Homework 1 CS5920 Spring 2022

NathanBellew

Problem 1 CIA Triad

A.

With an automated teller machine a user is able to provide information to the machine which will connect to their bank in order to transfer funds. In doing so this process uses Confidentiality by encrypting data sent from an ATM maintaining that the users PIN is safe. The use of a PIN here allows for the Integrity of the user's bank account information, the funds allotted in the account cannot be changed unless the user has both the card and PIN. An ATM also extends the availability of the banking network for the individual, by allowing a larger network of banking machines, the user has more availability to the banking services. The most important piece of the ATM is confidentiality, limiting access to funds only if the user has both a bank card that is acceptable and a PIN number that matches. These two pieces of information can be stolen and if confidentiality was not used by the system, any user of any bank could be compromised at any time. Further Integrity is the next important piece of the Triad, because the funds of a user's personal bank account should not be accessed by any other than the user. Maintaining integrity of both the data being transferred as well as the amount of funds being requested is a crucial part of the banking system. While Availability is last, the need for funds using an ATM is at most a convenience and can be solved by using a normal teller.

B.

A telephone switching system that routes calls through a switching network needs to focus mostly on integrity. Integrity in telephone network would ensure that the caller always reaches the correct telephone number and this number cannot be switched to a different telephone. By ensuring Integrity in a telephone connection, the user's connection cannot be interfered with and neither the caller nor recipient's current number change. The second most important piece of the triad is confidentiality, this ensures that the conversation held between both callers is kept secret and encrypted so that if any important information is shared, it is only shared between the two. Availability ensures that the call is always maintained and both callers are able to adequately understand each other, this can be a security issue if the call pertains to some important information such as a classified call between the Air Force and the Navy. Availability does not affect security in the same way as a lack of integrity would, a lack of integrity could allow for the telephone call to become hijacked and the caller may be connected with another source all together. I argue that it is more important than Confidentiality because if the destination call is changed or modified in such a way as to add another person, then the need for Confidentiality would no longer matter, as all the information would be shared decrypted to the new recipient.

C.

Any time security relates to a device that is needed for personal health, Availability is almost always the most important factor. The number one goal in security is to preserve and maintain human life, any time there is a medical device that requires an emergency response, availability is crucial. Availability is used to ensure that the piece of software or hardware is always accessible to the user, which for a medical device needs to be at any time the user may need it. The second and almost as important piece of the triad is integrity, because this is a medical device the data is incredibly important to the health of the user. If any piece is modified it could effect the output of the medical device which could have negative or even deadly consequences against the user. Confidentiality is also important to medical devices, the ability to connect to a device which uses a remote programmer to perform treatment settings should always have strong confidentiality. Without confidentiality, the medical devices could be easily accessed by other users there these other users can take full control of the system.

Problem 2 Security News

MoonBounce: The Dark Side of UEFI Firmware

MoonBounce is a firmware rootkit that was discovered at the end 2021, the rootkit has been detected by Kaspersky's Firmware Scanner. The attacks seem to be associated with APT41, a threat actor that has been widely reported to be Chinese. The targets seem to be focused mostly on companies and organizations that deal with advanced technology transportation. The MoonBounce firmware is a tampered version of a UEFI firmware made to embed a malicious code. The reason it took so long to discover this is due to the fact that the code itself lives in the SPI flash, which is located on the motherboard instead of the hard disk, this makes it harder to detect and allows it to affect the system even if the operating system is reinstalled. MoonBounce is able to install and deploy user-mode malware which can download malicious payloads from the internet, in doing so it does not leave traces on the hard drive, facilitating a fileless attack with a small footprint. Although firmware rootkits are not new, older rootkits generally are more easily detectable, this points to an evolution in both firmware rootkit complexity and a highly funded attacker. Kaspersky managed to track down the user logs of the attackers after they had gained a foothold on the network. It appears the attackers are trying to earn a permanent foothold on the network, by modifying the Active Directory domain and creating a permanent and invisible remote command connection. Although MoonBounce is not inherently malicious, the attackers are able to carry out attacks unseen due to the camouflage provided by the MoonBounce firmware.

