Assignment 7

1. **In this chapter we discussed access control lists (ACLs) and capabilities (aka C-lists).**
   1. Give two advantages of capabilities over ACLs.

* It is easy to add/delete users
* Easy to delegate, avoid confused deputy
  1. Give two advantages of ACLs over capabilities.
* Easy to change rights to a resource
* Good when users manage their own files
* Protection is data oriented

1. **What is the "need to know" principle and how can compartments be used to enforce this principle?**

The “need to know” principle is that people are only allowed to see information that they need to know in order to do their job. Compartments enforce the “need to know” principle by grouping data of the same security level into different compartments and people are only allowed to see information of an specific compartment even though they have the clearance of that security level. By using compartments we can restrict information flow and prevent people from seeing information beyond their needs.

1. **This problem deals with covert channels.**
   1. Describe a covert channel involving the print queue and estimate the realistic capacity of your covert channel.

A covert channel using the print queue, involves with the sender filling up the print queue and the receiver will be checking the print queue for information. If the sender fills up the print queue, it will mean the binary 1, and if the sender empties the print queue, then that means a 0 in binary. The receiver will only have to check if the print queue is full or empty at every interval. The speed capacity of this covert channel heavily depends on the speed of the printer at filling and emptying the queue, and the agreed interval time. Suppose the printer can handle print and deletion very fast, and the agreed interval time is 20 seconds, we would get a capacity of 3 bits per second, but this capacity can be doubled if we use an interval time of just 10 seconds and the printer can handle such work.

* 1. Describe a subtle covert channel involving the TCP network protocol.

A covert channel involving TCP network protocol works by using it’s reserved field and sequence number to hide data. If we have a server, as a sender we can use spoofing to spoof that we are the receiver and then send a tcp packet with hidden data on the sequence number, once the server receives it, it will send an acknowledge packet back to the “sender” but since we spoofed as the receiver, the receiver will be the one that receives the acknowledge packet and then he will be able to see the sequence/ack number that contains the hidden data.

1. **We briefly discussed the following methods of inference control: query set size control; N-respondent, k% dominance rule; and randomization.**
   1. Explain each of these three methods of inference control.

N-respondent, k% dominance rule is to not release data that if k% or more of the result is contributed by N ore fewer subjects. An example of this would be querying the census database for the average net worth of individuals in Bill Gate’s neighborhood, with any reasonable techinique setting for N and k, no results should be returned, because else it can help people infer the information.

Query set size control, is when no response is returned if the size of the set is too small, this would make it difficult for people to infer a professor’s salary, but in the case of medical research, it might also prevent or distort important research.

Randomization, it is to add small amount of random noise to the data, a method like this could would prevent medical research into rare medical conditions because the noise would affect the data.

* 1. Briefly discuss the relative strengths and weaknesses of each of these methods.

N-respondent, k% dominance, this method is strong because it returns no result at all when there are N or fewer subjects that dominates k or more percentage of the data, and by having a reasonable N and k, there could be no way for people to infer data. The weakness of this method is that for study of rare diseases in the medical field, rare diseases often does not meet the N amount and the data would never be shown.

Query set size control, this method’s strength is that it will prevent people from getting any data if the set of the query is too small, this would prevent people to infer data by taking advantage of small set of data. One of the weakness of this is that sometimes the small set of data are very important, for example if we are doing medical research on rare diseases, it wouldn’t be called “rare” if the set size is big, yet we are not allowed to see that information because the set size is too small.

Randomization, the strength of this method is that it allows information to be viewed without having to worry about the size of the set or the having N or less subjects dominate more than k%, but the weakness of this method is that it provides inaccurate information, and in medical research, information should be as accurate as possible.

1. **A botnet consists of a number of compromised machines that are all controlled by an evil botmaster.**
   1. Most botnets are controlled using the Internet Relay Chat (IRC)protocol. What is IRC and why is it particularly useful for controlling a botnet?

IRC stands for Internet Relay Chat, it is an application layer protocol that allows people to send message publicly (there is also one to one), IRC works on a client/server model, when the client sends a message, the message is sent to the server and then other client receives the message through the server. IRC is useful in botnet because IRC channels/servers are easy to find and with just sending one message, every botnet will be able to receive the message. In addition, using IRC allows the botmaster to hide their identity easily by using anonymous proxy/ip.

* 1. Why might a covert channel be useful for controlling a botnet?

Covert channels are useful in controlling a botnet because it would be illogical to specifically send instructions to the bots in a botnet, but instead if we use a covert channel, it uses are shared medium to make communication and all the bots in a botnet can access that medium for instructions. Additionally covert channels are used to go around firewalls or to avoid raising suspicion to network monitoring softwares.

* 1. Design a covert channel that could provide a reasonable means for a botmaster to control a botnet.

A good covert channel would for a botnet would be through twitter or linkedin, by configuring the bot to check on a twitter or linkedin status, we can issue commands to the whole botnet by changing the status on the tweeter or linkedin account. This covert channel works because twitter and linkedin are just websites and they go through HTTP port 80/8080 and it is accessible from most computers, and even if corporates block twitter, It is most likely that linkedin will not be blocked, therefore we can remotely issue commands.

