Date
Page

HW-1

Ques. What is vulnerability?

A vulnerability is a weakness in an information system, system security procedures, internal controls, or implementation that can be exploited by cybercriminals to gain unauthorized access to a computer system. After exploiting a vulnerability, a cyberattack can run malicious code, install malware, and even steal sensitive data.

A condition that enables a threat event to occur.

However, vulnerability and risk are not the same thing, which can lead to confusion. Think of risk as the probability and impact of a vulnerability being exploited. If the impact and probability of a vulnerability being exploited is low, then there is low risk. Inversely, if the impact and probability of a vulnerability being exploited is high, then there is a high risk.

Every vulnerability is not exploitable. A vulnerability with at least one known, working attack vector is classified as an

exploitable vulnerability. The window of vulnerability is the time from when the vulnerability was introduced to when it is patched.

Ques. What is CVE and NVD?

CVE :- Common Vulnerabilities and Exposures (CVE) is a list of publicly disclosed vulnerabilities and exposures that is maintained by MITRE. CVE is simply a list of all publicly disclosed vulnerabilities that includes the CVE ID, a description, dates, comments and at least one public reference - for publicly known cybersecurity vulnerabilities.

NVD :- The National Vulnerability Database (NVD) is a database, maintained by NIST, that is fully synchronized with the MITRE CVE list. The NVD database includes all disclosed vulnerabilities, and includes a corresponding CVSS score. The CVE list feeds NVD, which then builds upon the information included in CVE Records to provide enhanced info for each record such as fix info, severity scores, and impact ratings.

As part of its enhanced info, NVD also provides advanced searching features such as by OS; by vendor name, product name, and/or version number; and by vulnerability type, severity, related exploit range, and impact.

Both CVE and NVD are sponsored by the US Federal Government and are available for free use by anyone.

Ques. Describe one current vulnerability (within last 6 months) and list out components affected by it.

NVD vulnerabilities :-
CVE-2022-37177 - HireVue-Broken-Or-Risky-Cryptographic-Algorithm

When visiting the interview invite link that HireVue provides for an interview, you are directed to a terms page where you agree to the terms before beginning the recorded and timed interview process. This initial HTTP response includes the interview questions that will be asked during the interview encoded using a rail fence cipher. This is easily decrypted which provides access to all of the interview questions before actually starting the process. If you do not agree to the terms

then the interview does not start so you can visit the link then leave as many times as you like until you agree to start the interview. HireVue's customers do not expect the interview questions to be known before the interview starts and an attempt at hiding them using a rail fence cipher is there but is a week form of encryption.

Rail fence cipher :- The rail fence cipher (also called a zigzag cipher) is a form of transposition cipher. It derives its name from the way in which it is encoded.

eg: encryption:

I/p = "GeeksforGeeks"

K = 3

o/p = GsGseKfreK eoe

decryption:

I/p = GsGseKfreK eoe

K = 3

O/p = "GeeksforGeeks"

In a transition cipher, the order of the alphabets is re-arranged to obtain the cipher text.

- In the rail cipher, the plain-text is written downwards and diagonally on successive rails of an imaginary fence.
- When we reach the bottom rail, we traverse

upwards moving diagonally, after reaching the top rail, the direction is changed again. Thus the alphabets of the message are written in a zig-zag manner.

- After each alphabet has been written, the individual rows are combined to obtain the cipher-text.

As we have seen, the number of columns in a rail fence cipher remains equal to the length of plain-text message. And the key corresponds to the number of rails.

- Hence, rail matrix can be constructed accordingly. Once we've got the matrix we can figure-out the spots where texts should be placed (using the same way of moving diagonally up and down alternatively).
- Then, we fill the cipher-text row wise. After filling it, we traverse the matrix in zig-zag manner to obtain the original text.

HW-2

Encrypted Text :- uibpmuibqkauvekuvaqlmzqvoittxcvaaaqj qtqbqmaeqttjmquxwzbivbqvbpqakw czam

1. Using brute force approach for decryption.

K=1 :- thaolthapjzhukjvuzpklypunhssvvzzpipspa plzdpssidptwvyahuapuaopzjvbyzl

K=2 :- sqznksqzoiyqtzjiutyojkxotmgrrvuyyoh orozokycorrhkosvuxzqtzotznoyiua xyk

K=3 :- ~~xlespxletvd~~ rfymjrfynhxfsihtsxnijuenslfq qutxxngnqnynjxbnqqqjnrutwyfsynsymn xhtzwxj

