Md. Ashiq Mahmood
Assistant Professor
Institute of Information and Communication Technology (IICT)
Khulna University of Engineering & Technology (KUET)

Security Protocols

What is Security Protocols?

A security protocol (cryptographic protocol or encryption protocol) is an abstract or concrete protocol that performs a security-related function and applies cryptographic methods, often as sequences of cryptographic primitives.

Types of Security Protocol

- IPSec and VPNs
- SSL and TLS
- Application Transparent Transport Layer Security
- Kerberos
- OSPF authentication
- SNMPv3

IPSec and VPNs

IPSec is defined by the IPSec Working Group of the IETF. It provides authentication, integrity, and data privacy between any two IP entities. Management of cryptographic keys and security associations can be done manually or dynamically using an IETF-defined key management protocol called Internet Key Exchange (IKE).

There are two versions of the IKE protocol:

I. IKE version 1.0 (IKEv1) is defined by RFC 2409, *The Internet Key Exchange (IKE)*, and related RFCs. This is the version that has been supported by z/OS® Communications Server for a number of years.

II. IKE version 2.0 (IKEv2) is defined by RFC 5996, *Internet Key Exchange Protocol: IKEv2*, and related RFCs. Support for IKEv2 is introduced with z/OS V1R12.

IPSec and VPNs(Cont..)

With IPSec, you can create virtual private networks (VPN). A VPN enables an enterprise to extend its private network across a public network, such as the Internet, through a secure tunnel called a security association. IPSec VPNs enable the secure transfer of data over the public Internet for same-business and business-to-business communications, and protect sensitive data within the enterprise's internal network.

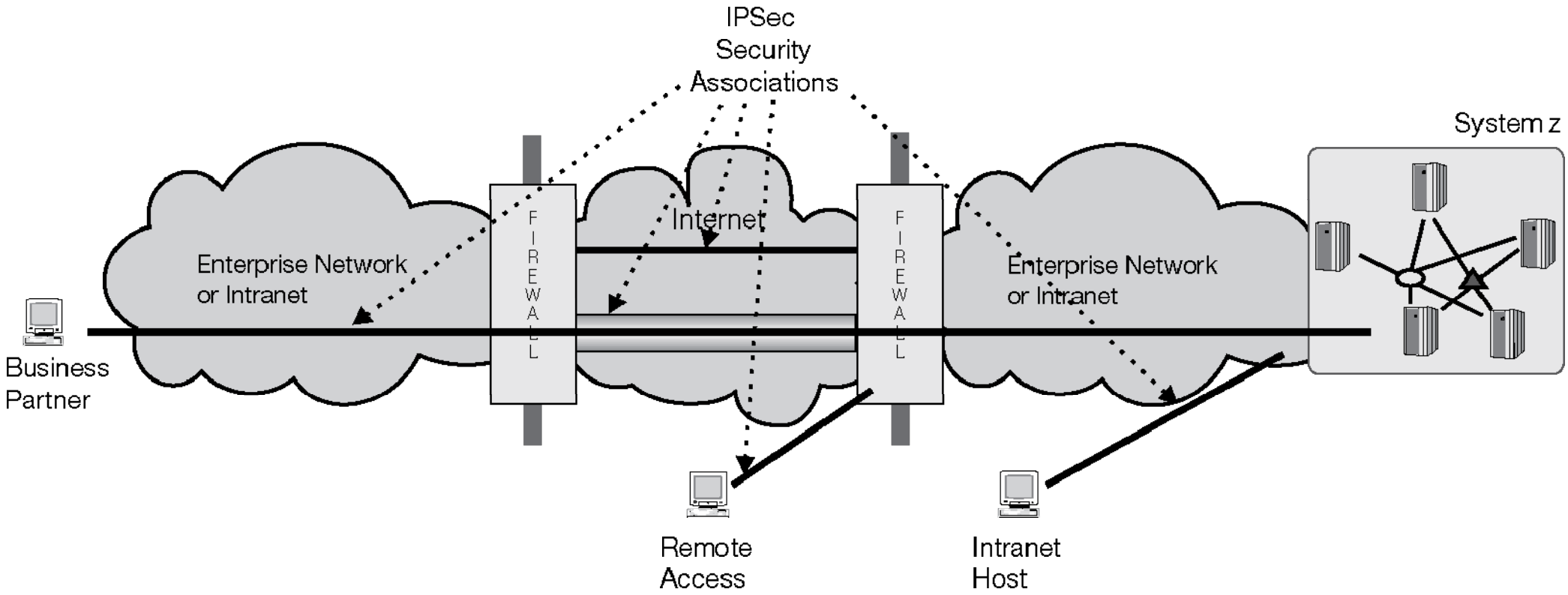


Fig- 1: e-business scenarios with virtual private networks

IPSec and VPNs(Cont..)

z/OS provides support for IKE and IPSec VPNs, including the following options:

- AH and ESP protocols
- Triple DES for strong encryption
- AES with several choices of mode or key length
- IPSec transport and tunnel mode encapsulation
- IKEv1 and IKEv2 negotiations with support for both aggressive and main mode in IKEv1
- Pre-shared key and digital signature methods of authentication
- NAT traversal (IPv4 only)

SSL and TLS

- SSL stands for Secure Sockets Layer and, in short, it's the standard technology for keeping an internet connection secure and safeguarding any sensitive data that is being sent between two systems, preventing criminals from reading and modifying any information transferred, including potential personal details. The two systems can be a server and a client (for example, a shopping website and browser) or server to server (for example, an application with personal identifiable information or with payroll information).

- TLS (Transport Layer Security) is just an updated, more secure, version of SSL.

- HTTPS (Hyper Text Transfer Protocol Secure) appears in the URL when a website is secured by an SSL certificate. The details of the certificate, including the issuing authority and the corporate name of the website owner, can be viewed by clicking on the lock symbol on the browser bar.

.

SSL and TLS(Cont..)

How do SSL certificates work?

SSL works by ensuring that any data transferred between users and websites, or between two systems, remains impossible to read. It uses encryption algorithms to scramble data in transit, which prevents hackers from reading it as it is sent over the connection. This data includes potentially sensitive information such as names, addresses, credit card numbers, or other financial details.

The process works like this:

- A browser or server attempts to connect to a website (i.e., a web server) secured with SSL.
- The browser or server requests that the web server identifies itself.
- The web server sends the browser or server a copy of its SSL certificate in response.
- The browser or server checks to see whether it trusts the SSL certificate. If it does, it signals this to the webserver.
- The web server then returns a digitally signed acknowledgment to start an SSL encrypted session.
- Encrypted data is shared between the browser or server and the webserver.

SSL and TLS(Cont..)

Types of SSL certificate

There are different types of SSL certificates with different validation levels. The six main types are:

- Extended Validation certificates (EV SSL)
- Organization Validated certificates (OV SSL)
- Domain Validated certificates (DV SSL)
- Wildcard SSL certificates
- Multi-Domain SSL certificates (MDC)
- Unified Communications Certificates (UCC)

SSL and TLS(Cont..)

Difference between SSL and TLS

SSL	TLS
1. SSL stands for Secure Socket Layer.	1. TLS stands for Transport Layer Security.
2. SSL (Secure Socket Layer) supports the **Fortezza** algorithm.	2. TLS (Transport Layer Security) does not support the **Fortezza** algorithm.
3. SSL (Secure Socket Layer) is the 3.0 version.	3. TLS (Transport Layer Security) is the 1.0 version.
4. In SSL(Secure Socket Layer), the Message digest is used to create a master secret.	4. In TLS(Transport Layer Security), a Pseudo-random function is used to create a master secret.
5. In SSL(Secure Socket Layer), the Message Authentication Code protocol is used.	5. In TLS(Transport Layer Security), Hashed Message Authentication Code protocol is used.
6. SSL is less reliable and slower.	6. TLS is highly reliable and upgraded. It provides less latency.
7. SSL has been depreciated.	7. TLS is still widely used.
8. SSL uses port to set up explicit connection.	8. TLS uses protocol to set up implicit connection.

Application Transparent Transport Layer Security

Application Transparent Transport Layer Security (AT-TLS) creates a secure session on behalf of an application. Instead of implementing TLS in every application that requires a secure connection, AT-TLS provides encryption and decryption of data based on policy statements that are coded in the Policy Agent.

Applications that are taking advantage of AT-TLS can be separated into three different types (basic, aware and controlling).

