IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.67
Volume 5, Issue 4, February 2025

Encrypted Insights: A Deep Dive into Cryptography

Sagar Laxman Bhor, Omkar Sampat Naykodi, Dipak Vilas Shelkande
Jay Santosh Maskare, Siddhi Ashok Kokane, Shubham Rohidas Mindhe
Shubham Shalivan Borhade
Shankarrao Butte Patil B.Sc. IT College, Junnar, Maharashtra, India

Abstract: *This paper explores cryptography, its importance, types of encryption techniques, and applications in various industries. It highlights cryptographic algorithms such as symmetric and asymmetric encryption, digital signatures, and key exchange mechanisms. Additionally, it discusses contemporary challenges, including brute force attacks, cryptanalysis, and emerging threats posed by quantum computing. The study also delves into future advancements like post-quantum cryptography and homomorphic encryption, which aim to strengthen data security in evolving digital landscapes.*

Keywords: Cryptography, Encryption, Cybersecurity, Data Security, Digital Signatures, Public Key Cryptography, Quantum Computing, Hashing, Blockchain Security

I. INTRODUCTION

Cryptography is the practice of securing information through encryption, ensuring only the intended recipient can access it. Historically used for encoding messages, modern cryptography is fundamental in areas such as banking, online security, and data protection. It employs advanced algorithms like the Advanced Encryption Standard (AES) to maintain confidentiality and safeguard sensitive data from cyber threats. Cryptography integrates computer science, mathematics, and engineering to develop secure communication systems.

[1]. Introduction to Cryptography:

Cryptography is the process of hiding or coding information so that only the person a message was intended for can read it. The art of cryptography has been used to code messages for thousands of years and continues to be used in bank cards, computer passwords, and ecommerce.

Modern cryptography techniques include algorithms and ciphers that enable the encryption and decryption of information, such as 128-bit and 256-bit encryption keys. Modern ciphers, such as the Advanced Encryption Standard (AES), are considered virtually unbreakable.

A common cryptography definition is the practice of coding information to ensure only the person that a message was written for can read and process the information. This cybersecurity practice, also known as cryptology, combines various disciplines like computer science, engineering, and mathematics to create complex codes that hide the true meaning of a message.

The Importance of Cryptography

Cryptography remains important to protecting data and users, ensuring confidentiality, and preventing cyber criminals from intercepting sensitive corporate information. Common uses and examples of cryptography include the following:

1. Privacy and confidentiality : Individuals and organizations use cryptography on a daily basis to protect their privacy and keep their conversations and data confidential. Cryptography ensures confidentiality by encrypting sent messages using an algorithm with a key only known to the sender and recipient. A common example of this is the messaging tool WhatsApp, which encrypts conversations between people to ensure they cannot be hacked or intercepted.

Cryptography also secures browsing, such as with virtual private networks (VPNs), which use encrypted tunnels, asymmetric encryption, and public and private shared keys.

2. Integrity : Similar to how cryptography can confirm the authenticity of a message, it can also prove the integrity of the information being sent and received. Cryptography ensures information is not altered while in storage or during

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.67
Volume 5, Issue 4, February 2025

transit between the sender and the intended recipient. For example, digital signatures can detect forgery or tampering in software distribution and financial transactions.

3. Nonrepudiation : Cryptography confirms accountability and responsibility from the sender of a message, which means they cannot later deny their intentions when they created or transmitted information. Digital signatures are a good example of this, as they ensure a sender cannot claim a message, contract, or document they created to be fraudulent. Furthermore, in email nonrepudiation, email tracking makes sure the sender cannot deny sending a message and a recipient cannot deny receiving it.

4. Key exchange: Key exchange is the method used to share cryptographic keys between a sender and their recipient.

Types of Cryptographic Algorithms

There are many types of cryptographic algorithms available. They vary in complexity and security, depending on the type of communication and the sensitivity of the information being shared.

1. Secret Key Cryptography : Secret key cryptography, also known as symmetric encryption, uses a single key to encrypt and decrypt a message. The sender encrypts the plaintext message using the key and sends it to the recipient who then uses the same key to decrypt it and unlock the original plaintext message.

2. Stream ciphers : Stream ciphers work on a single bit or byte at any time and constantly change the key using feedback mechanisms. A self-synchronizing stream cipher ensures the decryption process stays in sync with the encryption process by recognizing where it sits in the bit keystream. A synchronous stream cipher generates the keystream independently of the message stream and generates the same keystream function at both the sender and the receiver.

