# ****Private Key:****

In the Private key, the same key (secret key) is used for encryption and decryption. In this key is symmetric because the only key is copied or shared by another party to decrypt the cipher text. It is faster than public-key cryptography.

# ****Public Key:****

In a [Public key](https://www.geeksforgeeks.org/public-key-encryption/), two keys are used one key is used for encryption and another key is used for decryption. One key (public key) is used to encrypt the plain text to convert it into cipher text and another key (private key) is used by the receiver to decrypt the cipher text to read the message.

# Caesar Cipher Technique

The Caesar cipher is the simplest and oldest method of cryptography. The Caesar cipher method is based on a mono-alphabetic cipher and is also called a shift cipher or additive cipher. Julius Caesar used the shift cipher (additive cipher) technique to communicate with his officers. For this reason, the shift cipher technique is called the Caesar cipher. The Caesar cipher is a kind of replacement (substitution) cipher, where all letter of plain text is replaced by another letter.

Let's take an example to understand the Caesar cipher, suppose we are shifting with 1, then A will be replaced by B, B will be replaced by C, C will be replaced by D, D will be replaced by C, and this process continues until the entire plain text is finished.

Caesar ciphers is a weak method of cryptography. It can be easily hacked. It means the message encrypted by this method can be easily decrypted.

Plaintext: It is a simple message written by the user.

Ciphertext: It is an encrypted message after applying some technique.

The formula of encryption is:

En (x) = (x + n) mod 26

The formula of decryption is:

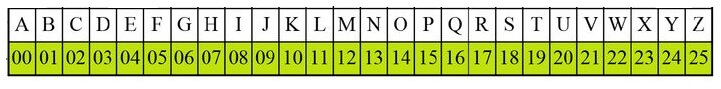
Dn (x) = (xi - n) mod 26

If any case (Dn) value becomes negative (-ve), in this case, we will add 26 in the negative value.

Where,

E denotes the encryption  
D denotes the decryption  
x denotes the letters value  
n denotes the key value (shift value)

Note: "i" denotes the offset of the ith number of the letters, as shown in the table below.



Example: 1 Use the Caesar cipher to encrypt and decrypt the message "JAVATPOINT," and the key (shift) value of this message is 3.

Encryption

We apply encryption formulas by character, based on alphabetical order.

The formula of encryption is:

En (x) = (x + n) mod 26

|  |  |  |
| --- | --- | --- |
| Plaintext: J → 09 | En: (09 + 3) mod 26 | Ciphertext: 12 → M |
| Plaintext: A → 00 | En: (00 + 3) mod 26 | Ciphertext: 3 → D |
| Plaintext: V → 21 | En: (21 + 3) mod 26 | Ciphertext: 24 → Y |
| Plaintext: A → 00 | En: (00 + 3) mod 26 | Ciphertext: 3 → D |
| Plaintext: T → 19 | En: (19 + 3) mod 26 | Ciphertext: 22 → W |
| Plaintext: P → 15 | En: (15 + 3) mod 26 | Ciphertext: 18 → S |
| Plaintext: O → 14 | En: (14 + 3) mod 26 | Ciphertext: 17 → R |
| Plaintext: I → 08 | En: (08 + 3) mod 26 | Ciphertext: 11 → L |
| Plaintext: N → 13 | En: (13 + 3) mod 26 | Ciphertext: 16 → Q |
| Plaintext: T → 19 | En: (19 + 3) mod 26 | Ciphertext: 22 → W |

The encrypted message is "MDYDWSRLQW". Note that the Caesar cipher is monoalphabetic, so the same plaintext letters are encrypted as the same letters. For example, "JAVATPOINT" has "A", encrypted by "D".

Decryption

We apply decryption formulas by character, based on alphabetical order.

