Exercise-3

Vineeth Mylavarapu
ID.801337050
College of Computing and Informatics

1. We have a symmetric encryption algorithm EK(M)=C. Here K is the secret key, M is the plaintext, and C is the ciphertext. We (and the attacker) know that the key length is 192 bits. The attacker eavesdrops on the communication line and gets a copy of the ciphertext C1. Now the attacker decides to conduct the brute force attack and try every possible key to get the plaintext M1. Let us assume that there is only one possible M1 and if the attacker sees it, he will know that this is the correct one. The attacker has 1,000,000 machines, with each machine having the capabilities to try 5,000,000 decryptions of C1 with different keys per second. If one machine finds the right key, it will automatically notify the attacker. Now please answer, how many years (roughly) does the attacker need to try 50% of the keys?

Note that Google has around 2 million servers. Also, check the Internet and see what the expected lifetime of the Sun is. Can you crack the key before that?

A. Given,

Key length = 192bits.
Using a brute force attack, the attacker needs to try 2^{192} different possible keys.

50% of kyes =2^192/2 = 2^191

The attacker maintains 1,000,000 machines each of them with capability to try 5,000,000 keys per second. Therefore, total capabilities of all the machines = 1,000,000 x 5,000,000 = 5,000,000,000,000 = $5x10^{12}$.

So, time required to try 50% of keys = (2^191seconds/5x10^12) Secs

= $6.277x10^{44}$ seconds

Converting the above into years we got

Time Required = $1.99x10^{37}$ years

Expected lifetime of the Sun = 5 billion years.

Hence, even with the technical skills of the attacker, it's not possible to crack 50% of the keys. Sometimes, the attacker can find the correct key within those 50% of the keys which is lucky or unexpected.

2. Let us consider the block cipher type of symmetric encryption. The basic idea is that you have a block of plaintext (for example, 128 bits) and a key (for example, 128 bits) as inputs to the encryption algorithm. The output will be a block of cipher-text (for example, 128 bits). If the encryption algorithm does not conduct compression, the output block size will be at least as large as the plaintext block (in other words, the cipher text is of the same size or longer than the plaintext.) Please explain why it is like this.

A. In a block cipher symmetric encryption, the plaintext is divided into fixed-size blocks, and each block is then encrypted using a key to produce a ciphertext of the same size as the plaintext. This is mandatory because the decryption process should be the same size of ciphertext as the original plaintext in order to accurately reverse the encryption operations and restore the original data. The algorithm uses mathematical operations to change the plaintext into a cipher form, but the size of the ciphertext must remain the same to ensure the confidentiality and integrity of the data being encrypted.

3. Bob has a public-private key pair (pub_Bob, pri_Bob). Alice needs to send some information to Bob. She wants to make sure that when Bob opens the message, he can verify that this is from Alice but not anyone else. So, she sends out the message as: [Alice, Epub_Bob(message)] to Bob. Basically, she first sends out her name in clear text, then encrypts the message with Bob's public key. Please discuss, can an attacker M send out a packet in Alice's name? How can he do it? Here we assume that M also has the public key of Bob. For the same question, if Alice sends out [Epub_Bob (Alice, message)], can M still impersonate Alice? (Here Alice puts her name in the encryption.)

A. As per public key cryptography, attacker M cannot send a packet in Alice’s name simply by Bob’s public key, whereas private key is used for decrypting messages that are encrypted with the respective public key.

Scenario 1:
In this scenario, Alice sends [Alice, Epub_Bob(message)] and M would not be able to encrypt a message with Bob's public key and impersonate Alice. This is because the clear text of Alice's name is still visible, and Bob can easily catch that the message is not from Alice if the encryption does not match the encryption for a message from Alice. So, M can impersonate Alice if M sends Epub_Bob (Alice, message) this message which Bob cannot detects.

Scenario 2:
In the second scenario, Alice sends [Epub_Bob (Alice, message)], it is even more difficult for M to impersonate Alice. This is due to M would have to not only encrypt the message with Bob's public key, but also correctly encrypt the name of Alice along with the message. This would require M to have access to Alice's private key, which should not be possible given that public key cryptography is designed to keep private keys secret.