Application type	SIOCTTLSCTL ioctl calls issued	ApplicationControlled setting in AT-TLS policy
Basic	application does not issue any AT-TLS ioctl calls	Off
Aware	query requests	Off
Controlling	query and control requests	On

AT-TLS(Cont..)

- A basic application is unaware that AT-TLS is performing encryption or decryption of data. Most applications can match this model.
- An aware application is aware of AT-TLS and can query information such as AT-TLS status, partner certificate, and derived RACF® user ID without any advanced setting in AT-TLS policy. A server that requires a RACF user ID derived from a partner certificate matches this model.
- A controlling application is aware of AT-TLS and needs to control the secure session. It must have the Application Controlled parameter in AT-TLS policy set to ON. Any application that must control when the initial handshake is done or when sessions or ciphers must be reset matches this model.

AT-TLS(Cont..)

The following APIs are supported by AT-TLS:Macro API (EZASMI)

- CALL instruction API (EZASOKET) supporting COBOL, PL/I, and System/370 assembler languages
- REXX socket API
- Language Environment® C socket call [ioctl( )]
- UNIX System Services Assembler Callable Service (BPX1IOC or BPX4IOC)
- CICS® C socket calls
- CICS CALL instruction API (EZASOKET - by including EZACICAL or EZACICSO)
- IMS CALL instruction API (EZASOKET)

TLS Handshake with Client Authentication

Client

Server

ClientHello →

← ServerHello

← Certificate*

← ServerKeyExchange*

Server requests certificate if client needs to be authenticated

← **CertificateRequest**

← ServerHelloDone

Client sends suitable certificate

Certificate* →

If server finds certificate unacceptable; server can send fatal failure alert message & close connection

ClientKeyExchange →

Client prepares digital signature based on messages sent using its private key

CertificateVerify* →

Server verifies client has private key

[ChangeCipherSpec] →

Finished →

← [ChangeCipherSpec]

← Finished

← Application Data →

Kerberos

- Kerberos: authentication service for users to access servers over network
- KDC has secret key with every user
- At login, user supplies ID and password
  - KDC authenticates user & generates session key
  - Session key & ticket-granting ticket (TGT) is sent to user encrypted using shared secret key
- To access a particular server, user sends request to KDC with server name and TGT
  - KDC decrypts TGT to recover session key & then returns ticket to client for desired server

OSPF authentication

- Communications Server OSPF (Open Shortest Path First) dynamic routing protocol supports message authentication and message integrity of OSPF routing messages through the use of the OSPF MD5 Authentication security protocol as defined by RFC 2328. OSPF MD5 Authentication ensures that an unauthorized IP resource cannot inject OSPF routing messages into the network without detection, thus ensuring the integrity of the routing tables in the OSPF routing network.

- OMPROUTE computes a secure MAC for the routing message using the MD5 algorithm. This MAC is sent with the routing message so that the message can be authenticated by the receiver.

SNMPv3

- SNMP stands for **Standard Network Management Protocol**. It is basically an Internet Standard Protocol which is used for monitoring and organizing information about the devices on IP network by sending and receiving requests. This protocol is used for organizing information from devices like switches, modems, routers, servers, printers etc.
- Currently, there are 3 versions of SNMP – SNMPv1, SNMPv2, SNMPv3.

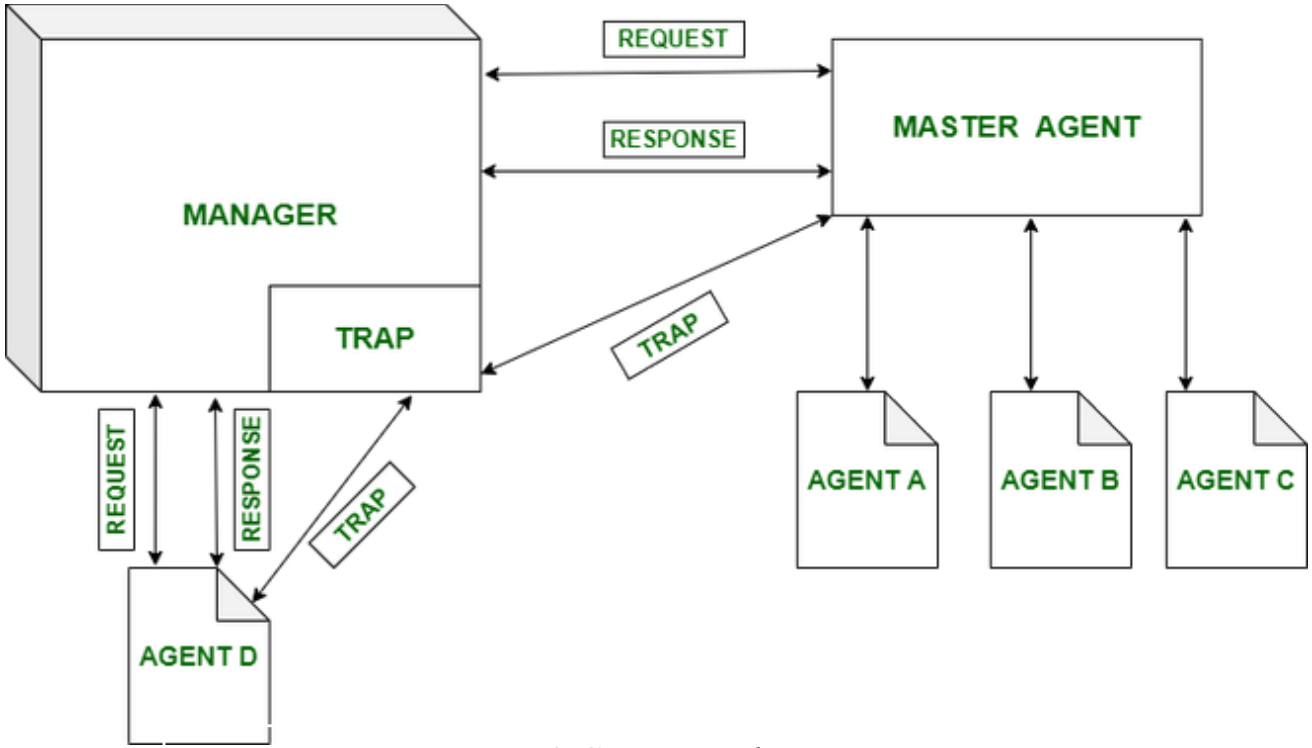


Fig- 2:SNMP architecture.

SNMPv3(Cont..)

Uses of SNMP in Networking :

- It is mainly used for monitoring and organizing networking resources.
- It is a standard internet protocol which is to be followed by everyone. It sets a standard for everyone network management, database management, and organizing data objects.
- Administrator computers (managers) use SNMP for monitoring the clients in the network.
- This protocol allows for management activities using applications like Management Information Base (MIB).

SNMPv3 Architecture :

- The architecture of the v3 consists of –
- Data definition language,
- Definition of MIB
- Protocol definition
- Security and administration.

SNMPv3(Cont..)

Special Features about SNMPv3 :

- v3 is the latest version of SNMP which involves great management services with enhanced security.
- The SNMPv3 architecture makes the use of User-based Security Model (USM) for security of the messages & the View-based Access Control Model (VACM) for accessing the control over the services.
- SNMP v3 security models supports authentication and encrypting.
- SNMPv3 supports Engine ID Identifier, which uniquely identifies each SNMP identity. The Engine ID is used to generate a unique key for authenticating messages.
- v3 provides secure access to the devices that send traps by authenticating users & encrypting data packets which are sent across the network.
- It also introduces the ability to configure and modify the SNMP agent using SET for the MIB objects. These commands enable deletion, modification, configuration and addition of these entries remotely.
- USM – For facilitating remote configuration and management of the security module.
- VACM – For facilitating remote configuration & management for accessing the controlling module.

SNMPv3(Cont..)

Mechanism of version 3 :

- 16-byte key between sender & receiver
- Triple Data Encryption Standard
- Advanced Encryption Standard
- Data Encryption Standard (DES) in Cipher Block Chaining (CBC) mode
- MD5 message-digest algorithm

Key Exchange

What is Key Exchange?

The key exchange method specifies how one-time session keys are generated for encryption and for authentication, and how the server authentication is done. The Diffie-Hellman Key Exchange is a method for exchanging secret keys over a non-secure medium without exposing the keys.