The formula of decryption is:

Dn (x) = (xi - n) mod 26

If any case (Dn) value becomes negative (-ve), in this case, we will add 26 in the negative value.

|  |  |  |
| --- | --- | --- |
| Ciphertext: M → 12 | Dn: (12 - 3) mod 26 | Plaintext: 09 → J |
| Ciphertext: D → 03 | Dn: (03 - 3) mod 26 | Plaintext: 0 → A |
| Ciphertext: Y → 24 | Dn: (24 - 3) mod 26 | Plaintext: 21 → V |
| Plaintext: A → 00 | En: (00 + 3) mod 26 | Ciphertext: 3 → D |
| Plaintext: T → 19 | En: (19 + 3) mod 26 | Ciphertext: 22 → W |
| Plaintext: P → 15 | En: (15 + 3) mod 26 | Ciphertext: 18 → S |
| Plaintext: O → 14 | En: (14 + 3) mod 26 | Ciphertext: 17 → R |
| Plaintext: I → 08 | En: (08 + 3) mod 26 | Ciphertext: 11 → L |
| Plaintext: N → 13 | En: (13 + 3) mod 26 | Ciphertext: 16 → Q |
| Plaintext: T → 19 | En: (19 + 3) mod 26 | Ciphertext: 22 → W |

The decrypted message is "JAVATPOINT".

Example: 2 Use the Caesar cipher to encrypt and decrypt the message "HELLO," and the key (shift) value of this message is 15.

Encryption

We apply encryption formulas by character, based on alphabetical order.

The formula of encryption is:

En (x) = (x + n) mod 26

|  |  |  |
| --- | --- | --- |
| Plaintext: H → 07 | En: (07 + 15) mod 26 | Ciphertext: 22 → W |
| Plaintext: E → 04 | En: (04 + 15) mod 26 | Ciphertext: 19 → T |
| Plaintext: L → 11 | En: (11 + 15) mod 26 | Ciphertext: 00 → A |
| Plaintext: L → 11 | En: (11 + 15) mod 26 | Ciphertext: 00 → A |
| Plaintext: O → 14 | En: (14 + 15) mod 26 | Ciphertext: 03 → D |

Note that the Caesar cipher is monoalphabetic, so the same plaintext letters are encrypted as the same letters. Like, "HELLO" has "L", encrypted by "A".

The encrypted message of this plain text is "WTAAD".

Decryption

We apply decryption formulas by character, based on alphabetical order.

The formula of decryption is:

Dn (x) = (xi - n) mod 26

|  |  |  |
| --- | --- | --- |
| Ciphertext: W → 22 | Dn: (22 - 15) mod 26 | Plaintext: 07 → H |
| Ciphertext: T → 19 | Dn: (19 - 15) mod 26 | Plaintext: 04 → E |
| Ciphertext: A → 00 | Dn: (00 - 15) mod 26 | Plaintext: 11 → L |
| Ciphertext: A → 00 | Dn: (00 - 15) mod 26 | Plaintext: 11 → L |
| Ciphertext: D → 03 | Dn: (03 - 15) mod 26 | Plaintext: 14 → O |

The decrypted message is "HELLO".

Note: If any case (Dn) value becomes negative (-ve), in this case, we will add 26 in the negative value. Like, the third letter of the ciphertext.

Dn = (00 - 15) mod 26  
= -15

The value of dn is negative, so 26 will be added to it.

= -15 + 26  
= 11

Advantages of Caesar cipher

Its benefits are as follows: -

It is very easy to implement.

This method is the simplest method of cryptography.

Only one short key is used in its entire process.

If a system does not use complex coding techniques, it is the best method for it.

It requires only a few computing resources.

Disadvantages of Caesar cipher

Its disadvantages are as follows: -

It can be easily hacked. It means the message encrypted by this method can be easily decrypted.

It provides very little security.

By looking at the pattern of letters in it, the entire message can be decrypted.

# Greatest Common Factor (GCD) :

The greatest common divisor (also known as greatest common factor, highest common divisor or highest common factor) of a set of numbers is the largest positive integer number that devides all the numbers in the set without remainder. It is the biggest multiple of all numbers in the set.

The GCD is most often calculated for two numbers, when it is used to [reduce fractions to their lowest terms](http://www.alcula.com/calculators/math/simplify-fractions/). When the greatest common divisor of two numbers is 1, the two numbers are said to be coprime or relatively prime.

How is the greatest common divisor calculated?

This calculator uses Euclid's algorithm. To find out more about the Euclid's algorithm or the GCD, see this [Wikipedia article](http://en.wikipedia.org/wiki/Greatest_common_divisor).

The GCD may also be calculated using the [least common multiple](http://www.alcula.com/calculators/math/lcm/) using this formula:

$ GCD(a,b) = \frac{\left| a \cdot b \right|}{LCM(a,b)} $

In mathematics, the **greatest common factor** or **greatest common divisor** of two or more integers is the largest positive integer that divides each of the integers completely.

