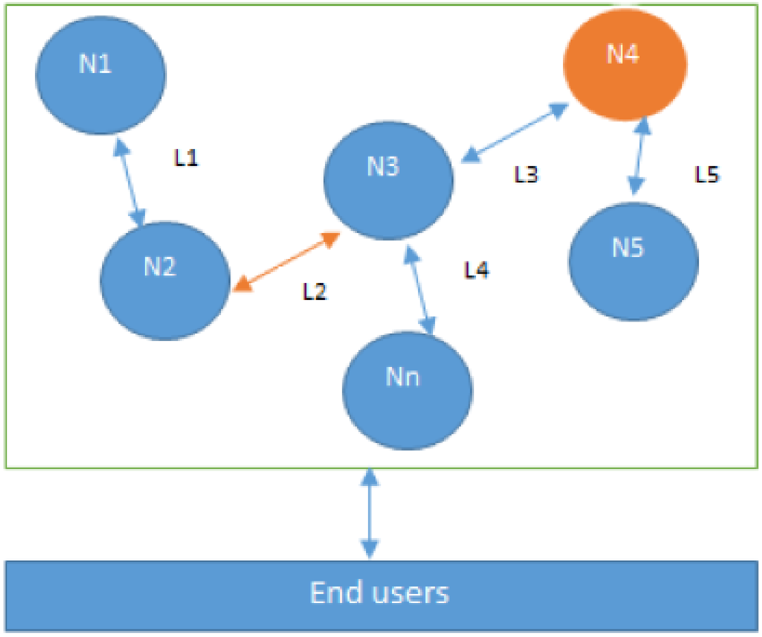
BLOCKCHAIN
Module -1 : Blockchain (BC) 101

History of Blockchain and Bitcoin:
Blockchain was introduced with the invention of Bitcoin in 2008. Itspractical implementation then occurred in 2009.
In 2008, a groundbreaking paper entitled Bitcoin: A Peer-to-Peer ElectronicCash System was written on the topic of peer-to-peer electronic cash underthe pseudonym **Satoshi Nakamoto**. It introduced the term **chain of blocks**. The termchain of blocks evolved over the years into the word **Blockchain**.

Distributed Systems: Distributed systems are a computing paradigm whereby two or more nodeswork with each other in a coordinated fashion to achieve a common outcome.It is modeled in such a way that end users see it as a single logical platform.
For example, Google's search engine is based on a large distributed system,but to a user, it looks like a single, coherent platform.

A node can be defined as an individual player in a distributed system. Allnodes are capable of sending and receiving messages to and from each other.nodes can be honest, faulty, or malicious, and they have memory and aprocessor. A node that exhibits irrational behavior is also known as aByzantine node after the Byzantine Generals Problem.
The inconsistent behavior of Byzantine nodes can be intentionallymalicious, which is detrimental to the operation of the network. Anyunexpected behavior by a node on the network, whether malicious or not, canbe categorized as Byzantine.



Design of a distributed system: N4 is a Byzantine node, L2 is broken or a slow network link

A small-scale example of a distributed system is shown in the followingdiagram. This distributed system has six nodes out of which one (**N4**) is aByzantine node leading to possible data inconsistency. **L2** is a link that isbroken or slow, and this can lead to partition in the network.

The primary challenge in distributed system design is coordination betweennodes and fault tolerance. Even if some of the nodes become faulty ornetwork links break, the distributed system should be able to tolerate this andcontinue to work to achieve the desired result.

Electronic Cash(eCash): eCash was a digital-based system that facilitated the transfer of funds anonymously. A pioneer in cryptocurrency, its goal was to secure the privacy of individuals that use the Internet for micropayments.

There are two forms of eCash, an online form and an offline form.

Online eCash: The term eCash was originally used by a company called DigiCash, founded by David Chaum.

With online eCash, information regarding currency is downloaded to a hard drive. It stays there until it is transferred to another person or business online. This is the basis of cryptocurrency, in a very simple way.

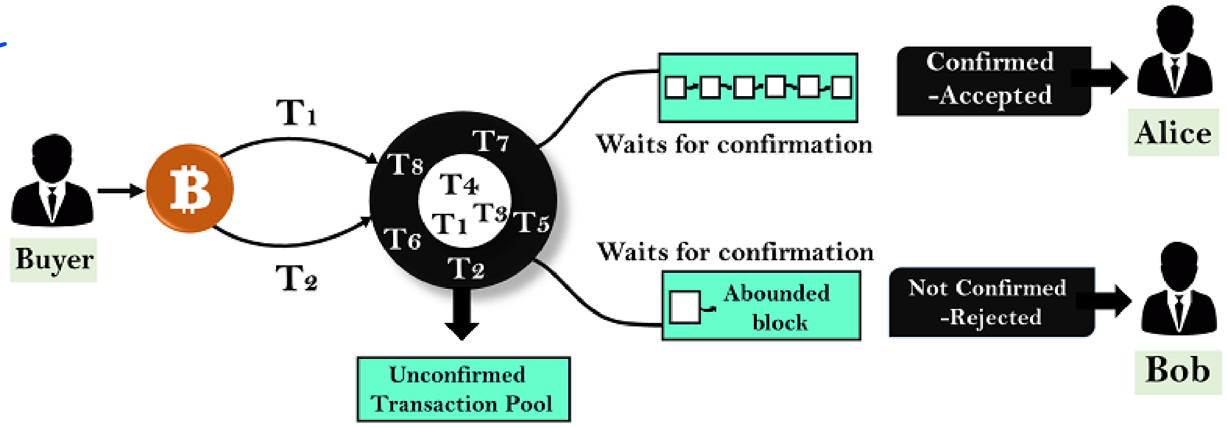
Offline eCash:The idea behind offline eCash has its roots in credit cards and debit cards. Offline eCash would function similarly to a debit card. Funds from a hard drive would be linked to a digitally encoded card. This card would replace paper money (like a debit card). However, the main difference here is that physical money no longer exists to begin with. With a debit card, physical money is still present, in a way.

Two fundamental e-cash system issues need to be addressed: accountability and anonymity.
Accountability is required to ensure that cash is spendable only once (double-spend problem) and that it can only be spent by its rightful owner. Double spend problem arises when same money can be spent twice. As it is quite easy to make copies of digital data, this becomes a big issue in digital currencies as you can make many copies of same digital cash.
Anonymity is required to protect users' privacy. As with physical cash, it is almost impossible to trace back spending to the individual who actually paid the money. This means you can carry out transactions with your real credit card or real bank account without the personal details of your account being revealed.

Double spending means spending the same money twice.

In a physical currency, the double-spending problem can never arise. But in digital cash-like bitcoin, the double-spending problem can arise. Hence, bitcoin transactions have a possibility of being copied and rebroadcasted. It opens up the possibility that the same BTC could be spent twice by its owner.

Let us suppose you have 1 BTC and try to spend it twice. You made the 1 BTC transactions to Alice. Again, you sign and send the same 1 BTC transaction to Bob. Both transactions go into the pool of unconfirmed transactions where many unconfirmed transactions are stored already. The unconfirmed transactions are transactions which do not pick by anyone. Now, whichever transaction first got confirmations and was verified by miners, will be valid. Another transaction which could not get enough confirmations will be pulled out from the network. In this example, transaction T1 is valid, and Alice will receive the bitcoin.



Anonymity refers to the absence of identifying information associated with an interaction. On-line interactions can facilitate both more and less anonymity than those carried out in the physical world. Interpersonal transactions across the Internet allow greater anonymity at one level, but there is often an identifying data trail left by the Internet user. Such data can include names, date-of-birth, credit card numbers, mailing addresses and buying patterns.

Accountability An action is accountable if it can be attributed to someone (or something – such as a service provider.