The key exchange methods in OpenSSH allowed in the evaluated configuration are:

- diffie-hellman-group-exchange-sha1
- diffie-hellman-group1-sha1
- diffie-hellman-group-exchange-sha256
- diffie-hellman-group14-sha1

Why key exchange is vital to secure file transfers

- To preserve data confidentiality during transmission, **secure file transfer protocols** like FTPS, HTTPS, and SFTP have to encrypt the data through what is known as **symmetric encryption**.
- In the real world, the two communicating parties would likely be geographically separated by long distances. One party might be in LA, while the other might be in New York, or perhaps even in Japan or Germany. What's more, the two parties might have never met at all.
- The key can't just be sent through ordinary methods because anyone who gets hold of it would then be able to decrypt all the files that the two parties would be sending to one another.

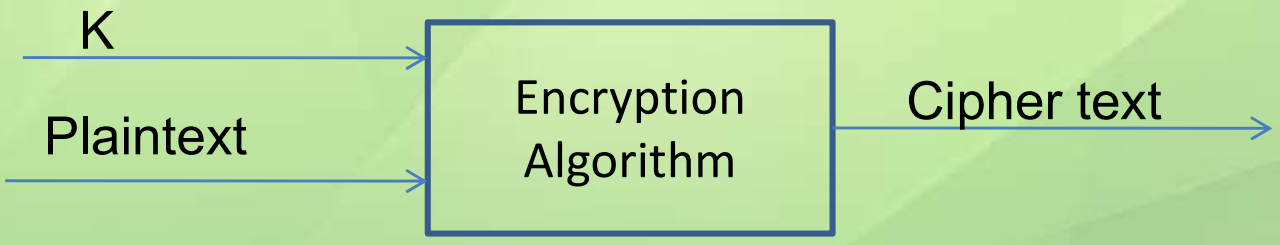
Fig- 3:KEY Exchange.

Basic Concepts of cryptography

Symmetric cryptosystem: here only one key is used in both encryption and decryption processes.

$$C = EA(K, P)$$

Where C-cipher text, EA- encryption algorithm, K- key, P- plain text.

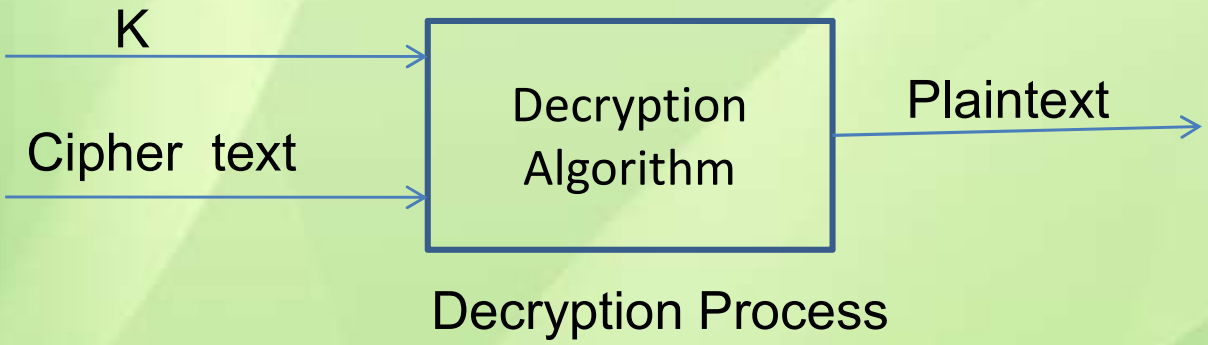


Encryption Process

Symmetric cryptosystem

P = DA(K, C)

Where DA- decryption algorithm.



Decryption Process

In symmetric cryptosystem key must be kept secret.

Asymmetric Cryptosystem

Here two keys are used. One is for encryption and another different one is for decryption. The key used for encryption is called public key and published for general use. The key used for decryption is called private or secret key. The owner will possess this (private) key and must be kept secret. In this system every one who possesses public key can encrypt the message, but only owner of the private key can decrypt the cipher text.

Asymmetric cryptosystem

$C = EA(K_{pub}, P)$

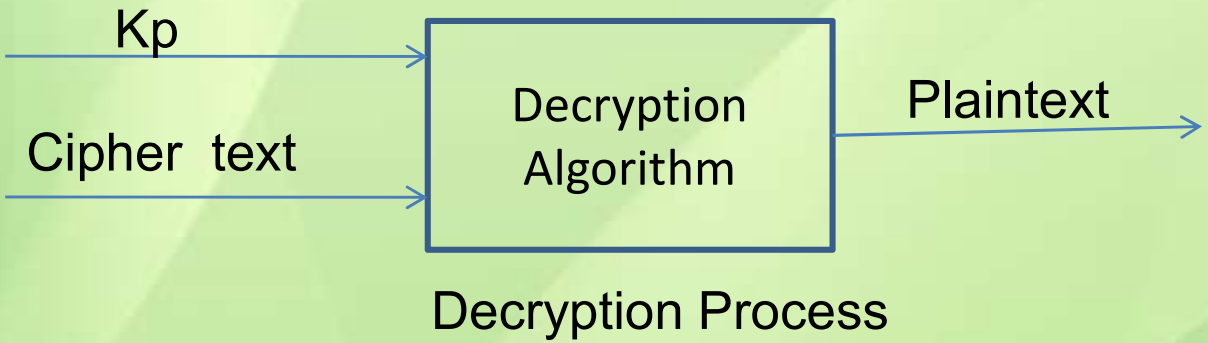
Where K_{pub} is the public key.

Encryption Process

Asymmetric cryptosystem

$$P = DA(K_p, C)$$

Where DA- decryption algorithm.



Decryption Process

In symmetric cryptosystem key must be kept secret.

Outline

- Overview of Cryptography
- Classical Symmetric Cipher
  - Substitution Cipher
  - Transposition Cipher
- Modern Symmetric Ciphers (DES)

Classical Substitution Ciphers

- Letters of plaintext are replaced by other letters or by numbers or symbols
- Plaintext is viewed as a sequence of bits, then substitution replaces plaintext bit patterns with ciphertext bit patterns

Caesar cipher

- Earliest known substitution cipher
- Replaces each letter by 3rd letter on
- Example:

```
meet me after the party
PHHW PH DIWHU WKH SDUWB
```

Caesar Cipher

- Define transformation as:

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

- Mathematically give each letter a number

a	b	c	d	e	f	g	h	i	j	k	l	m
0	1	2	3	4	5	6	7	8	9	10	11	12
n	o	p	q	r	s	t	u	v	w	x	y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

- Then have Caesar cipher as:

$C = E(p) = (p + k) \bmod (26)$

$p = D(C) = (C - k) \bmod (26)$

One-Time Pad

- If a truly random key as long as the message is used, the cipher will be secure - One-Time pad
- E.g., a random sequence of 0's and 1's XORed to plaintext, no repetition of keys
- Unbreakable since ciphertext bears no statistical relationship to the plaintext
- For **any** plaintext, it needs a random key of the same length
  - Hard to generate large amount of keys
- Have problem of safe distribution of key

Transposition Ciphers

- Now consider classical **transposition** or **permutation** ciphers
- These hide the message by rearranging the letter order, without altering the actual letters used
- Can recognise these since have the same frequency distribution as the original text

Transposition cipher

- Write message letters out diagonally over a number of rows
- Then read off cipher row by row
- E.g.,”meet me after the party” write message out as:

```
m e m a t r h p r y
e t e f e t e a t
```

- Giving ciphertext

```
MEMATRHPRYETEFETEAT
```

RSA Cryptosystem

This cryptosystem is invented by Rivest, Shamir and Adleman (RSA) in 1979.

It is a public key cryptosystem, which involves exponentiation modulo a number, *n* that is a product of two large prime numbers.

The 1024 bits key size is a typical key size for RSA cryptosystem.

Key Generation Process

1. Select at random two large prime numbers *p* and *q*.
   (The primes *p* and *q* might be, say, 100 decimal digits each.)
2. Compute *n* by the equation ***n = pq.***

3. **Compute** ***m = (p - 1) (q - 1).***

4. Select a small odd integer ***e*** that is relatively prime to **m**, where *gcd(e, m)=1.*

5. Compute ***d*** as the multiplicative inverse of ***e***, modulo **m** , *i.e.*, ***e*d mod m = 1***
5. Publish the pair ***p = (e, n)*** as RSA public key.

