Homework 3

Due Date: **Tuesday, 10/21, 11:59pm**

Instructions

1. Unless otherwise specified, any assignment involving programming may be completed with the programming language of your choice. If asked, you should be able to explain the details of your source code (e.g. program design and implementation decisions).
2. You are bound by the Stevens Honor System. All external sources must be properly cited, including the use of LLM-based tools (e.g. ChatGPT). Your submission acknowledges that you have abided by this policy.
3. Solutions are accepted only via Canvas, where all relevant files should be submitted as a **single `.zip` archive that expands into a directory**. This directory should include your ***typed*** **answers as a `.pdf` file** and **source code of any programming used in your solutions** with the following structure:

   ```
   <lastname>_<firstname>.hw<#>/
   ├── solutions.pdf
   └── src/
       └── <source code files>
   ```

   Your `src` folder should document all necessary steps for reproduction. If we are unable to reproduce your results, you may lose credit.

   There is a python validation script you should use to confirm it is correct.

Problem 1 - (*20 pts*) Identifying Malware

Classify the following malware based on the categories covered in Ch. 23 of Bishop and justify your answer. List all categories that apply.

1. (*5 pts*)A program that runs as a background process without the user's knowledge, silently monitoring that user's outgoing network requests and forwarding a list of connected domains to an unknown server every 10 minutes.
2. (*5 pts*)A program that behaves like `ls`, but when use to display the contents of a directory that contains more than 50 files, encrypts the directory and displays a message telling the user to send bitcoin to a particular address to receive a decryption key.
3. (*5 pts*)A program that, when run, sends a copy of itself to every known host in the user's `ssh` config file via `scp`.
4. (*5 pts*)A code segment that, when executed, inserts itself into every program it can write to by patching that program so that the inserted code segment is executed before the program performs its functions.

Problem 2 - (*20 pts*) Capabilities and Trojans

Consider how a system with capabilities as its access control mechanism could deal with Trojan horses.

1. (*10 pts*) In general, do capabilities offer more or less protection against Trojan horses than do access control lists? Justify your answer in light of the theoretical equivalence of ACLs and C-Lists.
2. (*10 pts*) Consider now the inheritance properties of new processes. If the creator controls which capabilities the created process is given initially, how could the creator limit the damage that a Trojan horse could do?

Problem 3 - (*30 pts*) Jeremy from Marketing

Darknet Diaries is an investigative podcast, chronicling true stories about hackers, breaches, shadow government activity, hacktivism, and cybercrime.

Listen to the episode *Jeremy from Marketing* about the experience of a pentester hired by a company to pose as a new hire and see how much he can hack on his first week on the job.

While listening, try and connect Tinker's account to the principles of system design covered in class and answer the following questions:

1. Jeremy and Tinker discuss single sign-on. What principle covered in class does single sign-on violate and how? On the flip side, name a principle that single sign on helps systems to adhere to and how?
2. At one point, Tinker was able to capture some users' passwords using Responder but he still wasn't able to log in as those users. Explain the mechanism that was in place to prevent him from doing so and which principle of system design it adheres to.
3. Several times throughout his experience in trying to hack the company, Tinker found a vulnerability but was thwarted in exploiting it by a security mechanism implemented by the company. Find one example of this (disjoint from the two questions above) and specify: the vulnerability that Tinker found and the security mechanism that ensured the company's data still remained safe.

Problem 4 - (*30 pts*) Honeywords

Introduced by Juels and Rivest, *Honeywords* comprise a non-cryptographic method for hardening password security against stolen password files after successful breaches into authentication servers. The idea is to employ *decoy* passwords so that any user's account is associated with not only one (the real) password, but also with many fake ones, and then distribute password verification *across two servers*, say one red and one blue, each storing and verifying "half" of the credentials needed to be verified in order to successfully authenticate a user (see also Figure 1). Read about the Honeywords authentication system from the original paper, and answer the following questions.

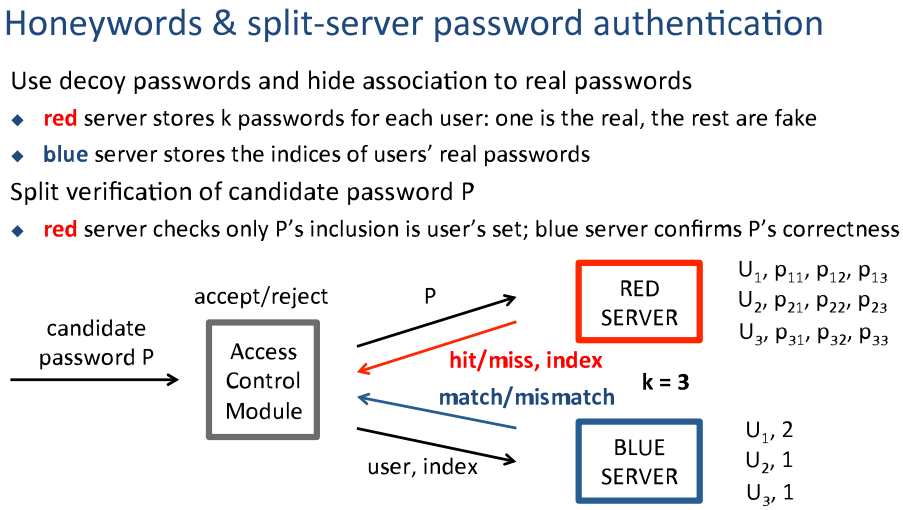

Figure 1: Hardening password security by employing decoy passwords in a split-server architecture.

1. (*10 pts*) How does this *split architecture* improve security? Consider the two cases where an attacker compromises one of only one of the servers.
2. (*10 pts*) How does this system make password cracking *detectable*.
3. (*5 pts*) What consititutes a *good* honeyword for a user whose real password is `Bo$tonRedSox76`, if honeywords are generated by tweaking real passwords? Provide a few examples (at least 3).
4. (*5 pts*) Stevens employs the honeywords authentication system, and you just stole the passwords list of an employee in the Office of the Registrar:
   - `00000000000000000000000000`
   - `itWb!%s45_3gMoI00286!*mooewTi409##21jUi`
   - `7304ASpaceOdyssey`
   - `2001ASpaceOdyssey`

   If you have *only one* chance to impersonate them (and try to increase your GPA), which password will you choose and why?