1. **Ross Anderson claims that "Some kinds of security mechanisms may be worse than useless if they can be compromised".**
   1. Does this statement hold true for inference control? Why or why not?

No, because we can reduce the information that is being leaked, it makes the attackers job harder and prevent the leak of obvious information.

* 1. Does this hold true for encryption? Why or why not?

Yes, this holds true for encryption because if we are using a weak encryption, then it is most likely that the attacker will be able to break the encryption, but encrypting with a weak encryption, we are actually telling the attacker that the information is important, but if we send it unencrypted, the attacker would not know that the information is important, because they might be filtering out unencrypted messages because they think they are unimportant.

* 1. Does this hold true for methods that are used to reduce the capacity of covert channels? Why or why not?

No, because by reducing the capacity of covert channel we reduce the speed/amount of information that is being leaked and we make the attacker’s job harder.

1. **This problem deals with audio CAPTCHAs.**
   1. Describe an example of a real-world audio CAPTCHA and explain how this CAPTCHA works, that is, explain how a program would generate the CAPTCHA and score the result, and what a human would need to do to pass the test.

An example of a real-world audio CAPTCHA is google’s reCAPTCHA, google’s reCAPTCHA is just a normal CAPTCHA like all other services, it mainly test the user by using visual test, but it also provides an audio CAPTCHA. The way this CAPTCHA works is having a database of different voices (Robot, men, women, squirrel, etc.) saying a letter or number, the CAPTCHA program will randomly generate a sequence of letter and numbers and then randomly select the audio files of voices that say those letters, the program will then play all those voices according to the order of the sequence of letter and numbers, and then the human would need to hear the audio and type the letter and numbers they hear.

* 1. For the CAPTCHA in part a, what information is available to an attacker?

The information available to the attacker is the audio that is played by the CAPTCHA, other than that, the attacker could probably know the different distortions that are used to distort the voices. The attacker can also assume that the text contains only numbers and letters.

1. **The reCAPTCHA project is an attempt to make good use of the effort humans put into solving CAPTCHAs. In reCAPTCHA, a user is shown two distorted words, where one of the words is an actual CAPTCHA, but the other is a word-distorted to look like a CAPTCHA- that an optical character recognition (OCR) program was unable to recognize. If the real CAPTCHA is solved correctly, then the reCAPTCHA program assumes that the other word was also solved correctly. Since humans are good at correcting OCR errors, reCAPTCHA can be used, for example, to improve the accuracy of digitized books.**
2. It is estimated that about 200,000,000 CAPTCHAs are solved daily. Suppose that each of these is a reCAPTCHA and each requires about 10 seconds to solve. Then, in total, about how much time would be spent by users solving OCR problems each day? Note that we assume two CAPTCHAs are solved for one reCAPTCHA, so 200,000,000 CAPTCHAs represents 100,000,000 reCAPTCHAs.

According to the information given, there are 100,000,000 reCAPTCHAs solved daily and each of them take 10 seconds, therefore we use have 10\* 100,000,000 = 1,000,000,000 seconds daily, which equals to 16,666,667 minutes, or 277,778 hours or 11574 days or 31.69 years

1. Suppose that when digitizing a book, on average, about 10 hours of human effort is required to fix OCR problems. Under the assumptions in part a, how long would it take to correct all of the OCR problems created when digitizing all books in the Library of Congress? The Library of Congress has about 32,000,000 books, and we assume that every CAPTCHA in the world is a reCAPTCHA focused on this specific problem.

We have 32,000,000 books to digitalize and each takes 10 hours of effort to fix OCR, therefore it takes 320,000,000 hours of work to fix OCR problems, and since we spend 277,778 hours solving captchas, it would take:

320,000,000/277,778 = 1152 days = 3.156 years

1. How could Trudy attack a reCAPTCHA system? That is, what could Trudy do to make the results obtained from a reCAPTCHA less reliable?

To attack the reCAPTCHA system Trudy would have to first input the correct letters for the first image either manually or using a bot with advanced OCR, and then write gibberish on the second part of the CAPTCHA.

1. What could the reCAPTCHA developer do to minimize the effect of attacks on the system?

Developers could group the answers by ip addresses and then check if the answers given in the CAPTCHAs are reliable, or they could see how many CAPTCHAs were solved from the same IP address and ignore those that have done unreasonable amount of CAPTCHAs.

1. **It has been widely reported that spammers sometimes pay humans to solve CAPTCHAs.**
2. Why would spammers want to solve lots of CAPTCHAs?

Spammers would like to solve a lot of CAPTCHAS because they are required to solve the CAPTCHAs in order to be able to spam, create email addresses, etc.

1. What is the current cost, per CAPTCHA solved (in U.S. dollars), to have humans solve CAPTCHAs?

According to NYtimes, it costs around $0.80 to $1.20 per 1000 CAPTCHAs solved by humans.

1. How might you entice humans to solve CAPTCHAs for you without paying them any money?

In order to entice humans to solve CAPTCHAs for free, we have to give them a reason to solve them, one example given from the book is free porn, but it can also work with by having a website to provide free music.

REFERENCES:

25b. <http://www.nytimes.com/2010/04/26/technology/26captcha.html>

Classmates: Jeffrey Su, Jay Patel.