6. Keep secret the pair ***s = (d, n)*** as RSA secret key.

Encryption and Decryption

Encryption Process: The transformation of a message M associated with a public key $p = (e, n)$, is as follows:

$$C = E(M) = M^e \pmod{n}.$$

Decryption Process: The transformation of a cipher text C associated with a secret key $S = (d, n)$ is as follows:

$$M = D(C) = C^d \pmod{n}.$$

Example of RSA cryptosystem

1. *Take* $p = 67$ and $q = 71$
2. *Compute* $n = p*q = 67*71 = 4757$
3. *Compute* $m = \varphi(n) = (p-1)(q-1) = 66*70 = 4620$
4. Choose $e = 83$, such that gcd(e, m) = 1.
5. *Compute* d = m-inv(83, 4620) = 167
6. Public key is (e, n) = (83, 4757)
7. Secret key is (d, n) = (167, 4757)

Example of RSA [cont..]

Take a message: **CONFIDENTIAL**

Message is encoded (letter to digit) as follows:

blank = 00, A = 01, B = 02 and so on.

C	O	N	F	I	D	E	N	T	I	A	L
03	15	14	06	09	04	05	14	20	09	01	12

By taking two letter as a block, we get following data:

0315 1406 0904 0514 2009 0112

Here we must consider that the value of each block must be less than the value of n.

So, $M = (m_1, m_2, m_3, m_4, m_5, m_6) = (315, 1406, 904, 524, 2009, 112)$

Example of RSA [cont..]

By decoding the number to letter we get the original message as follows:

315 = CO, 1406 = NF, 904 = ID, 514 = EN,
2009 = TI, 112 = AL

Then we get the message: **CONFIDENTIAL**

Authentication

What is authentication?

Authentication is the process of determining whether someone or something is, in fact, who or what it says it is. Authentication technology provides access control for systems by checking to see if a user's credentials match the credentials in a database of authorized users or in a data authentication server. In doing this, authentication assures secure systems, secure processes and enterprise information security.

Why is authentication important in cybersecurity?

Authentication enables organizations to keep their networks secure by permitting only authenticated users or processes to gain access to their protected resources. This may include computer systems, networks, databases, websites and other network-based applications or services.

What are authentication factors?

- **Knowledge factor.** The knowledge factor, or *something you know*, may be any authentication credentials that consist of information that the user possesses, including a personal identification number (PIN), a username, a password or the answer to a secret question.

- **Possession factor.** The possession factor, or *something you have*, may be any credential based on items that the user can own and carry with them, including hardware devices, like a security token or a mobile phone used to accept a text message or to run an authentication app that can generate a one-time password (OTP) or PIN.

- **Inherence factor.** The inherence factor, or *something you are*, is typically based on some form of biometric identification, including fingerprints or thumbprints, facial recognition, retina scan or any other form of biometric data.

- **Location factor.** *Where you are* may be less specific, but the location factor is sometimes used as an adjunct to the other factors. Location can be determined to reasonable accuracy by devices equipped with the Global Positioning System or with less accuracy by checking network addresses and routes. The location factor cannot usually stand on its own for authentication, but it can supplement the other factors by providing a means of ruling out some requests. For example, it can prevent an attacker located in a remote geographical area from posing as a user who normally logs in only from their home or office in the organization's home country.

- **Time factor.** Like the location factor, the time factor, or *when you are authenticating*, is not sufficient on its own, but it can be a supplemental mechanism for weeding out attackers who attempt to access a resource at a time when that resource is not available to the authorized user. It may also be used together with location. For example, if the user was last authenticated at noon in the U.S., an attempt to authenticate from Asia one hour later would be rejected based on the combination of time and location.

Authentication vs. authorization

AUTHENTICATION	AUTHORIZATION
• Usually the first step of a security access control	• Usually comes after authentication
• Verifies the user's identity	• Grants or denies permissions to the user do something
• Common methods include: username, password, answer to a security question, code sent via SMS or email	• Permissions are granted and monitored by the organization
• Uses biometric data like fingerprint, face recognition, retinal scan	• Common methods include: role-based access control and attribute-based access control
• It's visible by the user	• It's not visible by the user
• It's changeable by the user	• Cannot be changed by the user

What are the different types of authentication?

- **2FA.** This type of authentication adds an extra layer of protection to the process by requiring users to provide a second authentication factor in addition to the password. 2FA systems often require the user to enter a verification code received via text message on a preregistered mobile phone or mobile device, or a code generated by an authentication application.
- **MFA.** This type of authentication requires users to authenticate with more than one authentication factor, including a biometric factor, such as a fingerprint or facial recognition; a possession factor, like a security key fob; or a token generated by an authenticator app.
- **OTP.** An OTP is an automatically generated numeric or alphanumeric string of characters that authenticates a user. This password is only valid for one login session or transaction and is typically employed for new users or for users who lost their passwords and are given an OTP to log in and change to a new password.
- **Three-factor authentication.** This type of MFA uses three authentication factors -- usually, a knowledge factor, such as a password, combined with a possession factor, such as a security token, and an inherence factor, such as a biometric.
- **Biometrics.** While some authentication systems depend solely on biometric identification, biometrics are usually used as a second or third authentication factor. The more common types of biometric authentication available include fingerprint scans, facial or retina scans, and voice recognition.

What are the different types of authentication?(Cont..)

- **Mobile authentication.** Mobile authentication is the process of verifying users via their devices or verifying the devices themselves. This enables users to log into secure locations and resources from anywhere. The mobile authentication process involves MFA that can include OTPs, biometric authentication or Quick Response code

- **Continuous authentication.** With continuous authentication, instead of a user being either logged in or out, a company's application continually computes an authentication score that measures how sure it is that the account owner is the individual who is using the device.

- **Application programming interface (API) authentication.** The standard methods of managing API authentication are HTTP basic authentication, API keys and Open Authorization (OAuth).

Secret Splitting and Secret Sharing

What is Secret Sharing?

- Secret Sharing refers to cryptographic methods for taking a secret, breaking it up into multiple shares, and distributing the shares among multiple parties, so that only when the parties bring together their respective shares can the secret be reconstructed.

Classification/variants on secret sharing:

Based on the abilities the secret sharing can be classified into:

- **Proactive secret sharing:** In this method, new shares are used and old shares are not considered which helps in updating the shares periodically
- **Dynamic secret sharing:** The ability to change the access structure. The dealer has the ability to change a particular access structure out of a given set and/or to allow the participants to reconstruct different secret (in different time instants)
- **Secret sharing with veto capability:** It is the ability to block the reconstruction. It is a feature where qualified set can prevent any other set of participants from reconstructing the secret key

Depending on the computation power of the participants we have:

- **Computational secret sharing:** Participants (and the dealer) are computationally bounded. CSS allows achieving better information rate. Information rate (ρ) is defined as the ratio between average length of the share (in bits) given to the participants and the length of the secret
- **Verifiable secret sharing:** Dealers and players involved in plain secret sharing, some may or may not follow the protocol. As per verifiable secret sharing, honest players should be able to recover the secret and corrupted players should get no information on it

Based on the techniques used different classes of secret sharing can be identified:

- **Polynomial based secret sharing:** This scheme involves polynomials and interpolations, particularly Lagrange's interpolation for splitting and reconstructing the secret. Shares are evaluations of a randomly generated polynomial
- **CRT schemes:** Its rely on Chinese Remainder Theorem. CRT based, secret sharing scheme shares the secret S among 'n' parties by modulator arithmetic such that any 't' users can reconstruct the secret by the CRT
- **Anonymous secret sharing:** Here the identities of the participants are not required for reconstruction of the secret. The secret can be reconstructed without the knowledge of which participant holds which share
- **Systematic block code based secret sharing:** Multiple groups of secrets are packed into a group of large secrets by using the CRT and then shared by constructing a secret polynomial such that its coefficients are those large secrets
- **Black box secret sharing:** Schemes those are independent of the structure of the group or its order. Black-box secret sharing
- **Visual secret sharing:** Schemes of secrets and the shares are images. Here the picture is cut in to 'n' shares, only if an "n" shares are put together it makes the visible picture if not results in an image of different form

APPLICATIONS OF SECRET SHARING

Secret Sharing has broad applications in the situations where access to important resources has to be protected. Applications includes:

- Byzantine agreement
- E-voting
- Key management in network security
- Multi party secure computation
- Threshold cryptography
- Distributed certificate authorities
- Distributed information storage
- Location privacy
- Key management in ad-hoc networks
- Information hiding
- Secure online auctions
- Fair exchange

Blockchain

What Is a Blockchain?