4. The public key infrastructure (PKI) needs to handle the revocation of compromised keys. Currently, there are two basic approaches. The first one uses the certificate revocation list (CRL). The CRL is published periodically (for example, 8:00am every day). It contains the public key certificates that have been compromised. Another approach is to use Online Certificate Status Protocol (OCSP). Please study how OCSP works. Then write about 0.5 page to discuss the working procedure of OCSP, and the advantages and disadvantages of OCSP over the CRL.

A. Online Certificate Status Protocol (OCSP) is an internet protocol used to determine the status of a public key certificate. Unlike the Certificate Revocation List (CRL) approach, which requires clients to periodically check a list of revoked certificates, OCSP allows clients to check the status of a specific certificate in real-time. In OCSP, a client sends a request to the OCSP responder, which is an entity trusted by the certificate authority (CA), to verify the status of a certificate. The OCSP responder performs a lookup in its database and returns a response indicating whether the certificate is valid, revoked, or unknown. The response is signed by the OCSP responder to ensure authenticity and prevent tampering.

One of the main advantages of OCSP over CRL is that it provides real-time certificate revocation information, reducing the time it takes for clients to detect a compromised certificate. This is particularly important in situations where immediate action is required to prevent security breaches. Additionally, OCSP is more scalable and efficient than CRL, as it only requires clients to check the status of a specific certificate, rather than downloading and checking a large list of revoked certificates.

However, OCSP also has some disadvantages. It requires an active internet connection, and if the OCSP responder is unavailable, clients cannot obtain certificate status information. This can result in potential security breaches if a certificate has been revoked, but the client is unable to detect this due to the unavailability of the OCSP responder. Additionally, OCSP can also generate a high amount of network traffic, as each certificate must be checked individually, and this can put a strain on network resources.

5. David is a lobbyist, and he is secretly visiting different states for a marketing plan. Every midnight, David will send an email to Bob, who is his supervisor, to report the two states that he will visit tomorrow. To protect the information, David will encrypt the message with Bob's public key. For example, if David will visit North Carolina and South Carolina tomorrow, the message he sends to Bob will look like (NC, SC) pub-Bob. (Which means the short names of the two states encrypted by the public key of Bob.)
    - A reporter, Alice, is following the secret plan. Alice gets a copy of Bob's public key, but she does not know the private key of Bob. One night, Alice uses her laptop to eavesdrop on the message that David sends to Bob. She gets a copy of the encrypted message. Please illustrate how Alice can use forward search to figure out which two states David will visit tomorrow. Hint: this is an example of forward search attack.

A. This method of using a forward search to crack an encrypted message is not a practical or efficient way to do so. In this scenario, even if it is possible, the number of possible combinations would be much greater than 50, so a brute force search with only 50 combinations would be highly unlikely to succeed in cracking the encryption.

   Additionally, the encryption performed using Bob's public key is designed to be secure against eavesdropping and tampering, making it computationally infeasible to break, even with a brute force search. Hence, it is unlikely that Alice would be able to use a forward search attack to figure out which two states David will visit tomorrow, even if she has a copy of the encrypted message and Bob's public key.

6. Question.6

A. When a user tries to withdraw money from a bank's ATM machine, the machine first checks if the account number provided is listed in the bank's valid account numbers. Then, the ATM machine encrypts the PIN entered by the user and compares it to the encrypted version stored on the user's card to verify its authenticity. A flaw in this approach is that the system does not correlate the PIN with the corresponding account number.

   If a dishonest customer, M, has obtained the account number of another customer, C, M could create or modify a card with C's account number and M's PIN, represented as AcNr(C) and E_B(PIN(M)). In this scenario, if the modified card is used at the ATM, the machine will verify that the account number is valid and that the PIN stored on the card matches the one entered by the user. As a result, person M would be able to successfully withdraw money from C's account without having knowledge of C's actual PIN.

7. Alice's computer stores the files in the following way: for every file F, the computer will calculate the hash value of the file hash(F) and store it after the file. Every time when Alice login, the machine will automatically hash all the files and compare the results to the stored hash values. In this way, if by accident the hard drive is mis-functioning and flips a few bits in a file, Alice can immediately detect it since the hash value will be different. Now an attacker hacks into Alice's machine and he tries to change several files. The attacker also knows the hash function that the computer uses. Please describe what the attacker needs to do so that the next time Alice login, the machine will not detect the changes. Also, please discuss how we should improve the mechanism to detect such changes.