3. Block ciphers : Block ciphers encrypt one block of fixed-size data at a time. It will always encrypt a plaintext data block to the same ciphertext when the same key is used. A good example of this is the Feistel cipher, which uses elements of key expansion, permutation, and substitution to create vast confusion and diffusion in the cipher. The stages of encryption and decryption are similar if not identical, which means reversing the key reduces the code size and circuitry required for implementing the cipher in a piece of software or hardware.

4. Public Key Cryptography : Public key cryptography (PKC), or asymmetric cryptography, uses mathematical functions to create codes that are exceptionally difficult to crack. It enables people to communicate securely over a nonsecure communications channel without the need for a secret key. For example, proxy reencryption enables a proxy entity to reencrypt data from one public key to another without requiring access to the plaintext or private keys. A common PKC type is multiplication vs. factorization, which takes two large prime numbers and multiplies them to create a huge resulting number that makes deciphering difficult. Another form of PKC is exponentiation vs. logarithms such as 256-bit encryption, which increases protection to the point that even a computer capable of searching trillions of combinations per second cannot crack it.

5. RSA : RSA was the first and remains the most common PKC implementation. The algorithm is named after its MIT mathematician developers, Ronald Rivest, Adi Shamir, and Leonard Adleman, and is used in data encryption, digital signatures, and key exchanges. It uses a large number that is the result of factoring two selected prime numbers. It is impossible for an attacker to work out the prime factors, which makes RSA especially secure.

6. Eliptic Curve Cryptography (ECC) : ECC is a PKC algorithm based on the use of elliptic curves in cryptography. It is designed for devices with limited computing power or memory to encrypt internet traffic. A common use of ECC is in embedded computers, smartphones, and cryptocurrency networks like bitcoin, which consumes around 10% of the storage space and bandwidth that RSA requires.

7. Digital Signature Algorithm (DSA) : DSA is a standard that enables digital signatures to be used in message authentication. It was introduced by the National Institute of Standards and Technology (NIST) in 1991 to ensure a better method for creating digital signatures.

Diffie-Hellman and Key Exchange Algorithm (KEA)

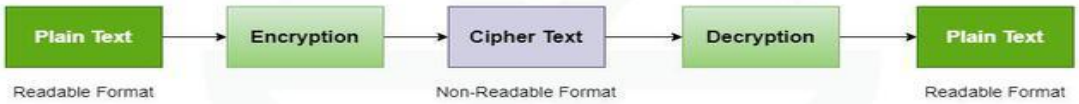
The Diffie-Hellman algorithm was devised in 1976 by Stanford University professor Martin Hellman and his graduate student Whitfield Diffie, who are considered to be responsible for introducing PKC as a concept. It is used for secret key exchanges and requires two people to agree on a large prime number.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.67
Volume 5, Issue 4, February 2025

KEA is a variation of the Diffie-Hellman algorithm and was proposed as a method for key exchange in the NIST/National Security Agency's (NSA) Capstone project, which developed cryptography standards for public and government use.

(1.1). Cryptography

Cryptography is a technique of securing information and communications through the use of codes so that only those persons for whom the information is intended can understand and process it. Thus, preventing unauthorized access to information. The prefix "crypt" means "hidden" and the suffix "graphy" means "writing". In Cryptography, the techniques that are used 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 them. These algorithms are used for cryptographic key generation, digital signing, and verification to protect data privacy, web browsing on the internet and to protect confidential transactions such as credit card and debit card transactions.



Block cipher schemes: DES and AES

DES and AES are both symmetric block cipher schemes, meaning they use only one key for data encryption and decryption. DES was originally designed by IBM in 1975 and consisted of 64-bit blocks and a 56-bit key. Its key size is now considered too short for today's cybersecurity and information security which is why this cipher is no longer considered secure.

AES uses a 128-bit block size and a 128-, 192- or 256-bit key size. The longer keys make AES much more secure than DES and its variants. Designed to replace DES, the AES standard was published by the National Institute of Standards and Technology in 2001. Today, AES is widely used for data security applications, including databases, web browsers, wireless systems and virtual private network

(1.2). Differences between AES and DES

Benchmark	AES	DES
Cipher Type	Substitution-Permutation Network	Feistel Network
Block Size	128 bits	64 bits
Key Size	128, 192, or 256 bits	56 bits
Number of Rounds	10, 12, or 14 rounds (depending on key size)	16 rounds
S-Box (Substitution Box)	Highly nonlinear substitution boxes	Relatively simple substitution boxes
Security	Considered highly secure	Insecure due to small key size and cryptanalytic attacks

