1. **Definitions:** Define any four of the following terms
   - Threat
   - Vulnerability
   - Risk
   - Malicious Logic
   - Audit
   - Sandbox(ing)

2. **Cryptography**
   (a) If a one-time pads are provably secure, why are they so rarely used in practice?
   (b) We know that public key systems can provide non-repudiation of origin. Now consider a secret key system, in which Alice and Bob share a secret key. Bob has a message that he claims came from Alice, and to prove it he shows both the cleartext messages and the ciphertext message. The ciphertext corresponds to the plaintext enciphered under the secret key that Alice and Bob share. Please explain why this does not satisfy the requirements of non-repudiation of origin?

3. **Authentication**
   (a) Does the `sum` program, which exclusive or's all words in its input to generate a one-word output, a good cryptographic hash function? Why or why not?
   (b) Does using passwords with a salt make attacking a specific account more difficult than using passwords without a salt? Please explain why or why not?

4. **Java Security**
   - List the components of the Java 2 Security model?
   - How does Java 2 security differ from JDK 1.0 and JDK 1.1?
   - How does Java 2 security mechanism prevent execution of malicious code?

5. **Audit**
   - Describe the principle of complete mediation?
   - Why is adherence to the above principle a necessity for logging file accesses?

6. **Malicious logic**
   (a) A computer system provides protection using the Bell-LaPadula policy. How would a virus spread if:
      - the virus was placed on the system at system low (the compartment which all other compartment dominate)?
      - the virus was placed on the system at system high (the compartment which dominates all compartments)?
   (b) Source code for a simple system utility `chfile` that updates files is listed below.
void foo(char *fname, char *pword) {
    char _fname[16];
    char _pword[16];
    int fd=0;
    strcpy(_fname, fname);
    strcpy(_pword, pword);
    fd= open(_fname, O_RDWR);
    write(fd, _pword, strlen(_pword));
    return;
}

int main(){
    char *p, *q;
    p= (char *)malloc(sizeof(char)*32);
    p= (char *)malloc(sizeof(char)*32);
    scanf("%s", p);
    scanf("%s", q);
    foo(p, q);
    return 0;
}

Listing 1: Code listing for program chfile

• Is it safe to run chfile setuid root? Please explain your answer.
• Is there any buffer overflow vulnerabilities in this program? If so, please illustrate the vulnerability—how it can be used to compromise system security—and describe a fix.

7. Intrusion Detection

Assume that you are working for a computer security company. Consider the following situation: You are using a user authentication program XYZ for a standalone machine not connected to the network. Hence only a single user can use it at once. The company that made this program has supplied the following information:

• At any given time XYZ could be in one of the following states: S1, S2, S3, and S4. The probability distribution with respect to time of S1:
• It is abnormal if the XYZ stays in S4 for more then an hour.
• The normal state flow of XYZ is S1 → S3 → S4 → S2 → S3 and back to S1
• The following function are also supplied:
  – State xyz - Return the state of XYZ at this moment.
  – Time() - Return system time 24-hour clock format.
  – Userinput - Returns what the username and password are separated by comma.
  – State_change - Returns 1 if XYZ has changed its state within the last second.
• They company states that if a user types “.?; =-25” as password then the program will crash and they are working on a fix.
They also state that XYZ program will compile, and run in its own memory space, any uncompiled program named id.c placed in /root/id.c when the system starts up.

You job is to create an ID system with the above information for XYZ.

(a) What ID models could be used to create the ID system for XYZ program, and why? Be specific.
(b) The company wants you to include a threshold metric model in your ID system. Do you think the above information will enable you to do this? Explain your answer.
(c) What kind of intrusion prevention mechanisms could you incorporate into your ID system?
(d) Design your ID system.