- A blockchain is a distributed database or ledger that is shared among the nodes of a computer network. As a database, a blockchain stores information electronically in digital format.

How Does a Blockchain Work?

- The goal of blockchain is to allow digital information to be recorded and distributed, but not edited. In this way, a blockchain is the foundation for immutable ledgers, or records of transactions that cannot be altered, deleted, or destroyed. This is why blockchains are also known as a distributed ledger technology (DLT).

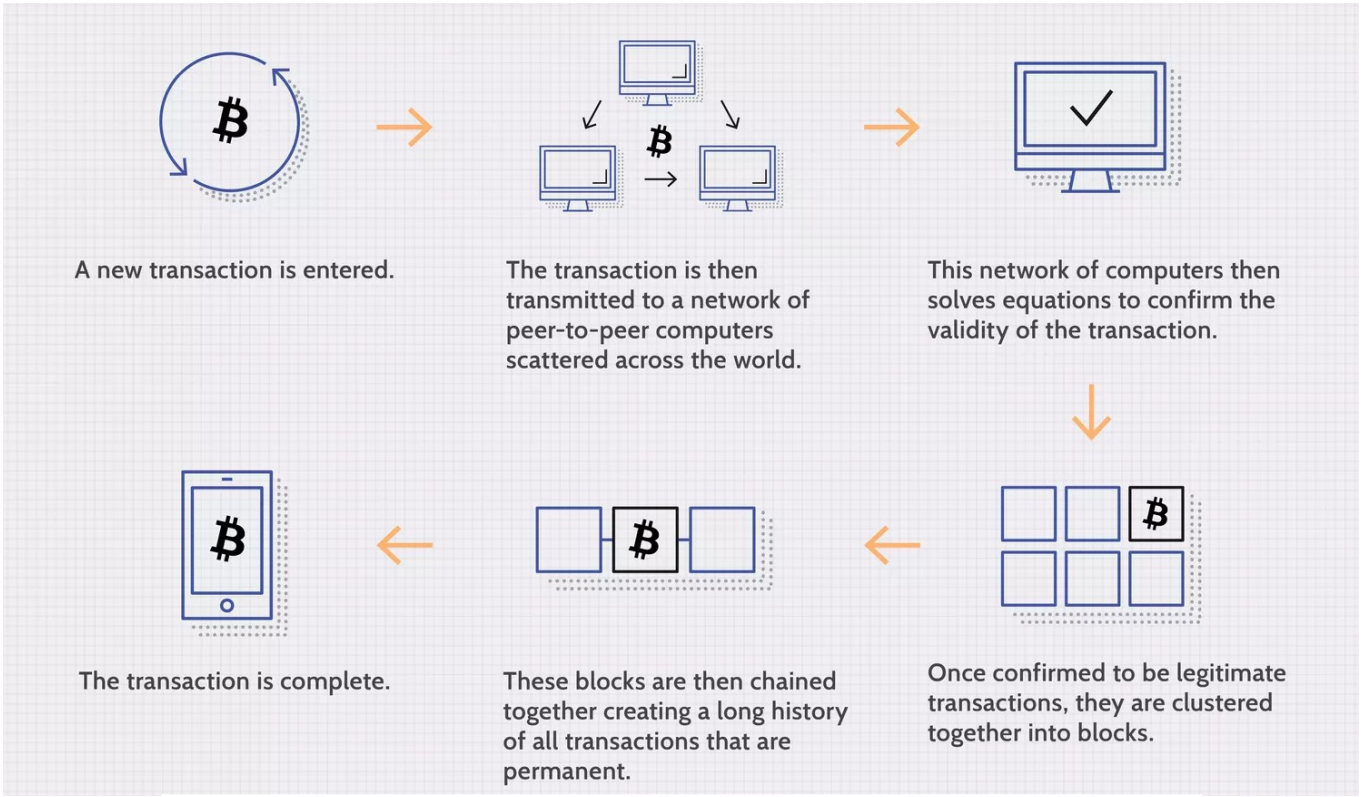


Fig- 5:Transaction Process of Blockchain.

- Attributes of Cryptocurrency

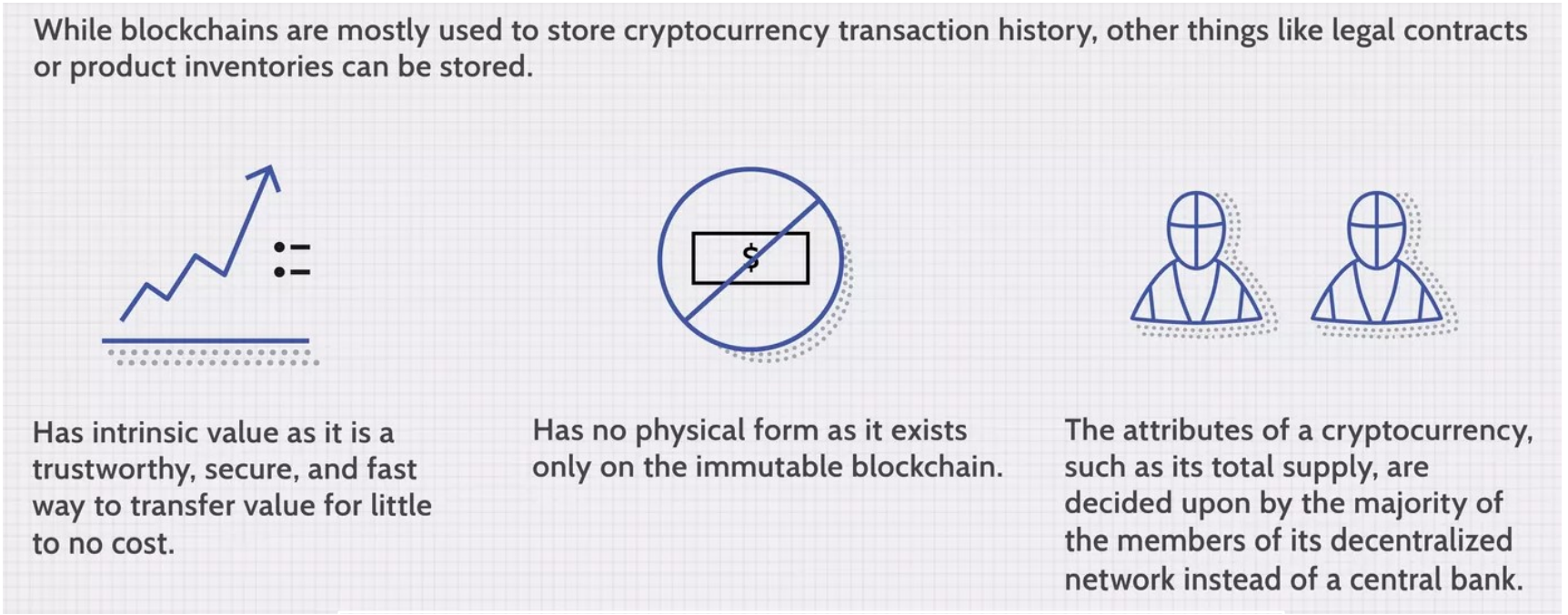


Fig- 6:Atributes of Cryptocurrency.

Blockchain vs. Banks

Feature	Banks	Bitcoin
Transaction Fees	•Card payments: This fee varies based on the card and is not paid by the user directly. Fees are paid to the payment processors by stores and are usually charged per transaction. The effect of this fee can sometimes make the cost of goods and services rise. •Checks: can cost between $1 and $30 depending on your bank. •ACH: ACH transfers can cost up to $3 when sending to external accounts. •Wire: Outgoing domestic wire transfers can cost as much as $25. Outgoing international wire transfers can cost as much as $45.	Bitcoin has variable transaction fees determined by miners and users. This fee can range between $0 and $50 but users have the ability to determine how much of a fee they are willing to pay. This creates an open marketplace where if the user sets their fee too low their transaction may not be processed.
Transaction Speed	•Card payments: 24-48 hours •Checks: 24-72 hours to clear •ACH: 24-48 hours •Wire: Within 24 hours unless international *Bank transfers are typically not processed on weekends or bank holidays	Bitcoin transactions can take as little as 15 minutes and as much as over an hour depending on network congestion.
Hours open	Typical brick-and-mortar banks are open from 9:00 am to 5:00 pm on weekdays. Some banks are open on weekends but with limited hours. All banks are closed on banking holidays.	No set hours; open 24/7, 365 days a year.