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.67
Volume 5, Issue 4, February 2025

Benchmark	AES	DES
Cryptanalytic Resistance	Resistant to known attacks	Vulnerable to differential, linear, side-channel attacks, and brute-force attacks
Adoption and Usage	Widely adopted and used in various applications	Deprecated and replaced by AES and 3DES
Standardization	Adopted as a standard by NIST in 2001	Adopted as a standard in the 1970s, later withdrawn
Performance	Efficient and suitable for software and hardware implementations	Slower compared to modern ciphers

(1.3). RSA Cryptosystem

This cryptosystem is one the initial system. It remains most employed cryptosystem even today. The system was invented by three scholars **Ron Rivest, Adi Shamir,** and **Len Adleman** and hence, it is termed as RSA cryptosystem.

We will see two aspects of the RSA cryptosystem, firstly generation of key pair and secondly encryption-decryption algorithms.

Generation of RSA Key Pair

Each person or a party who desires to participate in communication using encryption needs to generate a pair of keys, namely public key and private key. The process followed in the generation of keys is described below −

Generate the RSA modulus (n)

Select two large primes, p and q.

Calculate n=p*q. For strong unbreakable encryption, let n be a large number, typically a minimum of 512 bits.

Find Derived Number (e)

Number **e** must be greater than 1 and less than (p − 1)(q − 1).

There must be no common factor for e and (p − 1)(q − 1) except for 1. In other words two numbers e and (p – 1)(q – 1) are coprime.

Form the public key

The pair of numbers (n, e) form the RSA public key and is made public.

Interestingly, though n is part of the public key, difficulty in factorizing a large prime number ensures that attacker cannot find in finite time the two primes (p & q) used to obtain n. This is strength of RSA.

Generate the private key

Private Key d is calculated from p, q, and e. For given n and e, there is unique number d.

Number d is the inverse of e modulo (p - 1)(q – 1). This means that d is the number less than (p - 1)(q - 1) such that when multiplied by e, it is equal to 1 modulo (p - 1)(q - 1).

This relationship is written mathematically as follows −

$ed = 1 \mod (p − 1)(q − 1)$

(1.4). Elliptical curve cryptography (ECC)

Elliptical curve cryptography (ECC) is a public key encryption technique based on elliptic curve theory that can be used to create faster, smaller and more efficient cryptographic keys.

ECC is an alternative to the Rivest-Shamir-Adleman (RSA) cryptographic algorithm and is most often used for digital signatures in cryptocurrencies, such as Bitcoin and Ethereum, as well as one-way encryption of emails, data and software.

Public key cryptography systems, like ECC, use a mathematical process to merge two distinct keys and then use the output to encrypt and decrypt data. One is a public key that is known to anyone, and the other is a private key that is only known by the sender and receiver of the data.

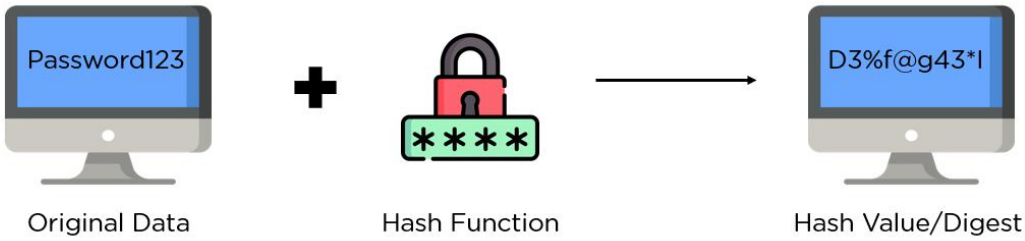
IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.67
Volume 5, Issue 4, February 2025

ECC formulated using the following equation:

$$y^2=x^3+ax+b$$

(1.5). What is Hashing?

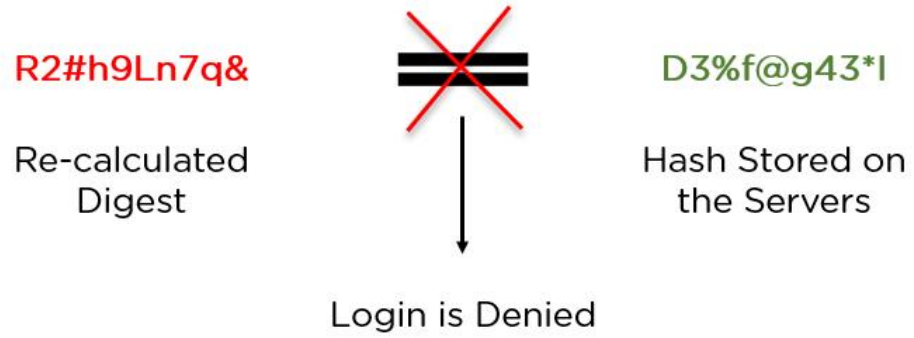
Hashing is the process of scrambling raw information to the extent that it cannot reproduce it back to its original form. It takes a piece of information and passes it through a function that performs mathematical operations on the plaintext. This function is called the hash function, and the output is called the hash value/digest**.**