In this section, we will learn about **factors, common factors,** and **greatest common factor**. Before moving to the greatest common factor, first, we will understand **factor** and **common factor**.

**Factors:** Factors are whole numbers that multiplied together to get another number. A number may have more than two factors. For example, **5×3=15, 1×15=15** where 5, 3, 1 and 15 are the factors of 15. Similarly, the factors of 24 are: **1×24=24, 2×12=24, 3×8=24, 4×6=24**. Hence, the factors of 24 are **1, 2, 3, 4, 6, 8, 12, 24**.

2.1 Try Catch

**Common Factors:** The factor(s) that are common in two or more numbers is called the common factor(s). In other words, common factor(s) are numbers that you can multiply together to produce another number. The numbers should divide exactly into two or more numbers. It is necessary to have at least two numbers to find the common factor(s). For example, we have to find the factor of 12 and 16.

**Factors of 12:** 1, 2, 3, 4, 6, 12

**Factors of 16:** 1, 2, 4, 8, 16

We see that **1, 2,** and **4** is common in both. So, these are the common factors of the integer 12 and 16.

In the above examples, we have observed that 1 and the number itself appears in both the factors. So, we can conclude that **1** and the **number itself** are the two factors of ever number.

**Greatest Common Factor:** It is the highest number that completely divides two or more numbers. It is abbreviated for **GCF**. It is also known as the **Greatest Common Divisor** (GCD) and the **Highest Common Factor** (HCF). It is used to simplify the fractions.

How to Find Greatest Common Factor

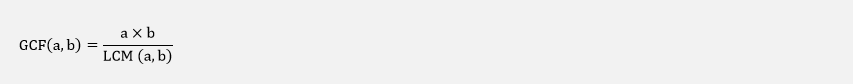
Follow the steps given below to find the greatest common factor.

Write all the factors of each number.

Select the common factors.

Select the greatest number, as GCF.

We can also use the following formula:



Note: Use the above formula only for two numbers.

Let's understand it through examples.

**Example 1: Find the GCF of 12 and 8.**

**Solution:**

Factors of 12: 1, 2, 3, 4, 6, 12

Factors of 8: 1, 2, 4, 8

Common Factors: 1, 2, 4

Greatest Common Factor: 4

**Hence, the GCF of 12 and 8 is 4.**

**Example 2: Find the GCF of 24 and 36.**

**Solution:**

Factors of 24: 1, 2, 3, 4, 6, 8, 12, 24

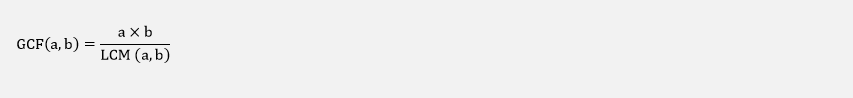
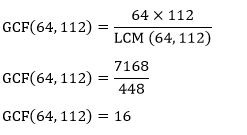
Factors of 36: 1, 2, 3, 4, 6, 9, 12, 18, 36

Common Factors: 1, 2, 3, 4, 6, 12

Greatest Common Factor: 12

**Hence, the GCF of 24 and 36 is 12.**

**Using GCF Formula**

**Hence, the GCF of 64 and 112 is 16.**

**Example 6: Find the GCF of 33 and 56.**

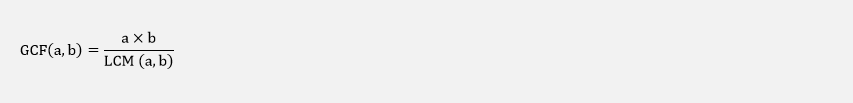
**Solution:**

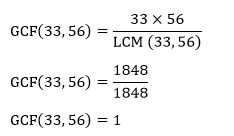
Factors of 33: 1, 3, 11, 33

Factors of 56: 1, 2, 4, 7, 8, 14, 28, 56

Common Factors: 1

Greatest Common Factor: 1

**Using GCF Formula**



**Hence, the GCF of 33 and 56 is 1.**

# What is the RSA algorithm (Rivest-Shamir-Adleman)?

The RSA algorithm (Rivest-Shamir-Adleman) is the basis of a cryptosystem -- a suite of cryptographic algorithms that are used for specific security services or purposes -- which enables public key encryption and is widely used to secure sensitive data, particularly when it is being sent over an insecure network such as the internet

**How does the RSA algorithm work?**

Alice generates her RSA keys by selecting two primes: p=11 and q=13. The modulus is n=p×q=143. The totient is n ϕ(n)=(p−1)x(q−1)=120. She chooses 7 for her RSA public key e and calculates her RSA private key using the Extended Euclidean algorithm, which gives her 103.