Accountability An action is accountable if it can be attributed to someone (or something – such as a service provider.

David Chaum solved both of these problems during his work in 1980s by using two cryptographic operations, namely blind signatures and secret sharing.

Blind signaturesallow for signing a document without actually seeing it. Blind signatures are typically employed in privacy-related protocols where the signer and message author are different parties. The money used in the system was called "CyberBucks."Examples include cryptographic election systems and digital cash schemes.

Secret sharing isa concept that enables the detection of double spending.

Blockchain Definition

***Layman's definition**: Blockchain is an ever-growing, secure, shared record keepingsystem in which each user of the data holds a copy of the records, which can onlybe updated if all parties involved in a transaction agree to update.*

Technical definition**: **Blockchain is a peer-to-peer, distributedledger that iscryptographically-secure, append-only, immutable (extremely hard to change), andupdateable only via consensus or agreement among peers.

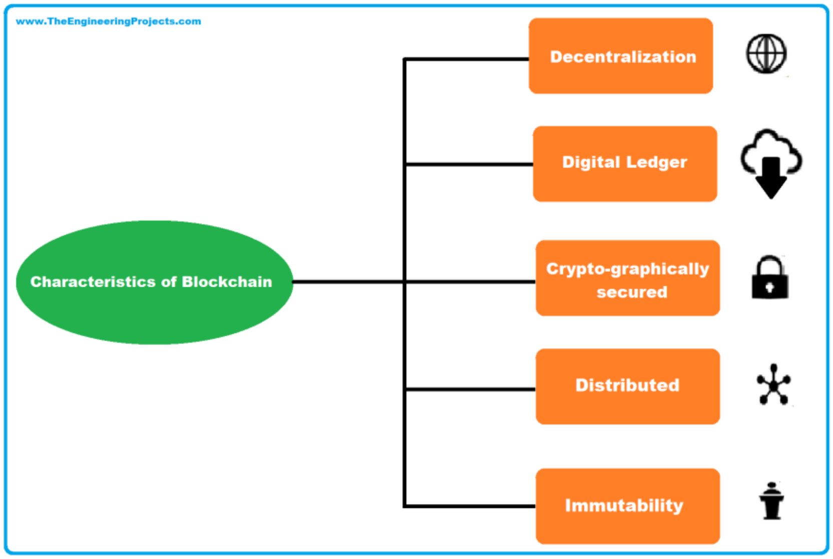
Peer-to-peer : This means thatthere is no central controller in the network, and all participants talk to eachother directly. This property allows for cash transactions to be exchangeddirectly among the peers without a third-party involvement, such as by abank.

Distributed ledger:A ledger is spread across thenetwork among all peers in the network, and each peer holds a copy of thecomplete ledger.

Cryptographically-secure: Ledger is cryptographically-secure, which means thatcryptography has been used to provide security services which make thisledger secure against tampering and misuse. These services include nonrepudiation,data integrity, and data origin authentication.

Append-only: Another property that we encounter is that blockchain is append-only, whichmeans that data can only be added to the blockchain in time-orderedsequential order. This property implies that once data is added to theblockchain, it is almost impossible to change that data and can be consideredpractically immutable.

Updateable via consensus: The most critical attribute of a blockchain is that it is updateable onlyvia consensus. This is what gives it the power of decentralization. In thisscenario, no central authority is in control of updating the ledger. Instead, anyupdate made to the blockchain is validated against strict criteria defined bythe blockchain protocol and added to the blockchain only after a consensushas been reached among all participating peers/nodes on the network. Toachieve consensus, there are various consensus facilitation algorithms whichensure that all parties are in agreement about the final state of the data on theblockchain network and resolutely agree upon it to be true.

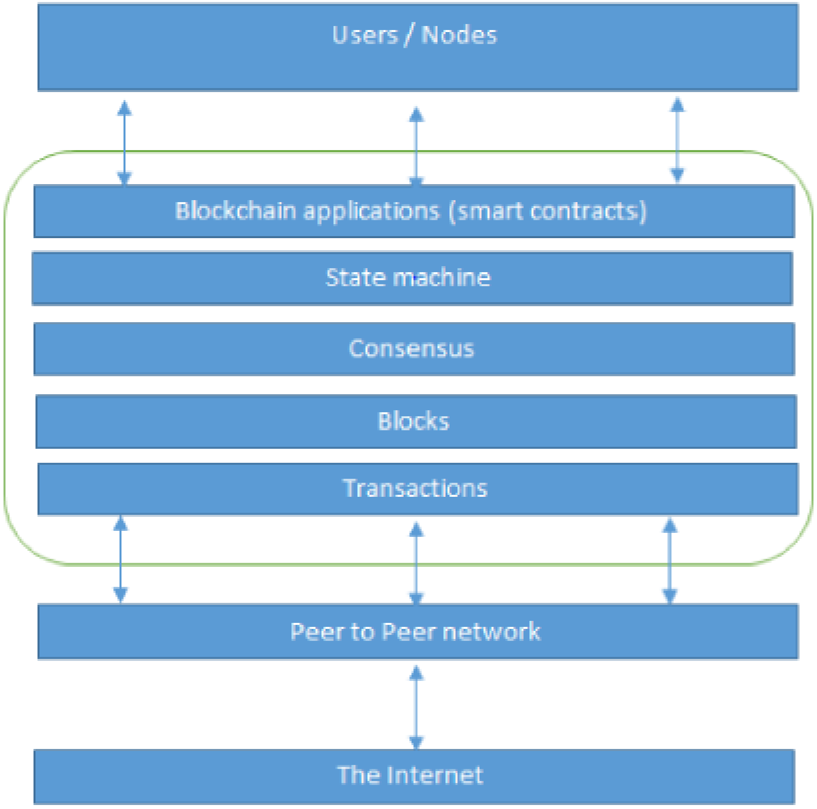

www.TheEngineeringProjects.com
Characteristics of Blockchain
Decentralization
Digital Ledger
Crypto-graphically secured
Distributed
Immutability

Network view of Blockchain

Blockchain can be thought of as a layer of a distributed peer-to-peer networkrunning on top of the internet, as can be seen in the following diagram. It isanalogous to SMTP, HTTP, or FTP running on top of TCP/IP.

At the bottom layer ,there is the internet, whichprovides a basic communication layer for any network. A peer-to-peer network runs on top of the internet, which hosts another layer ofblockchain. That layer contains transactions, blocks, consensus mechanisms,state machines, and blockchain smart contracts. All of these components area single logical entity, representing blockchain.

At the top, there are users or nodes thatconnect to the blockchain and perform various operations such as consensus,transaction verification, and processing.



The network view of a blockchain

From a business standpoint, a blockchain can be defined as a platform wherepeers can exchange value / electronic cash using transactions without theneed for a centrally-trusted arbitrator. For example, for cash transfers, banksact as a trusted third party. In financial trading, a central clearing house actsas an arbitrator between two trading parties.