As seen from the above image, the hash function is responsible for converting the plaintext to its respective hash digest. They are designed to be irreversible, which means your digest should not provide you with the original plaintext by any means necessary. Hash functions also provide the same output value if the input remains unchanged, irrespective of the number of iterations.

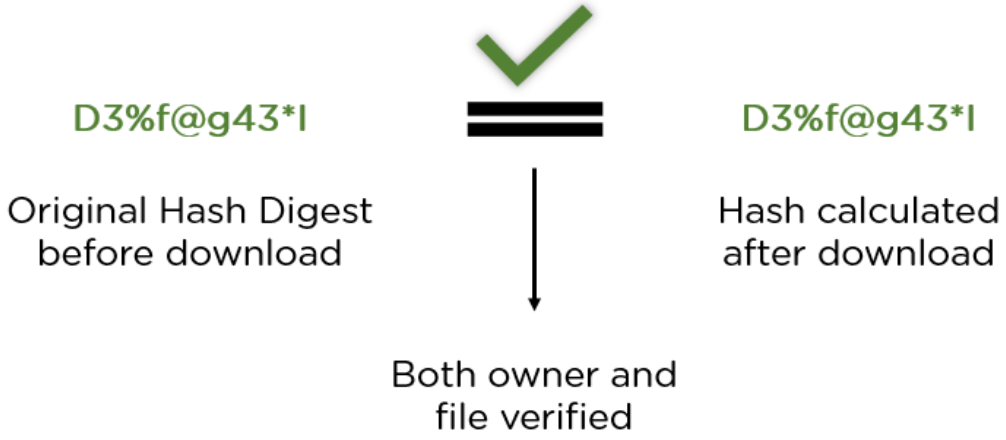
There are two primary applications of hashing:

Password Hashes: In most website servers, it converts user passwords into a hash value before being stored on the server. It compares the hash value re-calculated during login to the one stored in the database for validation.



IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.67
Volume 5, Issue 4, February 2025

Integrity Verification: When it uploads a file to a website, it also shared its hash as a bundle. When a user downloads it, it can recalculate the hash and compare it to establish data integrity.



Now that you understand the working of hash functions, look at the key topic in hand - SHA 256 algorithm.

What is the SHA-256 Algorithm?

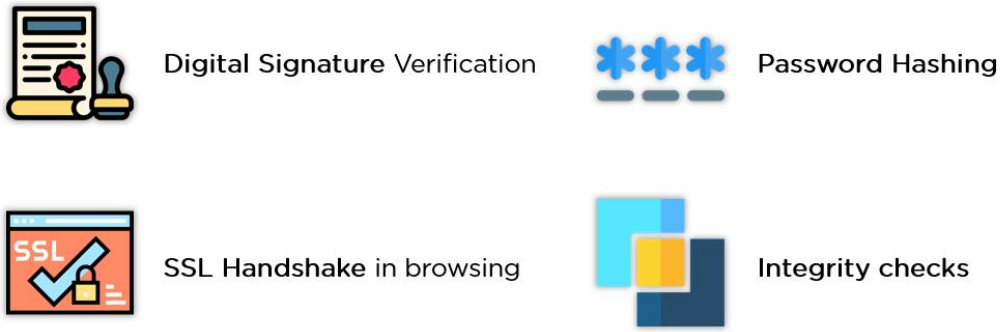
SHA 256 is a part of the SHA 2 family of algorithms, where SHA stands for Secure Hash Algorithm. Published in 2001, it was a joint effort between the NSA and NIST to introduce a successor to the SHA 1 family, which was slowly losing strength against brute forces attck

The significance of the 256 in the name stands for the final hash digest value, i.e. irrespective of the size of plaintext/cleartext, the hash value will always be 256 bits.

The other algorithms in the SHA family are more or less similar to SHA 256. Now, look into knowing a little more about their guidelines.