K=4 :- qexliqexmgwerhgsruvnhivmrkepptsuuvmf mpmxmiwampfimqtsvxerxmrxlmwqsyvui

K=5 :- pdwkhpdwlfvdqqfrqvlghulqjdoosrvvle lolwlhvzlooehlpsruwdqwlqwklvfrxu vh

K=6 :- ocvjgocvkeucpfeqpukfgtkpicnnrquukdk nkvkguykvnndqkorqtvcpvkpvjkueqwtuq

K=7 :- nbuifnbujdtboedpotjefsjohbmmqptt jcjm
jujftxjmmcfjnqpsuboujouijtdpvstf

K=8 :- mathematicsandconsideringallpossibilitieswill
be important in this course

2. Using frequency analysis method for decryption.

most frequent alphabet = "q"
in the given text is

most frequent alphabet = "e"
in the english language is

=> q - e = 12

2nd most frequent letter = "t"

=> t - q = 13

3rd most frequent letter = "a"

=> a - q = 16

4th most frequent letter = "o"

=> o - q = 2

5th most frequent letter = "i"

=> i - q = 8

RSA algo.

→ ~~we cannot~~ it is difficult to ~~factorize a large~~ integer.

→ encryption strength totally lies on the key size and if we double or triple the size, the strength of encryption increases exponentially.

→ Length of RSA keys can be typically 1024 or 2048 bits long.

Public Key :- — considering two co-prime no.

$P = 53$ & $Q = 59$

$n = 53 * 59 = 3127$

exponent e = integer

$1 < e < \phi(n)$

$\phi(n) = (p-1)(q-1)$

$= 3016$

consider $e = 3$

∴ public key = $n = 3127$, $e = 3$

Private Key :- $d = [K * \phi(n) + 1] / e$

K = any integer

consider $K = 2$

$d = 2011$

∴ private key = 2011

$H = 8$ $I = 9$

Encrypted data = $(8\ 9)^e \bmod n$

$(89)^3 \bmod 3127 = 1394$

decrypted data = $c^d \bmod n$

$= (1394)^{2011} \bmod 3127$

$= 89$

HW-3

Symmetric encryption uses a single Key to encrypt and decrypt. If you encrypt a zip file, then decrypt with the same Key, you are using symmetric encryption. Symmetric encryption is also called "Secret Key" encryption because the Key must be kept secret from third parties. Strengths of this method includes speed and cryptographic strength per bit of key; however, the major weakness is that the key must be securely shared before two parties may communicate securely.

Symmetric encryption may have stream and block modes. Stream mode means each bit is independently encrypted in a "stream". Block mode ciphers encrypt blocks of data each round; for example, 64 bits for the Data Encryption Standard (DES), and 128 bits for AES. Some block ciphers can emulate stream ciphers by setting the block size to 1 bit; they are still considered block ciphers.

Initializing vectors & Chaining

Some symmetric ciphers use an initialization vector to ensure that the first encrypted block of data is random. This ensures that identical plaintexts encrypt to diff ciphertexts.

And in this case at worst, the two message that begin the same ~~exact~~ way will encrypt only up to first difference.
Chaining (called feedback in stream mode) feeds the previous encrypted block into the next block ~~into the block~~ ready for encryption. This destroys patterns in the resulting ciphertext ~~.DES~~

Modes of DES:-
DES can use five different modes to encrypt data. The modes primary difference is block versus emulated stream, the use of initialization vectors, and whether errors in encryption will propagate to subsequent blocks.

The five modes are:-

- Electronic Code Book (ECB) - It is the original mode of DES, ~~ECB~~ ECB is the simplest and weakest form of DES. It uses no initialization vector or chaining. Identical plaintexts with identical keys encrypt to identical ciphertexts. Two plaintexts with partial identical portions, such as the header of a letter, encrypted with the same key will have partial identical ciphertext portions.

- Cipher Block Chaining (CBC) – CBC mode is a block mode of DES that XORs the previous encrypted block of ciphertexts to the next block of plaintext to be encrypted. The first encrypted is an initialization vector that contains random data. This "chaining" destroys patterns. One limitation of the CBC mode is that encryption errors will propagate; an encryption error in one block will cascade through subsequent blocks due to the chaining, therefore destroying their integrity.

- Cipher Feedback (CFB) – CFB mode is very similar to CBC, but the primary difference is that CFB is a stream mode. It uses feedback, which is the name for chaining when used in stream modes, to destroy patterns, and so errors propagate. CFB uses an initialization vector & destroys patterns, and so errors propagate.

- Output feedback (OFB) – OFB mode differs from CFB in the way feedback is accomplished. CFB uses the previous ciphertext for feedback. The previous ciphertext is the subkey XORed to the plaintext. OFB uses the subkey before it is XORed to the plaintext. Since, the subkey is not affected by encryption errors, errors will not propagate.

- Counter (CTR) - CTR mode is like OFB; the difference again is the feedback. CTR mode uses a counter, so this mode shares the same advantage as OFB in that patterns are destroyed and errors do not propagate. However, there is an additional advantage: since the feedback can be as simple as an ascending number, CTR mode encryption can be executed in parallel.