Key Terms

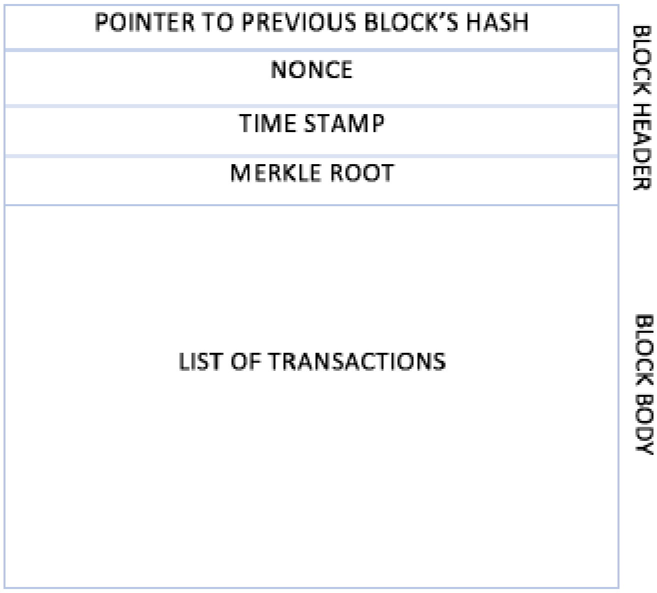
Block: A block is merely a selection of transactions bundled together and organizedlogically. A reference to a previous block is also included in the block unless it is agenesis block.The structure of ablock is dependent on the type and design of a blockchain. There are a few attributes that are essential to the functionalityof a block: the **block header**, which is composed of pointer to previousblock, the **timestamp**, **nonce**, **Merkleroot**, and the **block body** that containstransactions.

Transaction: A transaction is a record of an event, for example, the event oftransferring cash from a sender's account to a beneficiary's account. A blockis made up of transactions, and its size varies depending on the type anddesign of the blockchain in use.

Genesis block: A genesis block is the first block in the blockchain that ishardcoded at the time the blockchain was first started.

Nonce: Nonce is an abbreviation for "number only used once," which is a number added to a hashed—or encrypted—block in a blockchain
A nonce is used extensively in many cryptographic operations to provide replay protection, authentication, and encryption. In blockchain, it's used inPoW consensus algorithms and for transaction replay protection.

Merkle root is a hash of all of the nodes of a Merkle tree. Merkle trees arewidely used to validate the large data structures securely and efficiently. Inthe blockchain world, Merkle trees are commonly used to allow efficientverification of transactions. Merkle root in a blockchain is present in theblock header section of a block, which is the hash of all transactions in ablock. This means that verifying only the Merkle root is required to verify alltransactions present in the Merkle tree instead of verifying all transactionsone by one.



The generic structure of a block.

Generic elements of a blockchain

The elements of Blockchainare :

Address: Addresses are unique identifiers used in a blockchaintransaction to denote senders and recipients.

-An address is apublic key or derived from a public key.
-addresses can be reusedby the same userand are unique.
-a user generates a new onefor each transaction.
-Bitcoinis a pseudonymous system. End users are usually not directlyidentifiable
-a good practiceis for users to generate a new address for each transaction in order toavoid linking transactions to the common owner, thus preventingidentification.

Transaction: A transaction is the fundamental unit of a blockchain. Atransaction represents a transfer of value from one address to another.

Block: A block is composed of multiple transactions and other elements,such as the previous block hash (hash pointer), timestamp, and nonce.

Peer-to-peer network: isa network topology wherein all peers can communicate with each otherand send and receive messages.

Scripting or programming language:

-Scripts or programs performvarious operations on a transaction in order to facilitate variousfunctions.
-example, in Bitcoin, transaction scripts are predefined in alanguage called **Script**, which consist of sets of commands that allownodes to transfer tokens from one address to another.
-Script is a limitedlanguage, that it only allows essential operationsthat are necessary for executing transactions, but it does not allow forarbitrary program development.
-Bitcoin script language cannot be called Turing complete.
-Turing complete language means that it can perform anycomputation. It is named after Alan Turing who developed the idea of Turing machine
-Turingcomplete languages need loops and branching capability to performcomplex computations.
-Bitcoin's scripting language is not Turing complete, whereas Dthereum'sSolidity language is.

Virtual machine:

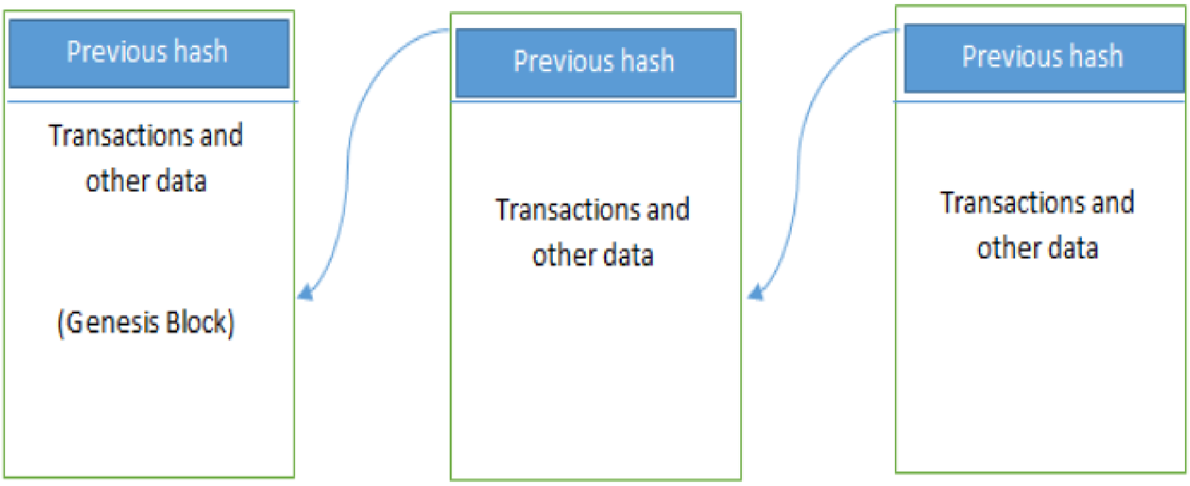
A virtual machine allows turing complete code to berun on a blockchain (as smart contracts); whereas a transaction script islimited in its operation.
-virtual machines are not available onall blockchains.
-Various blockchains use virtual machines to runprograms such as Ethereum Virtual Machine (EVM) and ChainVirtual Machine (CVM).
-EVM is used in Ethereumblockchain, CVM is a virtual machine developed for and used in an enterprise-gradeblockchain called Chain Core.

State machine:

-Blockchainis a state transitionmechanism where a state is modified from its initial form to the nextone and eventually to a final form by nodes on the blockchain network asa result of a transaction execution, validation, and finalization process.

Node:
-node in a blockchain network performs various functionsdepending on the role that it takes on.
-node can propose and validatetransactions and perform mining to facilitate consensus and secure theblockchain. This is achieved by a consensus protocol
-Nodes can also perform other functions such assimple payment verification (lightweight nodes), validation, and manyother functions depending on the type of the blockchain used and the roleassigned to the node.
-Nodes also perform a transaction signing function. Transactions are first created by nodes and then digitally signed bynodes using private keys as proof that they are the legitimate owner
-This is a token or virtual currency, such asBitcoin, but it can also be any real-world asset represented on theblockchain by using tokens.

Smart contract:
-These programs run on top of the blockchain andencapsulate the business logic to be executed when certain conditionsare met.
-are enforceable and automatically executable.
-is not available on all blockchain platforms,
-have many use cases, including identity management,capital markets, trade finance, record management, insurance, and egovernance.


Previous hash
Transactions and other data
(Genesis Block)
Previous hash
Transactions and other data
Previous hash
Transactions and other data

Generic structure of a blockchain

How blockchain works

Nodes are either **miners** who create new blocks and mint cryptocurrency (coins) or **block signers**who validates and digitally sign the transactions. A critical decision that every blockchain network has to make is to figure out that which node will append the next block to the blockchain. This decision is made using a consensus mechanism.

