protocol's vulnerability by FDR?