Feature	Banks	Bitcoin
Ease of Transfers	Government-issued identification, a bank account, and a mobile phone are the minimum requirements for digital transfers.	An internet connection and a mobile phone are the minimum requirements.
Account Seizures	Due to KYC laws, governments can easily track people's banks accounts and seize the assets within them for a variety of reasons.	If Bitcoin is used anonymously governments would have a hard time tracking it down to seize it.
Approved Transactions	Banks reserve the right to deny transactions for a variety of reasons. Banks also reserve the right to freeze accounts. If your bank notices purchases in unusual locations or for unusual items they can be denied.	The Bitcoin network itself does not dictate how Bitcoin is used in any shape or form. Users can transact Bitcoin how they see fit but should also adhere to the guidelines of their country or region.
Know Your Customer Rules	Bank accounts and other banking products require "Know Your Customer" (KYC) procedures. This means it is legally required for banks to record a customer's identification prior to opening an account.	Anyone or anything can participate in Bitcoin's network with no identification. In theory, even an entity equipped with artificial intelligence could participate.
Privacy	Bank account information is stored on the bank's private servers and held by the client. Bank account privacy is limited to how secure the bank's servers are and how well the individual user secures their own information. If the bank's servers were to be compromised then the individual's account would be as well.	Bitcoin can be as private as the user wishes. All Bitcoin is traceable but it is impossible to establish who has ownership of Bitcoin if it was purchased anonymously. If Bitcoin is purchased on a KYC exchange then the Bitcoin is directly tied to the holder of the KYC exchange account.
Security	Assuming the client practices solid internet security measures like using secure passwords and two-factor authentication, a bank account's information is only as secure as the bank's server that contains client account information.	The larger the Bitcoin network grows the more secure it gets. The level of security a Bitcoin holder has with their own Bitcoin is entirely up to them. For this reason it is recommended that people use cold storage for larger quantities of Bitcoin or any amount that is intended to be held for a long period of time.

How Are Blockchains Used?

- Banking and Finance
- Currency
- Healthcare
- Property Records
- Smart Contracts
- Supply Chains
- Voting

Benefits of Blockchains

- Accuracy of the Chain
- Cost Reductions
- Decentralization
- Efficient Transactions
- Private Transactions
- Secure Transactions
- Transparency
- Banking the Unbanked

Drawbacks of Blockchains

- Technology Cost
- Speed and Data Inefficiency
- Illegal Activity
- Regulation

Cryptocurrency and Cryptonomics

What is cryptocurrency and cryptnomics?

- **Cryptocurrency** is a digital payment system that doesn't rely on banks to verify transactions. It’s a peer-to-peer system that can enable anyone anywhere to send and receive payments.
- The manner in which money is transferred has been altered by crypto, changing the financial and economic landscape. The Economics of the crypto market tells us a market value of 1.92 trillion dollars of capitalization worldwide.

Cryptocurrency examples

- Bitcoin
- Ethereum
- Litecoin
- Ripple

How to buy cryptocurrency

- **Step 1: Choosing a platform**

  The first step is deciding which platform to use. Generally, you can choose between a traditional broker or dedicated cryptocurrency exchange:
  - **Traditional brokers.**
  - **Cryptocurrency exchanges.**
- **Step 2: Funding your account**
- **Step 3: Placing an order**

What can you buy with cryptocurrency?

- Technology and e-commerce sites
- Luxury goods
- Cars
- Insurance

Cryptocurrency fraud and cryptocurrency scams

- Fake websites
- "Celebrity" endorsements
- Romance scams

Comparison Table Of Different Types Of Cryptocurrency

Type	Main feature	Examples
Utility tokens	·Meant to provide access to platform service where they reside.	Funfair, Basic Attention Token, Brickblock,Timicoin, Sirin Labs Token, and Golem.
Security tokens	Usage and issuance governed by financial regulation.	Sia Funds, Bcap (Blockchain Capital), and Science Blockchain.
Payment tokens	Used for paying for goods and services inside and outside their own platforms. Almost every crypto falls in this category.	Monero, Ethereum, and Bitcoin.
Exchange tokens	Exchange tokens are native to crypto exchange platforms.	Binance Coin or BNB token, Gemini USD, FTX Coin for FTX Exchange, OKB for Okex exchange, KuCoin Token, Uni token, HT for Huobi exchange, Shushi, and CRO for Crypto.com.
Non-fungible tokens	Non-fungible tokens are cryptocurrencies with limited issuance that have unique identities and tokens that make them hard to copy or replicate.	Good examples include Logan Paul's video clips, Twitter Founder Jack Dorsey's first tweets NFT, EVERYDAYS: The First 5000 Days drawings by Mike Winklemann, better known as "Beeple", and several crypto kitties.

Different Types Of Cryptocurrency:

#1) Utility Tokens

#2) Security Tokens

#3) Payment Tokens

#4) Exchange Tokens

#5) Non-fungible Tokens

#6) DeFi Tokens Or Decentralized Finance Tokens

#7) Stablecoins – Fiat And Other Types

#8) Asset-backed Tokens

#9) Privacy tokens

Consenus Protocol

What are Consensus Protocols?

- Underlying the decentralized promise of blockchain technologies are consensus protocols. These protocols create a system of agreement, or rules, for a blockchain.

How do Consensus Protocols work?

- The consensus protocol at the heart of a blockchain network gives a specific method for verifying whether a transaction is true or not. It provides a method of review and confirmation of what data should be added to a blockchain's record. Because blockchain networks typically don't have a centralized authority dictating who is right or wrong, nodes on a blockchain all must agree on the state of the network, following the predefined rules, or protocol.

What do consensus protocols do?

- **Prevent a single entity taking control**

  If a network has consensus then all participating nodes agree on the state of a blockchain. Thus, data is recorded as the “truth” and the blockchain is able to function with more and more data added as transactions take place or smart contracts are executed.
- **Allows users on a decentralized to trust users without a controlling third-party**
- A consensus protocol prevents a single entity from controlling a blockchain or distorting the “truth” of what should be recorded.
- **Double spending** is an example of what could happen if one entity tried to take control of the entire network by creating its own version of the blockchain. For example, an attacker could spend some Bitcoin, then alter the block due to be recorded on the blockchain so it doesn’t show the spend. The attacker could broadcast their version of the blockchain, less the spend record. The attacker would have used some Bitcoin, but the coins would not be recorded as spent on the chain and could be spent again.
- Bitcoin’s consensus protocol, PoW, prevents this from happening because when that version of the blockchain is compared to other versions held on other nodes, it will be slightly different from everyone else's, and that version will be rejected by the other nodes.

What are some of the most common types of consensus protocol?

- **Proof-of-Work (PoW):** The first blockchain, Bitcoin, uses PoW. To validate transactions to the Bitcoin blockchain "miners," who are the nodes solve cryptographic, or mathematical problems, using their computers. Miners who solve a problem and validate and enable a block record are rewarded with bitcoin.

- **Proof-Of-Stake (PoS):** Ethereum is moving from PoW to PoS. In PoS there are "forgers" instead of miners. These forgers stake an amount of cryptocurrency which allows them a chance, based on probability, to be a block validator. The successful forger receives the relevant block transaction fees as a reward. Staking their own cryptocurrency on a block provides a disincentive for a forger to try and trick the network as they'll lose the stake if they're proven to be incorrectly adding transactions to the network.

- **Delegated Proof-of-Stake (DPoS):** This method functions in a similar way to PoS. But, instead of using probability, cryptocurrency holders are able to cast votes apportioned to their stake in order to appoint witnesses. These witnesses secure and validate the blockchain, they do not need their own cryptocurrency, but they do need votes. This consensus protocol is more centralized than others. DPoS is used by BitShares, Steem, and EOS.

- **Proof-of-Authority (PoA):** Arguably more centralized again, PoA has predetermined block validators. New blocks on a blockchain are only created when the validators are in majority. The protocol is similar to PoS. The validators are publicly known and accountable for determining their role and eligibility for PoS validation. A newer blockchain, Elysian, uses PoA as well as some Ethereum testnets, or test blockchains.

Permissioned Block Chain

What are the Permissioned Blockchains?

- Permissioned blockchains are blockchain networks that require access to be part of. In these blockchain types, a control layer runs on top of the blockchain that governs the actions performed by the allowed participants. As you can see, permissioned blockchains work entirely different than that of private and public blockchains. They are crafted to take advantage of blockchains without sacrificing the authority aspect of a centralized system.