How blockchain accumulates blocks

A general scheme for creating blocks and relationship is between transactions and blocks:

1. A node starts a transaction by first creating and then digitally signing it with its private key. A transaction can represent various actions in a blockchain. Most commonly this is a data structure that represents transfer of value between users on the blockchain network. Transaction data structure usually consists of some logic of transfer of value, relevant rules, source and destination addresses, and other validation information.
2. A transaction is propagated (flooded) by using a flooding protocol, called Gossip protocol, to peers that validate the transaction based on preset criteria. Usually, more than one node are required to verify the transaction.
3. Once the transaction is validated, it is included in a block, which is then propagated onto the network. At this point, the transaction is considered confirmed
4. The newly-created block now becomes part of the ledger, and the next block links itself cryptographically back to this block. This link is a hash pointer. At this stage, the transaction gets its second confirmation and the block gets its first confirmation.
5. Transactions are then reconfirmed every time a new block is created. Usually, six confirmations in the Bitcoin network are required to consider the transaction final.

Benefits and limitations of blockchain

- **Decentralization**:There is no need for a trusted third party or intermediary to validate transactions; instead, a consensus mechanism is used to agree on the validity of transactions.
- **Transparency and trust**: blockchains are shared and everyone can see what is on the blockchain, this allows the system to be transparent. As a result, trust is established. This is relevant in scenarios such as the disbursement of funds or benefits .
- **Immutability**: Once the data has been written to the blockchain, it is extremely difficult to change it back.
- **High availability**: As the system is based on thousands of nodes in a peer-to-peer network, and the data is replicated and updated on every node, the system becomes highly available. Even if some nodes leave the network or become inaccessible, the network as a whole continues to work, thus making it highly available. This redundancy results in high availability.
- **Highly secure**: All transactions on a blockchain are cryptographically secured and thus provide network integrity.
- **Simplification of current paradigms:**blockchain can serve as a single shared ledger among many interested parties, this can result in simplifying the model by reducing the complexity of managing the separate systems maintained by each entity.
- **Faster dealings**: In the financial industry, especially in post-trade settlement functions, blockchain can play a vital role by enabling the quick settlement of trades. Blockchain does not require a lengthy process of verification, reconciliation, and clearance because a single version of agreed-upon data is already available on a shared ledger between financial organizations.

- **Cost saving**: As no trusted third party or clearing house is required in the blockchain model, this can massively eliminate overhead costs in the form of the fees which are paid to such parties.

The most sensitive blockchain problems / challenges are as follows:

- Scalability
- Adaptability
- Regulation
- Relatively immature technology
- Privacy

Tiers of blockchain technology

Blockchain 1.0: This tier was introduced with the invention of Bitcoin, and it is primarily used for cryptocurrencies. The first generation of blockchain technology includes only cryptographic currencies. All alternative cryptocurrencies as well as Bitcoin fall into this category. It includes core applications such as payments and applications. This generation started in 2009 when Bitcoin was released and ended in early 2010.

Blockchain 2.0: Second blockchain generation is used by financial services and smart contracts. This tier includes various financial assets, such as derivatives, options, swaps, and bonds. Applications that go beyond currency, finance, and markets are incorporated at this tier. Ethereum, Hyperledger, and other newer blockchain platforms are considered part of Blockchain 2.0. This generation started when ideas related to using blockchain for other purposes started to emerge in 2010.

Blockchain 3.0: Third blockchain generation is used to implement applications beyond the financial services industry and is used in government, health, media, the arts, and justice. Ethereum, Hyperledger, and newer blockchains with the ability to code smart contracts are considered part of this blockchain technology tier. This generation of blockchain emerged around 2012 when multiple applications of blockchain technology in different industries were researched.

Blockchain X.0: This generation represents a vision of blockchain singularity where one day there will be a public blockchain service available that anyone can use just like the Google search engine. It will provide services for all realms of society. It will be a public and open distributed ledger with general-purpose rational agents (Machinaeconomicus) running on a blockchain.

MachinaEconomicus is a concept which comes from the field of Artificial Intelligence (AI) and computational economics. It can be defined as a machine that makes logical and perfect decisions.

Smart contracts are not supported by all blockchain platforms, such as Bitcoin. Not all blockchain platforms produce cryptocurrency or tokens, such as Hyperledger Fabric, and MultiChain

Features of a blockchain

Distributed consensus: allows a blockchain to present a single version of the truth, which is agreed upon by all parties without the requirement of a central authority.

Transaction verification: Any transactions posted from the nodes on the blockchain are verified based on a predetermined set of rules. Only valid transactions are selected for inclusion in a block.

Platform for smart contracts: A blockchain is a platform on which programs can run to execute business logic on behalf of the users. Not all blockchains have a mechanism to execute smart contracts; it is available on newer blockchain platforms such as Ethereum and MultiChain.

Transferring value between peers: Blockchain enables the transfer of value between its users via tokens. Tokens can be thought of as a carrier of value.

Generation of cryptocurrency: This feature is optional depending on the type of blockchain in use. A blockchain can create cryptocurrency as an incentive to its miners who validate the transactions and spend resources to secure the blockchain.

Smart property: It is now possible to link a digital or physical asset to the blockchain in such a secure and precise manner that it cannot be claimed by anyone else. You are in full control of your asset, and it cannot be double-spent or double-owned. Examples include Oculus hack, PS3 hack.

Provider of security: The blockchain is based on proven cryptographic technology that ensures the integrity and availability of data. Generally, confidentiality is not provided due to the requirements of transparency. This limitation is the leading barrier to its adoption by financial institutions and other industries that require privacy and confidentiality of transactions.Other security services, such as non-repudiation and authentication, are also provided by blockchain, as all actions are secured using private keys and digital signatures.

Immutability: This is another critical feature of blockchain: once records are added to the blockchain, they are immutable. There is the remote possibility of rolling back changes, but this is to be avoided at all costs as doing so would consume an exorbitant amount of computing resources. For example, with Bitcoin if a malicious user wants to alter previous blocks, then it would require computing the PoW once again for all those blocks that have already been added to the blockchain. This difficulty makes the records on a blockchain essentially immutable.

Uniqueness: This blockchain feature ensures that every transaction is unique and has not already been spent (double-spend problem). This feature is especially relevant with cryptocurrencies, where detection and avoidance of double spending are a vital requirement.

Types of blockchain

1. Distributed ledgers

All blockchains are fundamentally distributed ledgers, but all distributed ledgers are not necessarily a blockchain.

A critical difference between a distributed ledger and blockchain is that a distributed ledger does not consist of blocks of transactions to keep the ledger growing. Rather, a blockchain is a special type of shared database that is comprised of blocks of transactions. An example of a distributed ledger that does not use blocks of transactions is R3's Corda.

A distributed ledger is distributed among its participants and spread across multiple sites or organizations. This type of ledger can be either private or public. The records are stored contiguously instead of being sorted into blocks. This concept is used in Ripple which is a blockchain and cryptocurrency based global payment network

2. Distributed Ledger Technology

It should be noted that over the last few years, the terms distributed ledger or Distributed Ledger Technology (DLT) have grown to be commonly used to describe blockchain in finance industry. DLT is now a very active and thriving area of research in the financial sector.

From a financial sector point of view, DLTs are permissioned blockchains that are shared and used between known participants. DLTs usually serve as a shared database, with all participants known and verified. They do not have a cryptocurrency or do not require mining to secure the ledger.

3. Public blockchains

-not owned by anyone.

- are open to the public, and anyone can participate as a node in the decision-making process.

-Users may or may not be rewarded for their participation.

-All users of these permissionless or unpermissioned ledgers maintain a copy of the ledger on their local nodes

-use a distributed consensus mechanism to decide the state of the ledger.

-Bitcoin and Ethereum are both considered public blockchains.

4. Private blockchains

-are private.

-are open only to a group of individuals or organizations who have decided to share the ledger among themselves.

-HydraChain and Quorum come under this category.

-Optionally, both of these blockchains can also run in public mode if required

- **Semiprivate blockchains**

-hybrid model

-part of the blockchain is private and part of it is public.