MoonBounce is a threat to primarily system integrity and authenticity. MoonBounce does not directly effect confidentiality, the goal of MoonBounce is to have a backdoor into a system that allows for the discrete download of malicious software, that software is then used to compromise confidentiality. In addition, the goal of MoonBounce is not to break the system beyond use like in a denial of service attack, instead it is meant to be an invisible agent that lies within the system even after re-installing the operating system. MoonBounce is a threat to integrity because the system data can be altered or new data can be added without the consent of the user and it can be done in a way in which the user would never know the attack is occurring. In addition MoonBounce is able to authenticate malicious software and allow the software to be installed on the system remotely without leaving logs in the hard disk memory. Because MoonBounce can circumnavigate the users inputs and actively hide modifications made to the system, it is a major threat to the integrity of not only the active system but also any network the system is connected to.

MoonBounce: The Dark Side of UEFI Firmware

Problem 3. Affine Caesar Cipher

$$C = E([a,b]p) = (ap+b) mod 26$$

A.

The value of b does not have any limitations and can be completely arbitrary. Because the decryption function is the inverse of the encryption function and a is a coprime of mod m (in this case 26) then b should be removed automatically in the decryption process regardless of the number. This means that if b is some huge number the decryption process would not be effected no matter how large or small b is. Proof: E = Encryption D = Decryption

$E(x) = (ax+b) \mod m$
$D(x) = a^{-1}(x-b) \mod m$
$1 = aa^{-1} \mod m$

So Therefore

$D(E(x)) = a^{-1}(E(x)-b) \mod m$
$D(E(x)) = a^{-1}(((ax+b) \mod m)-b) \mod m$
$D(E(x)) = a^{-1}(ax+b-b) \mod m$
$D(E(x)) = a^{-1}ax \mod m$
$D(E(x)) = x \mod m$

Here you can see the b is canceled out prior to the evaluation of the a and inverse a, meaning even if the a does not follow the restrictions b will still be canceled out, therefore b has no limitations. ### B.

In order to produce distinct mapping for the Affine Caesar cipher, b is limited such that $ap+b \neq m$. So assuming a is a coprime of m the number created by

ap will not equal m, to ensure distinct mapping $ap + b$ must also not equal m.

C.

The numbers that a could possibly be are restricted so that a is coprime with m, in the Affine Caesar cipher m is 26, therefore the only numbers that a could be is: 1, 3, 5, 7, 9, 11, 15, 17, 19, 21, 23, and 25.

D.

The value of a is restricted to the coprime of m, this is because if it is not the coprime of m then the Cipher is no long one-to-one which could make it impossible to decrypt because there would be multiple solutions.

E.

As shown above there is a limited number of coprime numbers less than 26 and each of the values can have an addition shift from the b value. The a key can have 26 different addition shifts, because even if b is in the thousands the mod will bring that number down within the scope of 26 possible letters. So with 12 possible coprimes and 26 shifts the number of possible keys is $12 * 26 = 312$ possible keys that are both one-to-one and distinct.

Problem 4. Playfair Cipher

A.

Message: Attend crypto class and stay healthy

Step 1 : At te nd cr yp to cl as sa nd st ay he al th yq

Step 2 : Kp ok lg fp ar op di sy ys lg kq sk ob si on sr

Result : **Kpoklg fparop disyy slg kqsk obsionsr**

B.

Message easykey
step 1 ea sy ke yq
Step 2 as yk ea sr
Result asykeasr

C.

The solution clearly illustrates the main issue with the Playfair cipher, it can obfuscate some phrases but there will always exist edge cases where some letters within the initial message could repeat. This means that if the message contains letters that are all within the same row or column, the message will not be obfuscated to the same level as a longer message with a more diverse set of letters. ### D.

In order to calculate the total number of possible keys that can be produced by the Playfair Cipher, a factorial is needed. Because the Playfair Cipher is 5 x 5 then it can be determined that the total number of keys is:

$$25! \approx 2^{84}$$

E.

The standard Playfair Cipher is a 5x5 square, to calculate all possibilities including possible repeats it would be factorial 25 or 25! but because we only want to know distinct keys the formula changes to:

$$\frac{25!}{5^2} \approx 2^{79}$$