Benefits of Permissioned Blockchains

- Efficient performance
- Proper governance structure
- Decentralized storage
- Cost-Effective

Drawbacks of Permissioned blockchains

- Compromised security
- Control, Censorship, and Regulation

Types of blockchain and distributed ledger technologies

- **Public blockchains**

  Public blockchains are the most common type of blockchain that allows anyone to participate and do transactions or even participate in the consensus method. There are many prominent public blockchains out there.

  Bitcoin and Ethereum are two great examples. Bitcoin is the first generation cryptocurrency that utilizes the most basic idea of blockchain. Ethereum brings more to the table by providing the developers with the ability to develop distributed apps(dApps) using smart contracts.

- **Federated/Consortium blockchains – Permissioned blockchains**

  When it comes to permissioned blockchain, Federated / Consortium blockchain falls into the category. These blockchain doesn't allow any external people to take part in the blockchain. There are many benefits to it, including higher scalability and is an excellent choice for enterprise companies. There are a lot of permissioned blockchains, including R3, B3i, Hyperledger, and so on.

- **Private blockchain**

  The last type of blockchain that we are going to discuss is private blockchains. Private blockchains are "similar" to permissioned blockchains but have some differences that bring them apart. The private blockchains are not open to the "public" at all, whereas a permissioned blockchain might have some criteria for the public to join. Both of them are restrictive in nature, but their approach differs a little.

Difference between Permissioned and Permissionless Blockchain

Permissioned Blockchain vs Permissionless Blockchain		
Category	**Permissioned**	**Permissionless**
Speed	Faster	Slower
Privacy	Private membership	Transparent and open - anyone can become a member
Legitimacy	Legal	Allegal
Ownership	Managed by a group of nodes pre-defined	Public ownership - no one owns the network
Decentralization	Partially decentralized	Truly decentralized
Cost	Cost-effective	Not so cost-effective
Security	Less secure	More secure

List of Best Permissioned Blockchain

- **Hyperledger Fabric**: Hyperledger Fabric Framework is one of the popular Hyperledger projects that is maintained by Linux foundations. The modular architecture enables anyone to develop solutions or applications to plugin and play with different services. It uses smart contracts known as “chaincode” containing the system’s logic. Want to know more about Hyperledger? Check out our detail coverage here.
- **Quorum**: Quorum blockchain is an enterprise-focused Ethereum blockchain that is aimed at the financial industry. It is created by JP Morgan. We have already covered it in detail.
- **Corda**: Corda lets businesses build interoperable blockchain networks. It is open-source in nature and provides excellent value to businesses.

Challenges of Permissioned Solutions

- **Integration:** One of the biggest challenges that a business or organization has to go through integration challenges, especially when using APIs for communication. To make it easy, you can use Rhombus, Chain Link, and Oraclize. They are great tools that will help you connect and integrate services more efficiently.

- **Data Privacy:** Privacy is a big concern, especially when it comes to regulated industries. To ensure proper privacy, they must meet specific requirements. Quorum and Aztec are a good pick when it comes to developing permissioned blockchains that offer great privacy.

- **Data Access:** Accessing information in a permissioned blockchain can be slow compared. The right solution is to use The Graph which can expose data to the APIs and smart contracts for easy access.

- **Data Storage:** Data can be stored in many ways on a blockchain. However, not all of them are optimal, considering that a huge amount of data is stored on the blockchain. To ensure proper storage, it is advised to use data storage solutions such as IPFS Private Network, Big chain DB, and AWS Quantum Ledger.

- **Identity:** The last challenge that we are going to discuss is the participant's identity problem. As it is permissioned network, the identity of the peers is already known, which can cause issues during the consensus computations. To simplify the issue, it is advised to use Azure BaaS or uPort that will help your business to better leverage their identity protocols and solutions for decentralized applications.

Blockchain Security

What is Blockchain Security?

- Blockchain security is a complete risk management system for blockchain networks, incorporating assurance services, cybersecurity frameworks, and best practices to mitigate the risks of fraud and cyber-attacks.

Blockchain Security Challenges

- **Routing attacks**. Blockchains depend on immense data transfers performed in real-time. Resourceful hackers can intercept the data on its way to ISPs (Internet Service Providers). Unfortunately, blockchain users don't notice anything amiss.

- **51% attacks**. Large-scale public blockchains use a massive amount of computing power to perform mining. However, a group of unethical miners can seize control over a ledger if they can bring together enough resources to acquire more than 50% of a blockchain network's mining power. Private blockchains aren't susceptible to 51% attacks, however.

- **Sybil attacks**. Named for the book that deals with multiple personality disorder, Sybil attacks flood the target network with an overwhelming amount of false identities, crashing the system.

- **Phishing attacks**. This classic hacker tactic works with blockchain as well. Phishing is a scam wherein cyber-criminals send false but convincing-looking emails to wallet owners, asking for their credentials.

How to do Blockchain Penetration Testing?

To make it easy to understand, we've divided the blockchain penetration testing into the following 3 phases:

- **Phase 1: Information Gathering and Threat Modeling**

  In this phase, you can understand and analyze the business and functional requirements.

  This phase includes:

  - Understanding Blockchain architecture
  - Finding threat entry points within the organization
  - Gathering of publicly available data on potential exploits
  - Evaluate Smart Contract Business Logic
  - Setting objectives for conducting security testing
  - Full test strategy designing
  - Checking Compliance readiness
  - Setting up the testing environment
  - Creation of test data

- **Phase 2: Testing/Discovery**

  In this phase, you can use the data acquired in the first phase to play out the active testing of your blockchain to decide its development level estimated against best practices and industry guidelines.

  This phase includes:

  - API Security Testing
  - Functional Testing
  - Automatic and Manual Blockchain Security Analysis
  - Blockchain Static and Dynamic Testing
  - Network Vulnerability Assessment
  - Application Vulnerability Assessment
  - Blockchain Integrity Assessment
  - Documenting Testing Discoveries

- **Phase 3: Exploitation**

  In this phase, the objective is to use any weaknesses or security loopholes found in the Discovery stage. This is frequently done manually to get rid of false positives. The exploitation phase also involves the exfiltration of data from the target and looking after perseverance.

  This phase includes:

  - Verifying Security Weaknesses and Vulnerabilities
  - Exploiting Security Weaknesses and Vulnerabilities
  - Network Penetration Testing
  - Web Application Penetration Testing
  - Test against Social Engineering Attacks
  - Review and Document Discoveries

What are the Blockchain Security Testing tools?

- **SWC-registry** – Smart contract weakness classification and test cases.
- **MythX** – It is a smart contract security analysis API that supports Ethereum, Quorum, Vechain, Roostock, Tron, and other EVM-compatible blockchains.
- **Echidna** – It is a Haskell program designed for fuzzing/property-based testing of Ethereum smarts contracts.
- **Manticore** – It is a symbolic execution tool for the analysis of smart contracts and binaries.
- **Oyente** – A static analysis tool for smart contract security.
- **Securify 2.0** – Securify 2.0 is a security scanner for Ethereum smart contracts.
- **SmartCheck** – Static smart contract security analyzer.
- **Octopus** – It is a security analysis framework for the WebAssembly module and blockchain smart contract.
- **Surya** – Surya is a utility tool for smart contract systems.
- **Solgraph** – Generates a DOT graph that visualizes the function control flow of a Solidity contract and highlights potential security vulnerabilities.
- **Solidity security blog** – Contains a comprehensive list of crypto-related hacks, bugs, vulnerabilities, and preventative measures.
- **Awesome Buggy ERC20 Tokens** – A collection of vulnerabilities in ERC20 smart contracts with tokens affected

Mining Strategies

- The prospect of mining cryptocurrency can be daunting, but it doesn't have to be. The three biggest methods of mining are as follows:
  - software mining,
  - hardware mining, and
  - cloud-based mining.

  In general, most mining is done through speculation, because you need to calculate your profitability when mining cryptocurrency.

General strategies

- Look to mine the most profitable token.
- Look to mine coins that start with a low hash rate and acquire a lot of them – then hope they are added to an exchange.
- Mine the most profitable coin, sell it, and then buy other coins (that may or may not be mineable) that you believe to be the best investment. This is a form of speculative mining.
- Remember to use a mining profitability calculator to determine whether or not you will turn a profit on your desired token. Also, ensure that you calculate your electricity costs, since mining uses a tonne of electricity.