- this is still a concept today, no real world POCs have yet been developed.

-the private blockchain, the private part is controlled by a group of individuals, while the public part is open for participation by anyone.

-used in scenarios where the private part of the blockchain remains internal and shared among known participants, while the public part of the blockchain can still be used by anyone, optionally allowing mining to secure the blockchain.

-the blockchain as a whole can be secured using PoW, thus providing consistency and validity for both the private and public parts.

-this type of blockchain can also be called a **semi-decentralized model**, where it is controlled by a single entity but still allows for multiple users to join the network by following appropriate procedures.

- **Sidechains / Pegged Sidechains**

-coins can be moved from one blockchain to another and moved back again.

-usedin the creation of new altcoins (alternative cryptocurrencies) whereby coins are burnt as a proof of an adequate stake.

-Burnt or burning the coins means that the coins are sent to an address which is unspendable and this process makes the burnt coins irrecoverable.

-mechanism is used to bootstrap a new currency or introduce scarcity which results in increased value of the coin.

-This mechanism is also called Proof of Burn (PoB) and is used as an alternative method for distributed consensus to PoW and Proof of Stake (PoS).

-This applies to a one-way pegged sidechain.

-Second type is a two-way pegged sidechain, which allows the movement of coins from the main chain to the sidechain and back to the main chain when required.
-Rootstock is a leading examples of a sidechain, which enables smart contract development for Bitcoin
-It works by allowing a two-way peg for the Bitcoinblockchain, and this results in much faster throughput.

- **Permissioned ledger**

-participants of the network are already known and trusted.
-do not need to use a distributed consensus mechanism; instead, an agreement protocol is used to maintain a shared version of the truth
-for verification of transactions on the chain, all verifiers are already preselected by a central authority and there is no need for a mining mechanism.
-For example, Bitcoin can become a permissioned ledger if an access control layer is introduced on top of it that verifies the identity of a user and then allows access to the blockchain.

5. Shared ledger

This is used to describe any application or database that is shared by the public or a consortium. Generally, all blockchains, fall into the category of a shared ledger.

6. **Fully private and proprietary blockchains**

-no mainstream application of these types of blockchains, as they deviate from the core concept of decentralization in blockchain technology.
-example: to allow for collaboration and the sharing data between various government departments.
-no complex consensus mechanism is required

7. **Tokenized blockchains**

-are standard blockchains that generate cryptocurrency as a result of a consensus process via mining or initial distribution.
-Bitcoin and Ethereum are examples.

8. **Tokenlessblockchains**

-are designed such that they do not have the basic unit for the transfer of value.
-they are still valuable in situations where there is no need to transfer value between nodes and only the sharing of data among various trusted parties is required.
-similar to full private blockchains, the only difference being that use of tokens is not required.
-can also be thought of as a shared distributed ledger used for storing data.
-benefits are immutability, security, and consensus driven updates but are not used for common blockchain application of value transfer or cryptocurrency.

Consensus

Consensus is a process of agreement between distrusting nodes on the final state of data.
-different algorithms are used.
-an agreement between two nodes (in client-server systems, for example),
-when multiple nodes are participating in a distributed system and they need to agree on a single value, it becomes quite a challenge to achieve consensus. This process of attaining agreement common state or value among multiple nodes despite the failure of some nodes is known as **distributed consensus**.
-provides decentralization of control through an optional process known as **mining.**

Consensus mechanism

A consensus mechanism is a set of steps that are taken by most or all nodes in a blockchain to agree on a proposed state or value.

The following describes the requirements of consensus mechanism:

Agreement: All honest nodes decide on the same value

Termination: All honest nodes terminate execution of the consensus process and eventually reach a decision

Validity: The value agreed upon by all honest nodes must be the same as the initial value proposed by at least one honest node

Fault tolerant: The consensus algorithm should be able to run in the presence of faulty or malicious nodes (Byzantine nodes)

Integrity: This is a requirement that no node can make the decision more than once in a single consensus cycle

Types of consensus mechanisms: All consensus mechanisms are developed to deal with faults in a distributed system and to allow distributed systems to reach a final state of agreement.

Traditional Byzantine Fault Tolerance (BFT)-based: With no compute-intensive operations, such as partial hash inversion (as in BitcoinPoW), this method relies on a simple scheme of nodes that are publisher-signed messages. Eventually, when a certain number of messages are received, then an agreement is reached.

Leader election-based consensus mechanisms: This arrangement requires nodes to compete in a leader-election lottery, and the node that wins proposes a final value. For example, the PoW used in Bitcoin.

Consensus in blockchain

Consensus is a distributed computing concept that has been used in blockchain in order to provide a means of agreeing to a single version of the truth by all peers on the blockchain network.

The following are the consensus algorithms available today

- **Proof of Work (PoW):**
  -relies on proof that adequate computational resources have been spent before proposing a value for acceptance by the network.
  -used in Bitcoin, Litecoin, and other cryptocurrencyblockchains.
  -only algorithm successful against any collusion attacks on a blockchain network, such as the Sybil attack.
- **Proof of Stake (PoS):**
  -idea is that a node or user has an adequate stake in the system
  -user has invested enough in the system so that any malicious attempt by that user would outweigh the benefits of performing such an attack on the network.
  -introduced by Peercoin, used in the Ethereumblockchain version called Serenity.
  -Another concept in PoS is coin age, which is derived from the amount of time and number of coins that have not been spent.
  -the chances of proposing and signing the next block increase with the coin age.
- **Delegated Proof of Stake (DPoS):**
  -an innovation over standard PoS, whereby each node that has a stake in the system can delegate the validation of a transaction to other nodes by voting.
  -used in the BitSharesblockchain.
- **Proof of Elapsed Time (PoET):** Introduced by Intel in 2016, PoET uses a Trusted Execution Environment (TEE) to provide randomness and safety in the leader election process via a

POET works by having each validator randomly wait for a specified time before creating a block, in effect establishing a random order among the validators. The validator that waits the longest time is then allowed to create the next block and broadcast it to the network.

guaranteed wait time. It requires the Intel Software Guard Extensions (SGX) processor to provide the security guarantee for it to be secure.

- **Proof of Deposit (PoD):**
  -nodes that wish to participate in the network have to make a security deposit before they can mine and propose blocks.
  -used in the Tendermintblockchain.
- **Proof of Importance (PoI):**
  -relies on how large a stake a user has in the system and monitors the usage and movement of tokens by the user.
  -used in the NEM coin blockchain.

  In PoI, nodes are assigned an "importance score" based on a number of factors, including the amount of network currency they hold and the number of transactions they have validated.
- **Federated consensus or federated Byzantine consensus**:
  -used in the stellar consensus protocol.
  -Nodes retain a group of publicly-trusted peers and propagate only those transactions that have been validated by the majority of trusted nodes.
- **Reputation-based mechanisms**: As the name suggests, a leader is elected by the reputation it has built over time on the network. It is based on the votes of other members.
- **PBFT**: This mechanism achieves state machine replication, which provides tolerance against Byzantine nodes. Various other protocols including PBFT, PAXOS, RAFT, and Federated Byzantine Agreement (FBA) are also being used.
- **Proof of Activity (PoA):** This scheme is a combination of PoS and PoW, which ensures that a stakeholder is selected in a pseudorandom but uniform fashion. This is more energy-efficient as compared to PoW. It utilizes a new concept called ***Follow the Satoshi***. In this scheme, PoW and PoS are combined together to achieve consensus and good level of security. This scheme is more energy efficient as PoW is used only in the first stage of the mechanism, after the first stage it switches to PoS which consumes negligible energy.
- **Proof of Capacity (PoC):** This scheme uses hard disk space as a resource to mine the blocks. This is different from PoW, where CPU resources are used. In PoC, hard disk space is utilized for mining and as such is also known as hard drive mining. This concept was first introduced in the Burstcoincryptocurrency.
- **Proof of Storage (PoS):** This scheme allows for the outsourcing of storage capacity. This scheme is based on the concept that a particular piece of data is probably stored by a node which serves as a means to participate in the consensus mechanism. Several variations of this scheme have been proposed, such as Proof of Replication, Proof of Data Possession, Proof of Space, and Proof of Space-Time.