Bob wants to send Alice an encrypted message, M, so he obtains her RSA public key (n, e) which, in this example, is (143, 7). His [plaintext](https://www.techtarget.com/searchsecurity/definition/plaintext) message is just the number 9 and is encrypted into [ciphertext](https://www.techtarget.com/whatis/definition/ciphertext), C, as follows:

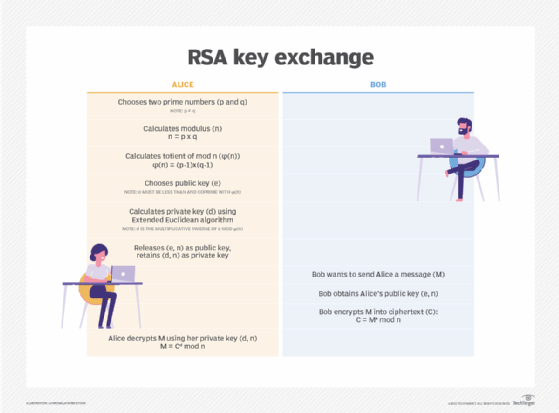
Me mod n = 97 mod 143 = 48 = C

When Alice receives Bob's message, she decrypts it by using her RSA private key (d, n) as follows:

Cd mod n = 48103 mod 143 = 9 = M

To use RSA keys to [digitally sign a message](https://www.techtarget.com/searchsecurity/answer/Which-private-keys-and-public-keys-can-create-a-digital-signature), Alice would need to create a [hash](https://searchsqlserver.techtarget.com/definition/hashing) -- a message digest of her message to Bob -- encrypt the hash value with her RSA private key, and add the key to the message. Bob can then verify that the message has been sent by Alice and has not been altered by decrypting the hash value with her public key. If this value matches the hash of the original message, then only Alice could have sent it -- authentication and non-repudiation -- and the message is exactly as she wrote it -- integrity.

Alice could, of course, encrypt her message with Bob's RSA public key -- confidentiality -- before sending it to Bob. A [digital certificate](https://www.techtarget.com/searchsecurity/definition/digital-certificate) contains information that identifies the certificate's owner and also contains the owner's public key. Certificates are signed by the [certificate authority](https://www.techtarget.com/searchsecurity/definition/certificate-authority) that issues them, and they can simplify the process of obtaining public keys and verifying the owner.



# Cryptography and its Types

[Cryptography](https://www.geeksforgeeks.org/cryptography-introduction-to-crypto-terminologies/) is technique of securing information and communications through use of codes so that only those person for whom the information is intended can understand it and process it. Thus preventing unauthorized access to information. The prefix “crypt” means “hidden” and suffix “graphy” means “writing”. In Cryptography the techniques which are use to protect information are obtained from mathematical concepts and a set of rule based calculations known as algorithms to convert messages in ways that make it hard to decode it. These algorithms are used for cryptographic key generation, digital signing, verification to protect data privacy, web browsing on internet and to protect confidential transactions such as credit card and debit card transactions.

# ****Features Of Cryptography are as follows:****

1. **Confidentiality:** Information can only be accessed by the person for whom it is intended and no other person except him can access it.
2. **Integrity:** Information cannot be modified in storage or transition between sender and intended receiver without any addition to information being detected.
3. **Non-repudiation:** The creator/sender of information cannot deny his intention to send information at later stage.
4. **Authentication:** The identities of sender and receiver are confirmed. As well as destination/origin of information is confirmed.