Embark on a transformative journey through our Cyber security Bootcamp, where you'll delve deep into the intricacies of cutting-edge technologies like the SHA-256 algorithm. Uncover the cryptographic principles that make this algorithm the cornerstone of blockchain security, all while honing your skills in defending against cyber threats.

Applications of SHA algorithm



IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.67
Volume 5, Issue 4, February 2025

As seen in the image above, the SHA algorithm is being used in a lot of places, some of which are as follows:
Digital Signature Verification: Digital signatures followasymmetric encryption methodology to verify the authenticity of a document/file. Hash algorithms like SHA 256 go a long way in ensuring the verification of the signature.
Password Hashing: As discussed above, websites store user passwords in a hashed format for two benefits. It helps foster a sense of privacy, and it lessens the load on the central database since all the digests are of similar size.
SSL Handshake: The SSL handshake is a crucial segment of the web browsing sessions, and it's done using SHA functions. It consists of your web browsers and the web servers agreeing on encryption keys and hashing authentication to prepare a secure connection.
Integrity Checks: As discussed above, verifying file integrity has been using variants like SHA 256 algorithm and the MD5 algorithm. It helps maintain the full value functionality of files and makes sure they were not altered in transit.

(1.6). What are Digital Signatures?

Digital signatures use asymmetric key cryptography. Asymmetric key cryptography also known as public key cryptography uses public and private keys to encrypt and decrypt data.
The public key can be shared with anyone.
The private key is the secret key that is kept a secret.
In short, it can be summarized as a digital signature a code that is attached to the message sent on the network. This code acts as proof that the message hasn't been tampered with along its way from sender to receiver.
A digital signature is intended to solve the problem of tampering and impersonation and tampering thus it gives a recipient reason to believe:
The message is sent by the claimed sender i.e. Authentication.
The sender cannot deny having sent the message i.e. Non-repudiation.
The message was not altered in the transit i.e. Integrity.

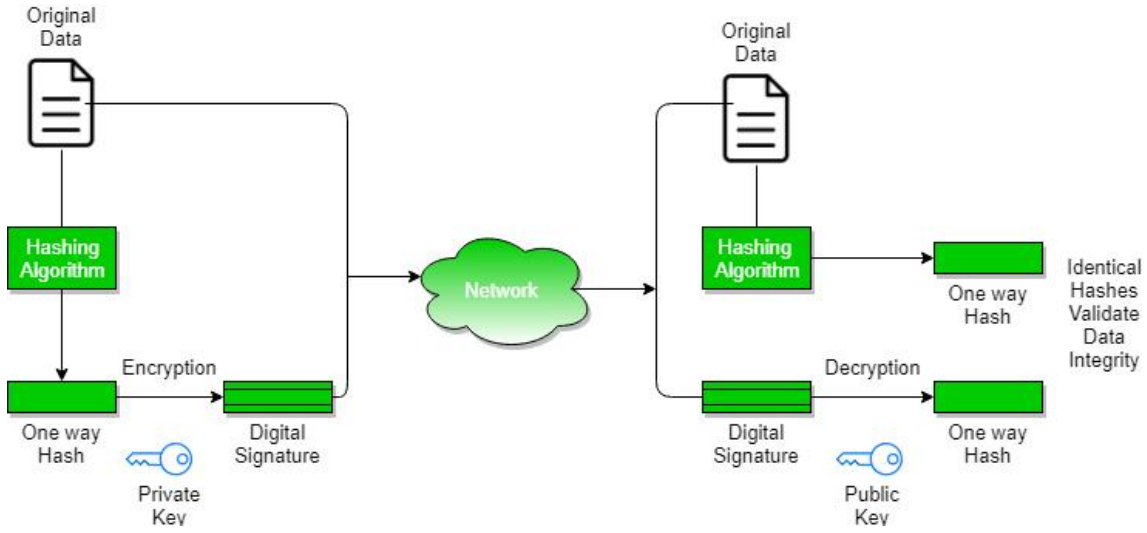
Why are Digital Signatures Important?

Digital signatures are important to achieve three results: Data integrity, authenticity, and non-repudiation.
1. Data Integrity: It is preserved by using the hash function in signing and verifying algorithms. Any change in the message will produce a completely different signature. This way Bob can verify that the message sent by Alice was not modified along its way.
2. Authenticity: The message is verified using the public key of the sender. When Alice sends a message to Bob. Bob uses the public key of Alice for verification and Alice's public key cannot create the same signature as Kev's private key.

How do Digital Signatures Work?