CAP theorem and blockchain

CAP theorem, also known as Brewer's theorem. The theory states that any distributed system cannot have consistency, availability, and partition tolerance simultaneously:

Consistency is a property which ensures that all nodes in a distributed system have a single, current, and identical copy of the data.

Availability means that the nodes in the system are up, accessible for use, and are accepting incoming requests and responding with data without any failures as and when required. In other words, data is available at each node and the nodes are responding to requests.

Partition tolerance ensures that if a group of nodes is unable to communicate with other nodes due to network failures, the distributed system continues to operate correctly. This can occur due to network and node failures.

Let's imagine that there is a distributed system with two nodes.Consistency is achieved if both nodes have the same shared state; that is, they have the same up-to-date copy of the data. Availability is achieved if both nodes are up and running and responding with the latest copy of data. Partition tolerance is achieved if communication does not break down between two nodes (either due to network issues, Byzantine faults, and so forth), and they are able to communicate with each other.

Consistency is achieved using consensus algorithms in order to ensure that all nodes have the same copy of the data. This is also called state machine replication. The blockchain is a means for achieving **state machine replication.**There are two types of faults that a node can experience.

Fail-stop fault: This type of fault occurs when a node merely has crashed. Fail-stop faults are the easier ones to deal with of the two fault types.

Byzantine faults: The second type of fault is one where the faulty node exhibits malicious or inconsistent behavior arbitrarily. This type is difficult to handle since it can create confusion due to misleading information. This can be a result of an attack by adversaries, a software bug, or data corruption. State machine replication protocols such as PBFT was developed to address this second type of faults.

Decentralization

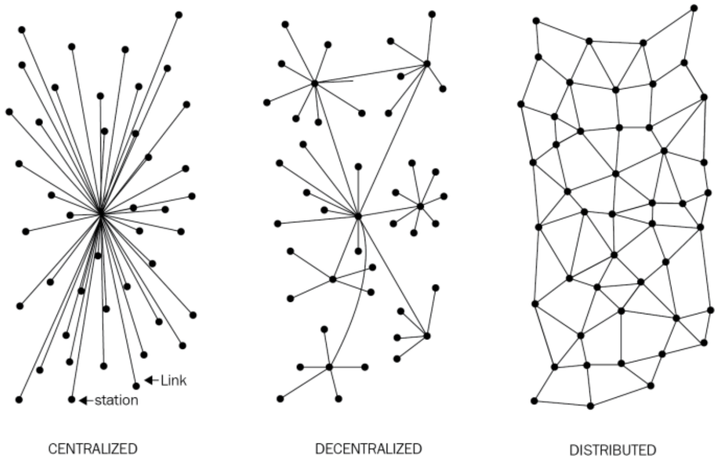
The basic idea of decentralization is to distribute control and authority to the peripheries of an organization instead of one central body being in full control of the organization.

Decentralization using blockchain

Decentralization is a core benefit and service provided by blockchain technology. By design, blockchain is a perfect vehicle for providing a platform that does not need any intermediaries and that can function with many different leaders chosen via consensus mechanisms. This model allows anyone to compete to become the decision-making authority. This competition is governed by a consensus mechanism, and the most commonly used method is known as Proof of Work (PoW).

Information and Communication Technology (ICT) has conventionally been based on a centralized paradigm whereby database or application servers are under the control of a central authority, such as a system administrator. With Bitcoin and the advent of blockchain technology, this model has changed and now anyone to start a decentralized system and operate it with no single point of failure or single trusted authority. It can either be run autonomously or by requiring some human intervention, depending on the type and model of governance used in the decentralized application running on blockchain.

The following diagram shows the different types of systems that currently exist: central, decentralized, and distributed.



Different types of networks/systems

Centralized systems are conventional (client-server) IT systems in which there is a single authority that controls the system, and who is solely in charge of all operations on the system. All users of a centralized system are dependent on a single source of service. The majority of online service providers including Google, Amazon, eBay, Apple's App Store, and others use this conventional model for delivering services.

A **distributed system**, data and computation are spread across multiple nodes in the network. In a distributed system, computation may not happen in parallel and data is replicated across multiple nodes that users view as a single, coherent system. Variations are used to achieve fault tolerance and speed.

The critical difference between a decentralized system and distributed system is that in a distributed system, there still exists a central authority that governs the entire system; whereas, in a decentralized system, no such authority exists.

A **decentralized system** is a type of network where nodes are not dependent on a single master node; instead, control is distributed among many nodes. This is analogous to a model where each department in an organization is in charge of its own database server, thus taking away the power from the central server and distributing it to the subdepartments who manage their own databases.

Methods of decentralization

Two methods can be used to achieve decentralization: disintermediation and competition (Contest-driven decentralization).

Disintermediation

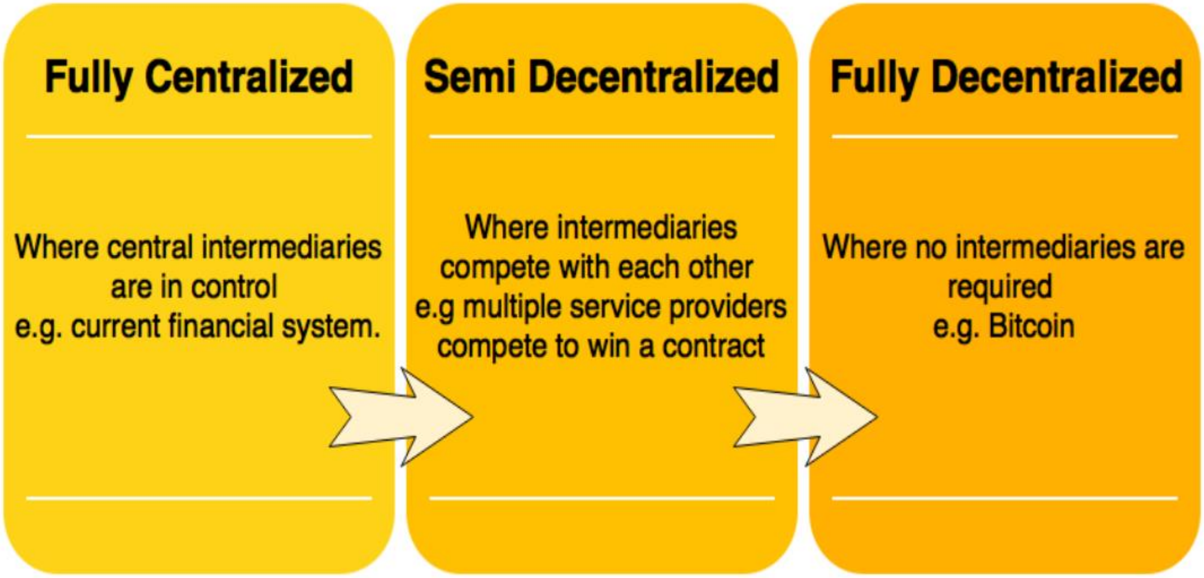
Imagine that you want to send money to a friend in another country. You go to a bank who, for a fee, will transfer your money to the bank in that country. In this case, the bank maintains a central database that is updated, confirming that you have sent the money. With blockchain technology, it is possible to send this money directly to your friend without the need for a bank. All you need is the address of your friend on the blockchain. This way, the intermediary; that is, the bank, is no longer required, and decentralization is achieved by disintermediation.