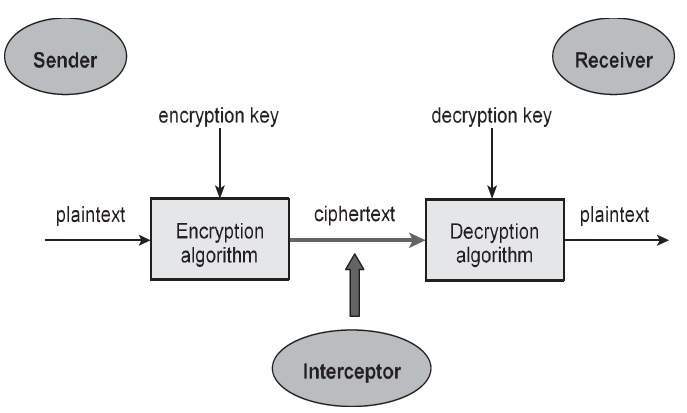
Types Of Cryptography: In general there are three types Of cryptography:

1. **Symmetric Key Cryptography:** It is an encryption system where the sender and receiver of message use a single common key to encrypt and decrypt messages. Symmetric Key Systems are faster and simpler but the problem is that sender and receiver have to somehow exchange key in a secure manner. The most popular symmetric key cryptography system are Data Encryption System(DES) and Advanced Encryption System(AES).
2. **Hash Functions:** There is no usage of any key in this algorithm. A hash value with fixed length is calculated as per the plain text which makes it impossible for contents of plain text to be recovered. Many operating systems use hash functions to encrypt passwords.
3. **Asymmetric Key Cryptography:** Under this system a pair of keys is used to encrypt and decrypt information. A receiver’s public key is used for encryption and a receiver’s private key is used for decryption. Public key and Private Key are different. Even if the public key is known by everyone the intended receiver can only decode it because he alone know his private key. The most popular asymmetric key cryptography algorithm is RSA algorithm.

# Cryptosystems

A cryptosystem is an implementation of cryptographic techniques and their accompanying infrastructure to provide information security services. A cryptosystem is also referred to as a **cipher system**.

Let us discuss a simple model of a cryptosystem that provides confidentiality to the information being transmitted. This basic model is depicted in the illustration below −



The illustration shows a sender who wants to transfer some sensitive data to a receiver in such a way that any party intercepting or eavesdropping on the communication channel cannot extract the data.

The objective of this simple cryptosystem is that at the end of the process, only the sender and the receiver will know the plaintext.

# Components of a Cryptosystem

The various components of a basic cryptosystem are as follows −

* **Plaintext.** It is the data to be protected during transmission.
* **Encryption Algorithm.** It is a mathematical process that produces a ciphertext for any given plaintext and encryption key. It is a cryptographic algorithm that takes plaintext and an encryption key as input and produces a ciphertext.
* **Ciphertext.** It is the scrambled version of the plaintext produced by the encryption algorithm using a specific the encryption key. The ciphertext is not guarded. It flows on public channel. It can be intercepted or compromised by anyone who has access to the communication channel.
* **Decryption Algorithm,** It is a mathematical process, that produces a unique plaintext for any given ciphertext and decryption key. It is a cryptographic algorithm that takes a ciphertext and a decryption key as input, and outputs a plaintext. The decryption algorithm essentially reverses the encryption algorithm and is thus closely related to it.
* **Encryption Key.** It is a value that is known to the sender. The sender inputs the encryption key into the encryption algorithm along with the plaintext in order to compute the ciphertext.
* **Decryption Key.** It is a value that is known to the receiver. The decryption key is related to the encryption key, but is not always identical to it. The receiver inputs the decryption key into the decryption algorithm along with the ciphertext in order to compute the plaintext.

For a given cryptosystem, a collection of all possible decryption keys is called a **key space**.

An **interceptor** (an attacker) is an unauthorized entity who attempts to determine the plaintext. He can see the ciphertext and may know the decryption algorithm. He, however, must never know the decryption key.

# Types of Cryptosystems :

Fundamentally, there are two types of cryptosystems based on the manner in which encryption-decryption is carried out in the system −

* Symmetric Key Encryption
* Asymmetric Key Encryption

The main difference between these cryptosystems is the relationship between the encryption and the decryption key. Logically, in any cryptosystem, both the keys are closely associated. It is practically impossible to decrypt the ciphertext with the key that is unrelated to the encryption key.

# what is the digital signature in cyber security

A digital signature is a way of verifying the authenticity and integrity of a message, document, or software using a mathematical technique based on public key cryptography. A digital signature allows the receiver to confirm that the message comes from the original sender and has not been tampered with in transit. A digital signature consists of three algorithms: key generation, signing, and verification. The sender uses their private key to encrypt a hash value of the message, which is called the signature. The receiver uses the sender’s public key to decrypt the signature and compare it with the hash value of the message. [If they match, the digital signature is valid and the message is authentic](https://www.bing.com/aclk?ld=e8O69vT4TGX7uz70xS_JSjCzVUCUx32je_7DLrSRQhlGRbsm9jG4a96z5clb5610IJknqeRtKKms9FFjP5oJwQSWCNBMLlEetD8fDR0vPDEqrXMW0p8z3A95QsXg-QWIdDuzc0SUSp09arqBpKrvMmgbeCgr6i806ULXJ79_v6iKil2wtK&u=&rlid=ed08d6a760a31cf5581bb257a5d20918)..