A. Below are pointers that attacker should perform next time:
    - calculate the hash value of the changed files.
    - replace the original hash value stored after the file with the newly calculated hash value.

    In this way, when Alice logs in, the computer will calculate the hash value of the files and compare it with the stored hash values. Since the attacker has changed the stored hash values to match the new values calculated from the changed files, the computer will not detect any changes.

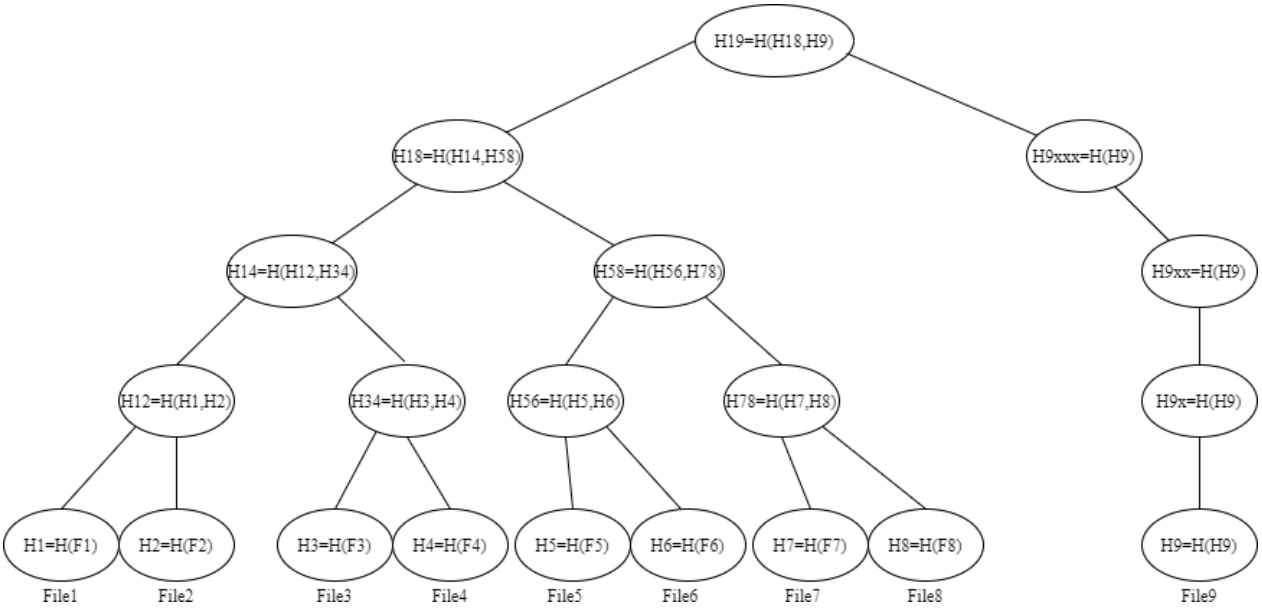
    To improve the mechanism:

    - use a digital signature.
    - verification using the corresponding public key to ensure the authenticity and integrity of the files.

    Additionally, to further enhance security, the private key used for signing the files can be stored in a secure location, separate from the computer, to prevent unauthorized access.

8. Please draw a binary Merkle's hash tree with 9 leaf nodes. The leaf nodes are labeled as Leaf1 to Leaf9, which correspond to the hash values of the files F1 to F9, respectively. Now please answer:
    a. For each node in the tree, please label clearly how the hash value is calculated based on its children; Please note that the number of leaf nodes is not power of 2. Therefore, you may need to change the way in which intermediate nodes are calculated. There are different ways to handle this. Please label clearly how you calculate each node.
    b. Now the creator of the file F6 needs to verify that his file's hash value is integrated in the root of the tree. Please show the minimum number of hash values in the tree that the creator needs to accomplish the task. Please also show how the verification is accomplished.

A.



In order to check if the hash value of F6 is included in the root hash H19, the creator needs to obtain the minimum number of hashes required. To calculate H19, the hashes of H18 and H9 are needed. In turn, to determine H18, the hashes of H14 and H58 are required. The hash value of H58 can be determined by using H56 and H78, and to calculate H56, the hash value of H5 is required. Therefore, the minimum number of hash values the creator needs to verify the integration of F6 into the root hash H19 are H5, H78, H14, and H9.