Contest-driven decentralization

In the method involving **competition**, different service providers compete with each other in order to be selected for the provision of services by the system. This paradigm does not achieve complete decentralization. It ensures that an intermediary or service provider is not monopolizing the service.

This method will not result in full decentralization, but it allows smart contracts to make a free choice. This way, an environment of competition is cultivated among service providers where they compete with each other to become the data provider of choice.

In the following diagram, varying levels of decentralization are shown. On the left-hand side, the conventional approach is shown where a central system is in control; on the right-hand side, complete disintermediation is achieved as intermediaries are entirely removed. Competing intermediaries or service providers are shown in the center. At that level, intermediaries or service providers are selected based on reputation or voting, thus achieving partial decentralization.


Fully Centralized
Where central intermediaries are in control e.g. current financial system.
Semi Decentralized
Where intermediaries compete with each other e.g multiple service providers compete to win a contract
Fully Decentralized
Where no intermediaries are required e.g. Bitcoin

Scale of decentralization

Routes to decentralization /How to decentralize

The following four questions whose answers provide a clear understanding as to how a system can be decentralized:

1. What is being decentralized?
2. What level of decentralization is required?
3. What blockchain is used?
4. What security mechanism is used?

The first question simply asks you to identify what system is being decentralized. This can be any system, such as an identity system or a trading system.

The second question asks you to specify the level of decentralization required. It can be full disintermediation or partial disintermediation.

The third question asks developers to determine which blockchain is suitable for a particular application. It can be Bitcoinblockchain, Ethereumblockchain, or any other blockchain that is fit for the specific application.

Finally, the security mechanism can be atomicity-based, where either the transaction executes in full or does not execute at all. This deterministic approach ensures the integrity of the system. Other mechanisms may include one based on reputation, which allows for varying degrees of trust in a system.

The decentralization framework example

Let's evaluate a money transfer system as an example of an application selected to be decentralized. The answers to these questions are as follows:

1. Money transfer system
2. Disintermediation
3. Bitcoin
4. Atomicity

The responses indicate that the money transfer system can be decentralized by removing the intermediary, implemented on the Bitcoinblockchain, and that a security guarantee will be provided via atomicity.

Atomicity will ensure that transactions execute successfully in full or not execute at all. We have chosen Bitcoinblockchain because it is the longest established blockchain which has stood the test of time.

Blockchain and full ecosystem decentralization

To achieve complete decentralization, the environment around the blockchain also should be decentralized. The blockchain is a distributed ledger that runs on top of conventional systems. These elements include storage, communication, and computation.

Storage: Data can be stored directly in a blockchain and hence achieves decentralization. A disadvantage of this approach is that a blockchain is not suitable for storing large amounts of data by design. It can store simple transactions and some arbitrary data.

An Alternative is to use Distributed Hash Tables (DHTs). DHTs were used initially in peer-to-peer file sharing software, such as BitTorrent, Napster, Kazaa, and Gnutella.

Two primary requirements are high availability and link stability, which means that data should be available when required and network links also should always be accessible. InterPlanetary File System (IPFS) possesses both of these properties. IPFS uses DHT and Merkle Directed Acyclic Graph (DAG) to provide storage and searching functionality, respectively. The incentive mechanism for storing data is based on a protocol known as Filecoin, which pays incentives to nodes that store data using the Bitswap mechanism. The Bitswap mechanism lets nodes keep a simple ledger of bytes sent or bytes received in a one-to-one relationship.

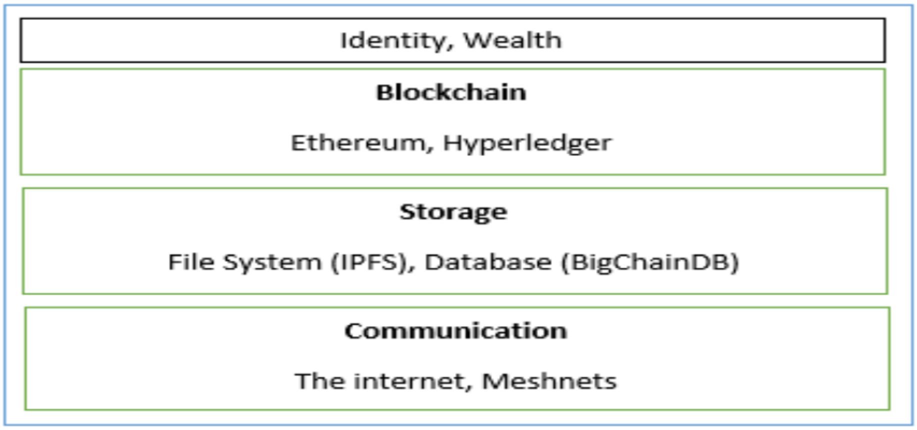
Communication: The internet (the communication layer in blockchain) is considered to be decentralized. There is a need to provide control to individual users in such a way that access to their data is guaranteed and is not dependent on a single third party. Access to the internet (the communication layer) is based on Internet Service Providers (ISPs) who act as a central hub for internet users. If the ISP is shut down for any reason, then no communication is possible with this model.

An alternative is to use mesh networks where nodes can talk directly to each other without a central hub such as an ISP.

Computing power and decentralization: Decentralization of computing or processing power is achieved by a blockchain technology such as Ethereum, where smart contracts with embedded business logic can run on the blockchain network.

The following diagram shows a decentralized ecosystem overview. At the bottom layer, the internet or Meshnets provide a decentralized communication layer. On the next layer up, a storage layer uses technologies such as IPFS and BigchainDB to enable decentralization. Finally, at the next level up, you can see that blockchain serves as a decentralized processing (computation) layer. Blockchain can, in a limited way, provide a storage layer too, but that severely hampers the speed and capacity of the system.

Therefore, other solutions such as IPFS and BigchainDB are more suitable to store large amounts of data in a decentralized way. The Identity, Wealth layers are shown at the top level. Identity on the internet is a vast topic, and systems such as BitAuth and OpenID provide authentication and identification services with varying degrees of decentralization and security assumptions:



Decentralized ecosystem

Smart contracts

A smart contract is a decentralized program. A smart contract usually contains some business logic and a limited amount of data. The business logic is executed if specific criteria are met. Actors or participants in the blockchain use these smart contracts, or they run autonomously on behalf of the network participants.

Decentralized Organizations

DOs are software programs that run on a blockchain and are based on the idea of actual organizations with people and protocols. Once a DO is added to the blockchain in the form of a smart contract or a set of smart contracts, it becomes decentralized and parties interact with each other based on the code defined within the DO software.

Decentralized Autonomous Organizations: a Decentralized Autonomous Organization (DAO) is also a computer program that runs atop a blockchain and embedded within it are governance and business logic rules. DAOs are autonomous, which means that they are fully automated and contain artificially-intelligent logic.
Decentralized Autonomous Corporations: DACs can earn a profit via shares offered to the participants and to whom they can pay dividends. DACs can run a business automatically without human intervention based on the logic programmed into them.
Decentralized Autonomous Societies: DASs are a concept whereby an entire society can function on a blockchain with the help of multiple, complex smart contracts and a combination of DAOs and Decentralized Applications (DApps) running autonomously.
Decentralized Applications (DApps): DAOs, DACs, and DOs are DApps that run on top of a blockchain in a peer-to-peer network. DApps are software programs that can run on their respective blockchains, use an existing established blockchain, or use only the protocols of an existing blockchain. These are called Type I, Type II, and Type III DApps.