Let's have a look at the series of steps involved in working of digital signatures:
1. Signing the message with the private key: Digital signature is created using signing software that creates a one-way hash function of the data to be signed. The private key of the sender is used to encrypt the hash value generated. The encrypted hash value along with the hash algorithm constitutes the digital signature. The sender will now send the message along with the encrypted hash value to the receiver. The receiver can only decrypt the hash value using the sender's public key.
2. Verifying the message with the public key: At the receiver end, there are two steps, to generate the hash of the message and decryption of the signature. By using the sender's public key, the signature can be decrypted. if the decrypted hash matches the second computed hash value then it proves that the message hasn't been changed since it was signed. If the two hash values don't match then it means that the message has been tampered with along its way.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.67
Volume 5, Issue 4, February 2025



Applications of Digital Signatures

Digital signatures can be used in various fields like Finance, Healthcare, etc. Below are some of the applications of digital signatures:

1. Healthcare: Digital signatures are used in healthcare to improve the efficiency of administrative and treatment processes to strengthen data security. For example, for prescribing medicines and admissions to hospitals. They can be used to prevent fraudulent prescriptions and medical records.

2. Legal: Digital signatures can be used to reduce the time to close contracts that require multiple parties to validate and sign them. Due to the immutable nature of the blockchain, the contract validity can be trusted thus allowing parties to sign the contract at their convenience.

3. Government: Digital signatures are used by the government worldwide for a variety of reasons like processing tax returns, managing contracts, verifying B2G transactions, etc.

4. Financial services: Digital signatures can be used in expense reports, audits, loan agreements, etc.

5. Manufacturing: Digital signatures are used in the manufacturing industry to speed up processes like product design, quality assurance, and marketing sales. The use of digital signatures in Manufacturing is governed by organizations like ISO, NIST, and DMC.

6. Cryptocurrencies: Digital signatures are used in cryptocurrencies to authenticate the blockchain, and manage transaction data associated with the cryptocurrency.

Digital Signature Algorithms

1. RSA-based signature schemes: RSA is an asymmetric cryptographic algorithm.It can be used for performing a digital signature over a message. RSA Signature is quite reliable, strong, and secure.

2. Rabin signature algorithm: Rabin signature algorithm was one of the first digital signature schemes that were proposed. Hashing was introduced as an essential step in the signing process. It has relatively less use or standardization outside IEEE P1363.

3. ECDSA: Elliptic Curve Digital Signature Algorithm (ECDSA) is bitcoin's current digital signature scheme. This scheme uses shorter keys and has few computational requirements than the RSA system. This scheme uses elliptic curves instead of finite fields and relies on the discrete log problem instead of the difficulty of factoring primes for security.

4. ElGamal signature scheme: ElGamal digital signature scheme is based on the algebraic properties of modular exponentiation together with the difficulty of computing discrete logarithms. This is rarely used in practice. Its variant developed at NSA and also known as Digital Signature Algorithm is much more widely used.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.67
Volume 5, Issue 4, February 2025

(1.7). Differences between Active Attack and Passive Attack

No	Active Attack	Passive Attack
1	Attacker needs to have control media or network.	Attacker observe the communication in media or network.
2	It can be easily detected.	It cannot be easily detected.
3	It affects the system.	It does not affect the system.
4	It involves modification in data.	It involves in monitoring in data.
5	It does not check for loopholes or vulnerabilities.	It scans the ports and network in search for loopholes and vulnerabilities.
6	It is difficult to prevent network from active attack.	Passive attack can be prevented.
7	Types of active attack: Masquerade, replay, denial of service, modification of message.	Types of passive attack: release of message content, Traffic analysis.

(1.8). Brute Force Attack Definition

A brute force attack is a hacking method that uses trial and error to crack passwords login credentials and encryption keys. It is a simple yet reliable tactic for gaining unauthorized access to individual accounts and organizations' systems and networks. The hacker tries multiple usernames and passwords, often using a computer to test a wide range of combinations, until they find the correct login information.

The name "brute force" comes from attackers using excessively forceful attempts to gain access to user accounts. Despite being an old cyberattack method, brute force attacks are tried and tested and remain a popular tactic with hackers.

Types of Brute Force Attacks

There are various types of brute force attack methods that allow attackers to gain unauthorized access and steal user data.

1. Simple brute force attacks

A simple brute force attack occurs when a hacker attempts to guess a user's login credentials manually without using any software. This is typically through standard password combinations or personal identification number (PIN) codes. These attacks are simple because many people still use weak passwords, such as "password123" or "1234," or practice poor password etiquette, such as using the same password for multiple websites. Passwords can also be guessed by hackers that do minimal reconnaissance work to crack an individual's potential password, such as the name of their favorite sports team.