# what are the types of attacks

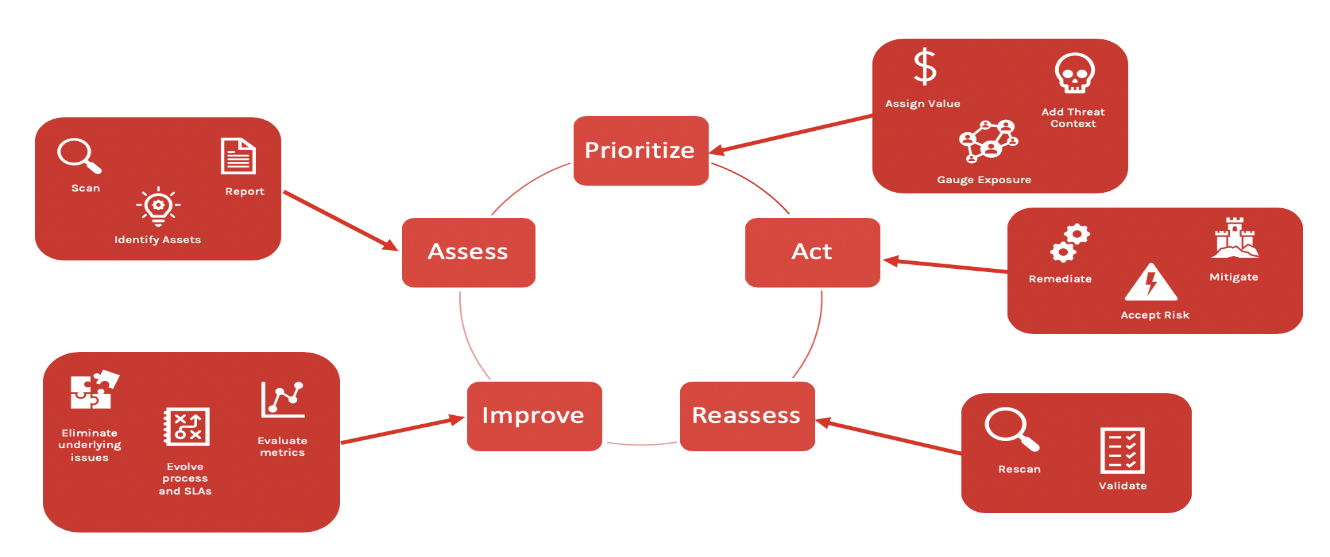
There are many types of cyberattacks that can target different aspects of a computer system, network, or application. Some of the most common types of cyberattacks are:

* **Malware**: Malware is any malicious software or code that is designed to harm or compromise a computer, network, or server. [Malware can include ransomware, trojans, spyware, viruses, worms, keyloggers, bots, and more](about:blank)[1](https://www.javatpoint.com/types-of-cyber-attacks)[2](https://www.crowdstrike.com/cybersecurity-101/cyberattacks/most-common-types-of-cyberattacks/).
* **Phishing**: Phishing is a type of social engineering attack that attempts to trick users into revealing sensitive information or clicking on malicious links or attachments. [Phishing emails often impersonate legitimate entities or individuals and use deceptive techniques to lure users into falling for the scam](about:blank)[1](https://www.javatpoint.com/types-of-cyber-attacks)[2](https://www.crowdstrike.com/cybersecurity-101/cyberattacks/most-common-types-of-cyberattacks/).
* **Man-in-the-middle (MITM)**: MITM is a type of attack that intercepts the communication between two parties and alters or steals the data being exchanged. [MITM attacks can be used to eavesdrop on confidential information, modify transactions, or redirect users to malicious websites](about:blank)[1](https://www.javatpoint.com/types-of-cyber-attacks)[2](https://www.crowdstrike.com/cybersecurity-101/cyberattacks/most-common-types-of-cyberattacks/).
* **Denial-of-service (DoS) or distributed denial-of-service (DDoS)**: DoS and DDoS are types of attacks that aim to disrupt the availability or performance of a system, network, or service by overwhelming it
* with illegitimate requests or traffic. [DoS and DDoS attacks can cause slowdowns, crashes, or outages for the targeted system](about:blank)[1](https://www.javatpoint.com/types-of-cyber-attacks)[2](https://www.crowdstrike.com/cybersecurity-101/cyberattacks/most-common-types-of-cyberattacks/).
* **SQL injection**: SQL injection is a type of attack that exploits a vulnerability in a web application that uses a SQL database. [SQL injection allows an attacker to execute malicious SQL commands on the database and access, modify, or delete data](about:blank)[1](https://www.javatpoint.com/types-of-cyber-attacks)[2](https://www.crowdstrike.com/cybersecurity-101/cyberattacks/most-common-types-of-cyberattacks/).
* **Zero-day exploit**: A zero-day exploit is a type of attack that takes advantage of a previously unknown vulnerability in a software or hardware system. [A zero-day exploit gives an attacker an opportunity to compromise the system before the vendor or developer can release a patch or fix](about:blank)[1](https://www.javatpoint.com/types-of-cyber-attacks)[2](https://www.crowdstrike.com/cybersecurity-101/cyberattacks/most-common-types-of-cyberattacks/).
* **Password attack**: A password attack is a type of attack that tries to guess or crack the passwords of users or systems. [Password attacks can use brute force methods, dictionary methods, or phishing methods to obtain passwords](about:blank).

# what is the vulnerability life cycle management system.

The vulnerability life cycle management system is a cybersecurity process that aims to identify, assess, prioritize, and act on the vulnerabilities that exist in an IT system, network, or application. Vulnerabilities are the weaknesses or flaws that can be exploited by cyberattackers to compromise the security or functionality of a system. [The vulnerability life cycle management system helps to prevent or mitigate the impact of cyberattacks by finding and fixing the vulnerabilities before they can be exploited](about:blank)[1](https://www.crowdstrike.com/cybersecurity-101/vulnerability-management/vulnerability-management-lifecycle/)[2](https://heimdalsecurity.com/blog/vulnerability-management-lifecycle/).

**The vulnerability life cycle management system consists of five main stages:**



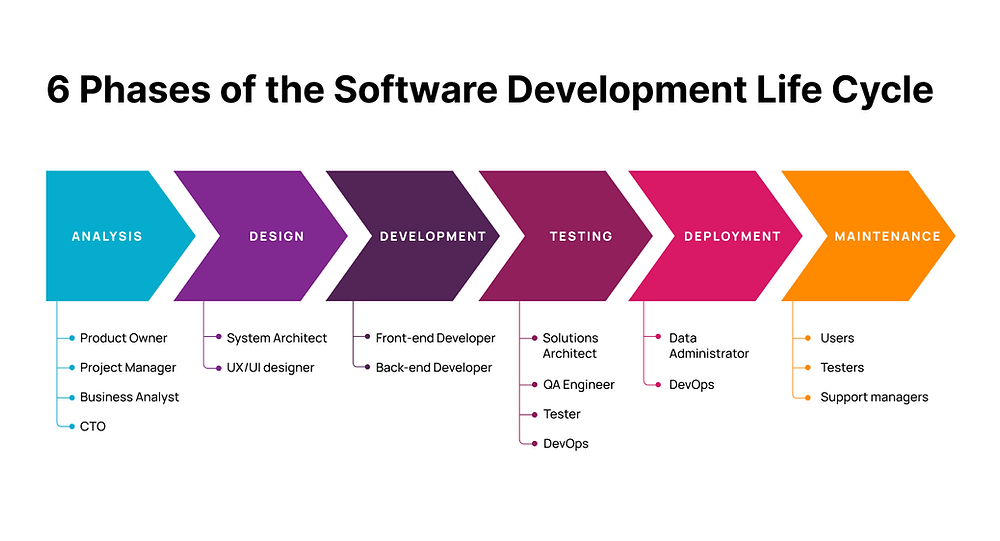
* **Assessment**: This stage involves scanning and testing the system for vulnerabilities using various tools and methods. [The assessment stage produces a report that lists the vulnerabilities and their severity levels](about:blank)[1](https://www.crowdstrike.com/cybersecurity-101/vulnerability-management/vulnerability-management-lifecycle/)[2](https://heimdalsecurity.com/blog/vulnerability-management-lifecycle/).
* **Prioritization**: This stage involves ranking the vulnerabilities based on their risk and impact. [The prioritization stage helps to allocate the resources and time for the remediation process](about:blank)[1](https://www.crowdstrike.com/cybersecurity-101/vulnerability-management/vulnerability-management-lifecycle/)[2](https://heimdalsecurity.com/blog/vulnerability-management-lifecycle/).
* **Act**: This stage involves applying the appropriate solutions to fix or mitigate the vulnerabilities. [The act stage can involve patching, updating, configuring, or replacing the vulnerable components](about:blank)[1](https://www.crowdstrike.com/cybersecurity-101/vulnerability-management/vulnerability-management-lifecycle/)[2](https://heimdalsecurity.com/blog/vulnerability-management-lifecycle/).
* **Reassessment**: This stage involves verifying that the solutions have been implemented correctly and effectively. [The reassessment stage can involve rescanning, retesting, or auditing the system for any remaining or new vulnerabilities](about:blank)[1](https://www.crowdstrike.com/cybersecurity-101/vulnerability-management/vulnerability-management-lifecycle/)[2](https://heimdalsecurity.com/blog/vulnerability-management-lifecycle/).
* **Improvement**: This stage involves analyzing the results and feedback from the previous stages and improving the vulnerability management process. [The improvement stage can involve updating the policies, procedures, tools, or best practices for vulnerability management](about:blank)[1](https://www.crowdstrike.com/cybersecurity-101/vulnerability-management/vulnerability-management-lifecycle/)[2](https://heimdalsecurity.com/blog/vulnerability-management-lifecycle/).

