CSCI 345 – Computer and Network Security

Final review, Spring 2020

What to study (chapters are from your textbook “Security in Computing”):

1. Ch. 1 (covered in midterm review):
    a. CIANA
    b. Vulnerabilities
    c. Controls
2. Ch. 2 (covered in midterm review):
    a. Authentication
    b. True / False positives / negatives
    c. Access control
    d. Crypto: DES, AES, public key, signatures, certificates, error codes
3. Ch. 3:
    a. Buffer overflow
        i. What are the causes of BO?
        ii. How is BO implemented?
        iii. Can we cause buffer overflow with languages other than C?
        iv. Find the parts in the code that may cause buffer overflow and suggest how you would resolve the issue.
    b. Other unintentional programming oversights!
    c. Malicious code
        i. What is the difference between worm and virus?
        ii. Discuss the basic characteristics of malware.
        iii. You got a file that you know is malware, how will you test it?
    d. Countermeasures
4. IN ADDITION TO Ch. 3
    a. Smashing the stack for fun and profit pdf posted on website
    b. OWASP checklist for secure programming
5. Ch. 4:
    a. Browser attacks
    b. Web attacks: injection, cookie interception & manipulation
        i. Why is SQL injection consistently No 1 in the OWASP top ten vulnerability list?
        ii. What are the typical problems in web testing?
        iii. Give scenarios that you would use to test a website.
        iv. What are the steps to test your website for cross site scripting?
        v. What are the pros and cons of using cookies?
        vi. Describe an attack to any of the pillars of security (CIANA) where you can manipulate cookies.
    c. Email attacks
6. Ch. 5:
    a. Security mechanisms for operating systems
    b. Secure by design: how is the OS designed to embed security?
    c. Rootkit: how it works, how we can defend against it?

7. Ch. 6:
    a. Threats
    b. DoS and DDoS
        i. Give the attack vectors for the described DDoS attack.
        ii. How can we defend against DDoS?
    c. Wireless
        i. Enumerate the wireless vulnerabilities, which pillars of security they affect, and how we can defend against these.
    d. Defense: firewalls, AVs, IDS/IPS (do not forget false/true positives/negatives), SSL/TLS, VPN, Onion routing
8. ADDITIONAL Networking material:
    a. Slides on TCP/IP stack (7_ComputerNetworks)
        i. What happens from the moment you turn on your laptop and you browse to a website until the moment you receive a response? Describe all the messages, protocols, and layers that are used in this scenario.
        ii. Enumerate the subnets.
        iii. Describe vulnerabilities in the TCP stack layers. How would you resolve these?
    b. TCP Sequence number attack
    c. ARP/MAC spoofing
    d. DNS poisoning
9. Ch. 12 (covered in midterm review):
    a. Symmetric crypto
    b. Asymmetric crypto
    c. Digital signatures
10. ADDITIONAL Crypto material:
    a. Slides
    b. Information theory – Entropy
11. <u>Questions at the end of each chapter</u>

Sample questions:

1. You are told to design an intrusion detection algorithm that identifies vulnerabilities by solely looking at transaction length, i.e., the algorithm uses a packet length threshold T that determines when a packet is marked as an attack. More formally, the algorithm is defined:

$$D(k,\ T) \rightarrow [0,1]$$

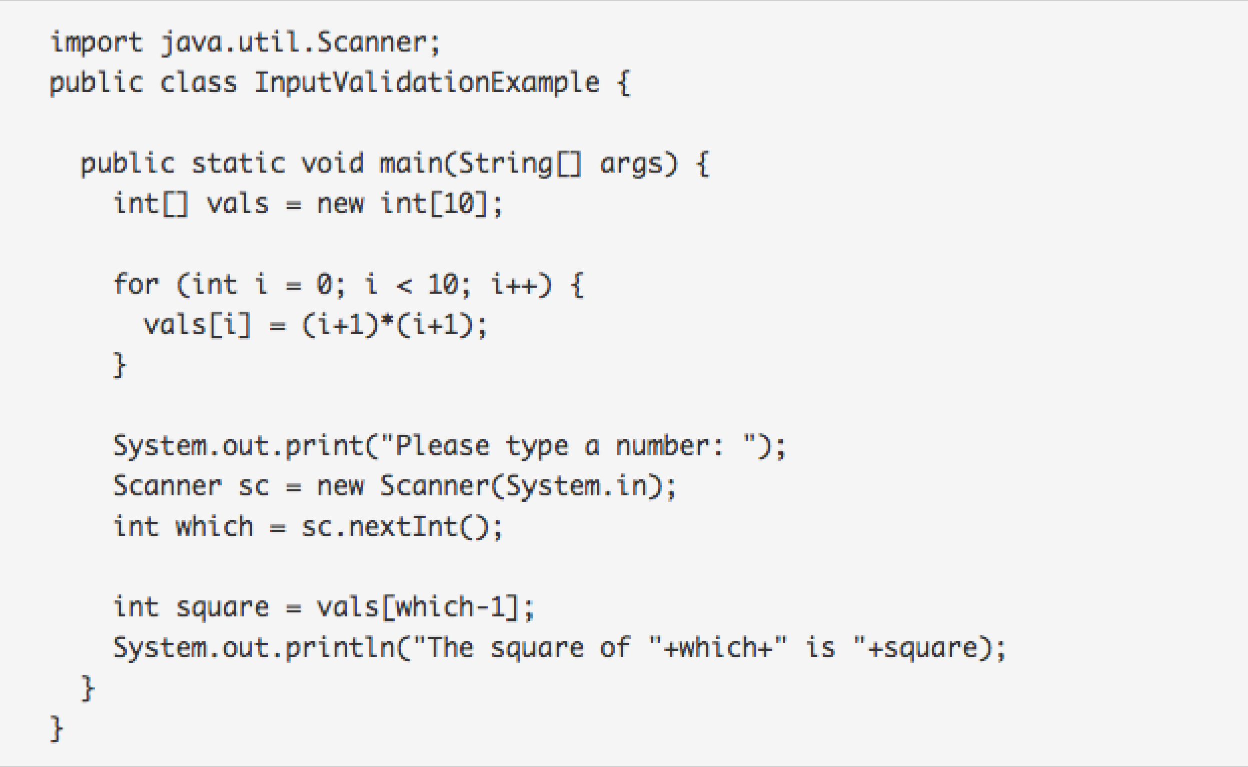
   where k is the packet length of a suspect packet in bytes, T is the length threshold, and (0,1) indicate that packet should or should not be marked as an attack, respectively. A packet with length p is marked as an attack if p < T. You are given the following data to use to design the algorithm.

   ➞ attack packet lengths: 1, 1, 2, 3, 5, 8

   ➞ non-attack packet lengths: 2, 2, 4, 6, 6, 7, 8, 9

   Find the true positive rate and the false positive rate for thresholds: 0-9. Draw the ROC curve (true positive rate on y axis, false positive rate on x axis)

2. Does open source coding and design help an attacker or does it help developers defend better? Justify your answer.
3. Describe a programming situation where least privilege strategy should be used to improve security.
4. Think how you can break this program and which OWASP secure coding techniques would make it better:



```
import java.util.Scanner;
public class InputValidationExample {

  public static void main(String[] args) {
    int[] vals = new int[10];

    for (int i = 0; i < 10; i++) {
      vals[i] = (i+1)*(i+1);
    }

    System.out.print("Please type a number: ");
    Scanner sc = new Scanner(System.in);
    int which = sc.nextInt();

    int square = vals[which-1];
    System.out.println("The square of "+which+" is "+square);
  }
}
```

5. Suggest a technique with which a browser can detect and block clickjacking attacks.
6. What attack is a financial institution seeking to counter by asking its customers to confirm that their expected security picture before entering sensitive data?
7. Suggest how to avoid cookie hijacking.
8. Given a network address and mask find the host number and network IP. Practice here: https://subnettingpractice.com/
9. Write a firewall or IDS rule for a specific attack signature. The syntax for Snort IDS will be given.