2. Dictionary attacks : A dictionary attack is a basic form of brute force hacking in which the attacker selects a target, then tests possible passwords against that individual's username. The attack method itself is not technically considered a brute force attack, but it can play an important role in a bad actor's password-cracking process. The name "dictionary attack" comes from hackers running through dictionaries and amending words with special characters and numbers. This type of attack is typically time-consuming and has a low chance of success compared to newer, more effective attack methods.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.67
Volume 5, Issue 4, February 2025

3. Hybrid brute force attacks : A hybrid brute force attack is when a hacker combines a dictionary attack method with a simple brute force attack. It begins with the hacker knowing a username, then carrying out a dictionary attack and simple brute force methods to discover an account login combination.
The attacker starts with a list of potential words, then experiments with character, letter, and number combinations to find the correct password. This approach allows hackers to discover passwords that combine common or popular words with numbers, years, or random characters, such as "SanDiego123" or "Rover2020."
4. Reverse brute force attacks : A reverse brute force attack sees an attacker begin the process with a known password, which is typically discovered through a network breach. They use that password to search for a matching login credential using lists of millions of usernames. Attackers may also use a commonly used weak password, such as "Password123," to search through a database of usernames for a match.
5. Credential stuffing : Credential stuffing preys on users' weak password etiquettes. Attackers collect username and password combinations they have stolen, which they then test on other websites to see if they can gain access to additional user accounts. This approach is successful if people use the same username and password combination or reuse passwords for various accounts and social media profiles.

Stronger password best practices include:
1. Create strong, multicharacter passwords: A basic rule of thumb is that passwords should be more than 10 characters in length and include capital and lowercase letters, symbols, andnumerals. This vastly increases the difficulty and time it takes to crack a password from a few hours to several years, unless a hacker has a supercomputer at hand.
2. Use elaborate passphrases: While using more characters is good password practice, some websites may have restrictions on the length of a password. As such, use complex passphrases to prevent attackers from succeeding with simple dictionary attacks. Passphrases are multiple words or segments with special characters that make them more difficult to guess.
3. Create password-building rules: Another good password tactic is to truncate words so they appear nonsensical to other people reading them. This can be done by removing vowels or only using the first two letters of words then building a phrase that makes sense out of a string of shortened words. For example, shortening the word "hope" to "hp" or "blue" to "bl."
4. Avoid common passwords: Frequently used passwords, such as a name, sports team, or simply "password," are extremely risky. Hackers know common words or phrases that people use in their passwords and deploy tactics based around these common words to hack into people's accounts.

(1.9). Limitations of Cryptography

Key Management Effective key management is a critical challenge in cryptography. Cryptographic keys are essential for encryption, decryption, and digital signatures.
Algorithm Vulnerabilities Cryptography relies on the strength of its underlying algorithms.
Quantum Computing Threat The emergence of quantum computers poses a potential threat to many traditional cryptographic methods.
Human Factor The security of cryptographic systems often relies on human behavior, which introduces the risk of human error.
Side-Channel Attacks Side-channel attacks exploit unintended information leakage during the execution of cryptographic algorithms.
Legal and Regulatory Challenges Cryptography is subject to legal and regulatory constraints in various countries.

(1.10). Applications of Cryptography

Computer passwords: Cryptography is widely utilized in computer security, particularly when creating and maintaining password.
Digital Currencies: To protect transactions and prevent fraud, digital currencies like Bitcoin also use cryptography.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.67
Volume 5, Issue 4, February 2025

Secure web browsing: Public key cryptography is used by the Secure Sockets Layer and Transport Layer Security (TLS) protocols to encrypt data sent between the web server and the client, establishing a secure channel for communication.

Electronic Signatures: Electronic signatures serve as the digital equivalent of a handwritten signature and are used to sign documents.

Authentication: Cryptography is used for authentication in many different situations, such as when accessing a bank account, logging into a computer, or using a secure network.

Cryptocurrencies: Cryptography is heavily used by cryptocurrencies like Bitcoin and Ethereum to protect transactions, thwart fraud, and maintain the network's integrity.

End-to-end Internet Encryption: End-to-end encryption is used to protect two-way communications like video conversations, instant messages, and email.

[2] Future Trends in Cryptography

(2.1)What is Post-Quantum Cryptography?