# what is SDLC

SDLC stands for Software Development Life Cycle, which is a process that defines the stages and tasks involved in creating, testing, and deploying software applications. [SDLC aims to ensure that the software meets the quality, cost, and time requirements of the clients and stakeholders1](https://www.javatpoint.com/software-engineering-software-development-life-cycle)[2](https://stackify.com/what-is-sdlc/).

There are different models or methodologies of SDLC, such as waterfall, agile, spiral, and iterative. [Each model has its own advantages and disadvantages, depending on the nature and scope of the software project3](https://www.tutorialspoint.com/sdlc/sdlc_overview.htm)[4](https://en.wikipedia.org/wiki/Systems_development_life_cycle).

The common stages of SDLC are:



* **Requirement analysis**: This stage involves gathering and analyzing the needs and expectations of the
* clients and users. [It also involves defining the scope, objectives, and feasibility of the software project1](https://www.javatpoint.com/software-engineering-software-development-life-cycle)[2](https://stackify.com/what-is-sdlc/).
* **Planning**: This stage involves estimating the cost, time, and resources required for the software project. [It also involves identifying the risks and challenges and developing strategies to mitigate them](about:blank)[1](https://www.javatpoint.com/software-engineering-software-development-life-cycle)[2](https://stackify.com/what-is-sdlc/).
* **Design**: This stage involves creating the architecture and design of the software system. [It also involves selecting the appropriate tools, technologies, and frameworks for the software development](about:blank)[1](https://www.javatpoint.com/software-engineering-software-development-life-cycle)[2](https://stackify.com/what-is-sdlc/).
* **Development**: This stage involves writing and coding the software according to the design specifications. [It also involves following the coding standards and best practices for software development](about:blank)[1](https://www.javatpoint.com/software-engineering-software-development-life-cycle)[2](https://stackify.com/what-is-sdlc/).
* **Testing**: This stage involves verifying and validating the functionality, performance, and quality of the software. [It also involves finding and fixing any errors or bugs in the software](about:blank)[1](https://www.javatpoint.com/software-engineering-software-development-life-cycle)[2](https://stackify.com/what-is-sdlc/).
* **Deployment**: This stage involves releasing and delivering the software to the clients or users. [It also involves installing and configuring the software on the target platforms or environments1](https://www.javatpoint.com/software-engineering-software-development-life-cycle)[2](https://stackify.com/what-is-sdlc/).
* **Maintenance**: This stage involves providing ongoing support and updates for the software. [It also involves resolving any issues or problems that may arise in the software1](https://www.javatpoint.com/software-engineering-software-development-life-cycle)[2](https://stackify.com/what-is-sdlc/).