Hardware mining

Hardware mining is incredibly popular, though its initial start-up cost will set you back a lot of money. When looking into mining hardware, it's important to take note of your chosen hardware's hash rate and power consumption. Below is a brief list of possible hardware options:

- **Halong Mining DragonMint T1**: The T1 has an impressive 16 TH per second hash rate, making it one of the most efficient pieces of hardware on the market. However, it does have a consumption of 1,480 Watts, so you will end up using a lot of power.
- **Pangolin Whatsminer M3X**: The M3X is an intensive piece of hardware that uses a ginormous amount of power, ranging between 1800-2100 Watts. This does come with a 12.5 TH per second hash rate, which is good, if still lower than the T1. However, the M3X does cost a lot less than the T1.
- **Bitmain Antminer S9i**: The S9i utilises a Dual ARM Cortex-A9 microprocessor with support for Gigabit Ethernet. This powerful processor enables the hardware to mine blocks that are submitted instantly. The S9i has a hash rate of 14 TH per second, as well as boasting a lower consumption rate than both the T1 and M3X, clocking in at roughly 1,320 Watts.

Hardware mining terms

- **Hash rate**: The hash rate refers to the amount of power a miner uses to solve a mathematical algorithm that mines the cryptocurrency.
- **Power consumption**: The power consumption informs the user how much electricity the mining hardware will use when operating. Power consumption is measured in Watts. To avoid using up huge amounts of power, you'll want to find hardware that uses a low number of Watts.
- **Energy efficiency**: Energy efficiency is measured in Joules. Similar to power consumption, the lower the amount of energy a miner uses, the better. For instance, if the number of Joules is low, it tends to suggest that the miner will consume less power and still put out the same amount of work.

Cloud-based mining

- In simple terms, cloud-based mining refers to mining using the shared power run from remote data centres. The only thing you need to start cloud-based mining is a home computer, a cryptocurrency wallet, and a few other basic essentials.

Pros of cloud-based mining

- You can do it at home without the need for an air-conditioned space to run a mining setup.
- No added electricity costs.
- You don’t need to offload equipment if mining ever stops being profitable.
- No risk of being let down by your mining equipment, which can cost a lot of money.

Cons of cloud-based mining

- High risk of fraud.
- Lower profits since it will be split with the operator.
- Lack of control and flexibility.

Types of cloud-based mining

- **Hosted mining**: This type of cloud-mining is when you are leased a mining machine that is hosted by the provider.
- **Virtual hosted mining**: This type creates a general purpose, virtual private server. You need to install mining software for this method.
- **Renting hash power**: This is where you rent an amount of hashing power from a provider without having a dedicated physical or virtual computer. This method is quite popular.

Mining contracts

- **Genesis-mining**: This is a popular company where users are able to buy hash rates for Bitcoin, Ethereum, and Monero mining.
- **Hashflar.io**: This is another popular site offering some of the better rates in the industry. You can mine based on hash algorithms such as SHA-256, Scrypt, Ethash, and X-11.
- **Hashing24**: This site offers a free demo, so users are able to test the waters if they are unsure about which contract to pick.
- **NiceHash**: This site offers cloud-mining, hash rental services, and multipool. They offer hashing power without contracts on a pay-as-you-go basis.

Decentralized Autonomus Organizations

What is DAO?

- A decentralized autonomous organization (DAO) is an emerging form of legal structure that has no central governing body and whose members share a common goal to act in the best interest of the entity.

Benefits of DAOs

- **Decentralization**. Decisions impacting the organization are made by a collection of individuals as opposed to a central authority that is often vastly outnumbered by their peers. Instead of relying on the actions of one individual (CEO) or a small collection of individuals (Board of Directors), a DAO can decentralize authority across a vastly larger range of users.
- **Participation.** Individuals within an entity may feel more empowered and connected to the entity when they have a direct say and voting power on all matters. These individuals may not have strong voting power, but a DAO encourages token holders to cast votes, burn tokens, or use their tokens in ways they think is best for the entity.
- **Publicity.** Within a DAO, votes are cast via blockchain and made publicly viewable. This requires users to act in ways they feel is best, as their vote and their decisions will be made publicly viewable. This incentivizes actions that will benefit voters' reputations and discourage acts against the community.
- **Community.** The concept of a DAO encourages people from all over the world to seamlessly come together to build a single vision. With just an internet connection, tokenholders can interact with other owners wherever they may live.

Limitations of DAOs

- **Speed.** If a public company is guided by a CEO, a single vote may be needed to decide a specific action or course for the company to take. With a DAO, every user is given an opportunity to vote. This requires a much longer voting period, especially considering time zones and prioritizes outside of the DAO.
- **Education.** Similar to the issue of speed, a DAO has the responsibility of educating a lot more people in regards to pending entity activity. A single CEO is much easier to keep comprised of company developments, while tokenholders of a DAO may have ranging educational backgrounds, understanding of initiatives, incentives, or accessibility to resources. A common challenge of DAOs is that while they bring a diverse set of people together, that diverse set of people must learn how to grow, strategize, and communicate as a single unit.
- **Inefficiency.** Partially summarizing the first two bullets, DAOs run a major risk of being inefficient. Because of the time needed to administrative educate voters, communicate initiatives, explain strategies, and onboard new members, it is easy for a DAO to spend much more time discussing change than implementing it. A DAO may get bogged down in trivial, administrative tasks due to the nature of needing to coordinate much more individuals.
- **Security.** An issue facing all digital platforms for blockchain resources is security. A DAO requires significant technical expertise to implement; without it, there may be invalidity to how votes are cast or decisions made. Trust may be broken and users leave the entity if they can't rely on the structure of the entity. Even through the use of multi-sig or cold wallets, DAOs can be exploited, treasury reserves stolen, and vaults emptied.

DAO Structure

Fig- 7: DAO Structure

DAO Treasury Tools

- **Parcel** - cost-effective treasury tools including payments, payroll, and payment requests
- **Multis** - all-in-one DAO treasury management toolkit built on Gnosis Safe
- **Coinshift** - accounts, multisig tooling, payouts, and accounting built on Gnosis Safe
- **Llama** - multi-signature wallet dApp for DAOs
- **Superfluid** - DAO tools for managing subscriptions, salaries, and income streams
- **Juicebox** - DAO fundraising platform
- **Utopia** - suite of treasury operation tools including payments, accounting, and reporting
- **Request** - payments, payroll, and accounting tools for DAOs

DAO Governance Tools

The most popular Ethereum DAO governance tools include:

- **Snapshot** - platform to submit and vote on governance proposals
- **Tally** - view, vote, and delegate votes for on-chain DAO governance proposals
- **Sybil** - on-chain governance and delegation tool built by Uniswap
- **Commonwealth** - all-in-one platform for discussions, voting, and funding
- **Boardroom** - seamless DAO governance toolkit
- **Paladin** - deposit, borrow and manage governance tokens

DAO Community Building Tools

The most popular Ethereum DAO communication tools include:

- **Discord** - the leading place for teams to communicate
- **Twitter** - the most important social media platforms for DAOs and Web3 companies
- **Telegram** - industry-leading messaging about great for large and small DAOs
- **Discourse** - scalable forum platform for DAOs and Web3 startups
- **Signal** - private messaging app preferred by people who appreciate privacy
- **Medium** - blog publishing platform and alternative to a self-hosted blog
- **Mirror** - article publishing and fundraising platform
- **Collab.land** - token gating tool for creating DAO member-only Discord channels
- **MintGate** - an Ethereum-based platform for building token-gated landing pages
-

Pros and Cons of DAO

Pros

- A range variety of individuals can collectively come together from around the work to act as a single entity.
- More individuals have a voice in the planning, strategy, and operations of the entity.
- As votes on the blockchain are publicly-viewable, tokenholders are naturally incentivized to act more responsibly.
- Members of a DAO may feel empowered to collaborate with like-minded individuals with similar goals within a single community.

Cons

- It often takes longer for decisions to be made as there are more voting participants.
- There is often more burden to educate users as the collective voting population are diverse with varying ranges of education and knowledge.
- More time is needed to cast votes or gather users due to the decentralized nature of the entity.
- Severe exploits such as theft of treasury reserves are possible if the DAO's security is not properly established and maintained.

Special Credit goes to

Monoget Saha

M.Sc. Student of IICT

for assisting to prepare the slide.

THANK YOU