Post-Quantum Cryptography (PQC) is a branch of cryptography that aims to develop algorithms and protocols capable of withstanding the computational power of quantum computers. Quantum computers utilize the principles of quantum mechanics to perform certain types of calculations at unprecedented speeds that are unattainable by classical computers. While this technological leap promises significant advancements in various fields, it poses a critical threat to current cryptographic systems, particularly those relying on the hardness of certain mathematical problems such as integer factorization and discrete logarithms.

How Does Post-Quantum Cryptography Work?

Post-Quantum Cryptography involves designing cryptographic algorithms that remain secure even when subjected to the very specific capabilities of quantum computers. Two quantum algorithms that will be able to run on quantum computers, Grover's algorithm and Shor's algorithm, highlight the vulnerabilities of current cryptographic systems:

Grover's Algorithm: This algorithm can search an unsorted database quadratically faster than classical algorithms. While it doesn't completely break symmetric key cryptography, it significantly reduces the security of algorithms like AES (Advanced Encryption Standard) and SHA-2 (Secure Hash Algorithm 2), necessitating longer keys to maintain security.

Shor's Algorithm: This algorithm can factorize large integers and solve discrete logarithm problems exponentially faster than classical algorithms. This poses a direct threat to asymmetric cryptographic systems like RSA, ECC, and DSA, rendering them ineffective once a sufficiently powerful quantum computer is available.

(2.2) What is homomorphic encryption?

Fully homomorphic encryption (FHE) is an innovative technology that can help you achieve zero trust by unlocking the value of data on untrusted domains without needing to decrypt it.

Today's business data is stored across hybrid multicloud environments, exposing it to various security and privacy risks. While encryption provides protection, the sensitive data typically must first be decrypted to access it for computing and business-critical operations.

This opens the door to potential compromise of privacy and confidentiality controls. Until now, those vulnerabilities have been the cost of doing business in the cloud and with third parties.

With fully homomorphic encryption, you can better enforce zero trust because the data is always encrypted and can be shared, even on untrusted domains in the cloud, while remaining unreadable by those doing the computation. In short, one can now do high-value analytics and data processing, by internal or external parties, without requiring that data to be exposed.

Benefits of homomorphic encryption

1. Gain valuable insights : Generate measurable economic benefits by allowing lines of business and third parties to perform big data analytics on encrypted data while maintaining privacy and compliance controls.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.67
Volume 5, Issue 4, February 2025

2. Collaborate confidently on hybrid cloud : Process encrypted data in public and private clouds and third-party environments while maintaining confidentiality controls.

3. Enable AI, analytics and machine learning (ML) : Use AI and ML to compute upon encrypted data without exposing sensitive information**.**

II. ACKNOWLEDGMENT

This work acknowledges various sources and experts whose contributions have shaped modern cryptography. Special thanks to researchers and organizations that have developed cryptographic standards and security protocols, ensuring safe digital interactions in an increasingly connected world.

III. CONCULSION

Cryptography remains a fundamental pillar of cybersecurity, enabling secure communication, financial transactions, and data protection. While traditional cryptographic methods continue to serve their purpose, emerging threats demand the evolution of more advanced encryption techniques. Post-quantum cryptography and homomorphic encryption offer promising solutions to address future security challenges. As technology advances, continuous research and development in cryptography will be essential in safeguarding digital assets and maintaining trust in online systems.

REFERENCES

[1].https://www.fortinet.com/resources/cyberglossary/what-is-cryptography
(1.1).https://www.geeksforgeeks.org/cryptography-and-its-types/
(1.2).https://www.shiksha.com/online-courses/articles/difference-between-aes-and-des-ciphers-blogId-158769
(1.3).https://www.tutorialspoint.com/cryptography/public_key_encryption.htm
(1.4).https://www.techtarget.com/searchsecurity/definition/elliptical-curve-cryptography
(1.5).https://www.simplilearn.com/tutorials/cyber-security-tutorial/sha-256-algorithm
(1.6).https://www.geeksforgeeks.org/what-is-digital-signature/.
(1.7).https://www.chiragbhalodia.com/2021/09/difference-between-active-and-passive-attack.html#google_vignette
(1.8).https://www.fortinet.com/resources/cyberglossary/brute-force-attack#:~:text=A%20brute%20force%20attack%20is,and%20organizations'%20systems%20and%20networks.
(1.9).https://library.mosse-institute.com/articles/2023/08/cryptography-common-use-cases-and-limitations.html
(1.10).https://www.geeksforgeeks.org/cryptography-and-its-types/
(2.1).https://www.synopsys.com/glossary/what-is-post-quantum-cryptography.html
(2.2).https://www.ibm.com/think/topics/homomorphic-encryption