Requirements of a Decentralized Application:

- DApp should be fully open source and autonomous, and no single entity should be in control of a majority of its tokens.
- Data and records of operations of the application must be cryptographically secured and stored on a public, decentralized blockchain to avoid any central points of failure.
- A cryptographic token must be used by the application to provide access and rewards to those who contribute value to the applications, for example, miners in Bitcoin.
- The tokens must be generated by the DApp according to a standard cryptographic algorithm. This generation of tokens acts as a proof of the value to contributors (for example, miners)

DApp examples

KYC-Chain: This application provides the facility to manage Know Your Customer (KYC) data securely and conveniently based on smart contracts.
OpenBazaar: This is a decentralized peer-to-peer network that enables commercial activities directly between sellers and buyers instead of relying on a central party, such as eBay and Amazon.
Lazooz: This is the decentralized equivalent of Uber. It allows peer-to-peer ride sharing and users to be incentivized by proof of movement, and they can earn Zooz coins.

Platforms for decentralization

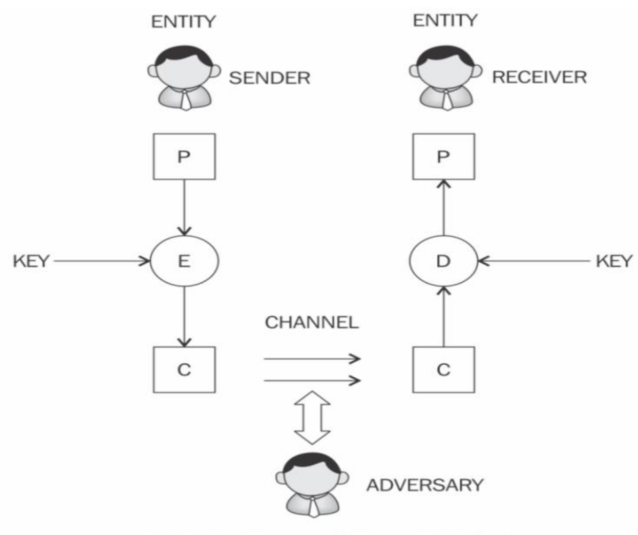
1. **Ethereum**: is the first blockchain to introduce a Turing-complete language and the concept of a virtual machine. With the availability of its Turing-complete language called Solidity, endless possibilities have opened for the development of decentralized applications. This blockchain was first proposed in 2013 by VitalikButerin, and it provides a public blockchain to develop smart contracts and decentralized applications. Currency tokens on Ethereum are called Ethers
2. **MaidSafe:** provides a Secure Access For Everyone (SAFE) network that is made up of unused computing resources, such as storage, processing power, and the data connections of its users. The files on the network are divided into small chunks of data, which are encrypted and distributed randomly throughout the network. This data can only be retrieved by its respective owner. One key innovation of MaidSafe is that duplicate files are automatically rejected on the network, which helps reduce the need for additional computing resources needed to manage the load. It uses Safecoin as a token to incentivize its contributors.
3. **Lisk**: is a blockchain application development and cryptocurrency platform. It allows developers to use JavaScript to build decentralized applications and host them in their respective sidechains. Lisk uses the Delegated Proof of Stake (DPOS) mechanism for consensus whereby 101 nodes can be elected to secure the network and propose blocks. It uses the Node.js and JavaScript backend, while the frontend allows the use of standard technologies, such as CSS3, HTML5, and JavaScript. Lisk uses LSK coin as a currency on the blockchain. Another derivative of Lisk is Rise, which is a Lisk-based decentralized application and digital currency platform. It offers a greater focus on the security of the system.

Cryptography

- Cryptography is the science of making information secure in the presence of adversaries.
- Ciphers are algorithms used to encrypt or decrypt data, so that if intercepted by an adversary, the data is meaningless to them without decryption, which requires a secret key.
- Mathematics
  - Set: A set is a collection of distinct objects, for example, X = {1, 2, 3, 4, 5}
  - Group: A group is a commutative set with one operation that combines two elements of the set. The group operation is closed and associated with a defined identity element. Each element in the set has an inverse.
  - Field : A field is a set that contains both additive and multiplicative groups
  - A finite field A finite field is one with a finite set of elements. Prime finite fields are used in Elliptic Curve Cryptography (ECC) to construct discrete logarithm problems.
  - Order: The order is the number of elements in a field. It is also known as the cardinality of the field.

- An abelian group: An abelian group is formed when the operation on the elements of a set is commutative.
- Prime fields: A prime field is a finite one with a prime number of elements. Addition and multiplication operations are performed modulo p, that is, prime.
- Ring: If more than one operation can be defined over an abelian group, that group becomes a ring.
- A cyclic group: A cyclic group is a type of group that can be generated by a single element called the group generator.
- Modular arithmetic: Numbers in modular arithmetic wrap around when they reach a certain fixed number. This fixed number is a positive number called modulus, and all operations are performed concerning this fixed number

Cryptography : A generic cryptography model is shown in the following diagram:



A model of the generic encryption and decryption model

Cryptography in blockchain refers to the use of mathematical algorithms and protocols to secure the transactions and data stored on a blockchain network. The main purpose of cryptography in blockchain is to ensure that the transactions are secure and tamper-proof, and that the privacy of users is maintained.

In a blockchain network, cryptographic techniques are used to create secure digital signatures, which are used to verify the identity of the users who are conducting transactions. These digital signatures are created using public-key cryptography, which involves the use of a pair of public and private keys. The private key is used to sign a transaction, while the public key is used to verify the signature.

Cryptography is also used to encrypt the data stored on the blockchain, making it difficult for unauthorized users to access or modify the information. This helps to ensure the security and privacy of the data stored on the blockchain, and is essential for maintaining the integrity and trust in the system.

P, E, C, and D represent plaintext, encryption, ciphertext, and decryption, respectively.
Entity: Either a person or system that sends, receives, or performs operations on data
Sender: This is an entity that transmits the data
Receiver: This is an entity that takes delivery of the data
Adversary: This is an entity that tries to circumvent the security service
Key: A key is data that is used to encrypt or decrypt other data
Channel: Channel provides a medium of communication between entities

Cryptography Services
Confidentiality: assurance that information is only available to authorized entities.
Integrity: assurance that information is modifiable only by authorized entities.
Authentication: assurance about the identity of an entity or the validity of a message. There are two types of authentication mechanisms, namely entity authentication and data origin authentication.

- **Entity authentication**: assurance that an entity is currently involved and active in a communication session. Traditionally, users are issued a username and password that is used to gain access to the various platforms with which they are working. This practice is known as single-factor authentication, as there is only one factor involved, namely, something you know, that is, the password and username. This type of authentication is not very secure for a variety of reasons, for example, password leakage; therefore, additional factors are now commonly used to provide better security. The use of additional techniques for user identification is known as multifactor authentication (or two-factor authentication if only two methods are used)

Various authentication factors are

The first factor is something you have, such as a hardware token or a smart card. This mechanism protects the user by requiring two factors of authentication. A user who has access to the hardware token and knows the login credentials will be able to access the system.

The second factor is something you are, which uses biometric features to identify the user. With this method, a user's fingerprint, retina, iris, or hand geometry is used to provide an additional factor for authentication

- **Data origin authentication** or message authentication :assurance that the source of the information is indeed verified. Data origin authentication guarantees data integrity because if a source is corroborated, then the data must not have been altered. Various methods, such as Message Authentication Codes (MACs) and digital signatures are most commonly used.

Non-repudiation: assurance that an entity cannot deny a previous commitment or action by providing incontrovertible evidence. It is a security service that offers definitive proof that a particular activity has occurred. The non-repudiation protocol usually runs in a communication network.There are two communications models that can be used to transfer messages from originator A to recipient B:

A message is sent directly from originator A to recipient B.

A message is sent to a delivery agent from originator A, which then delivers the message to recipient B.

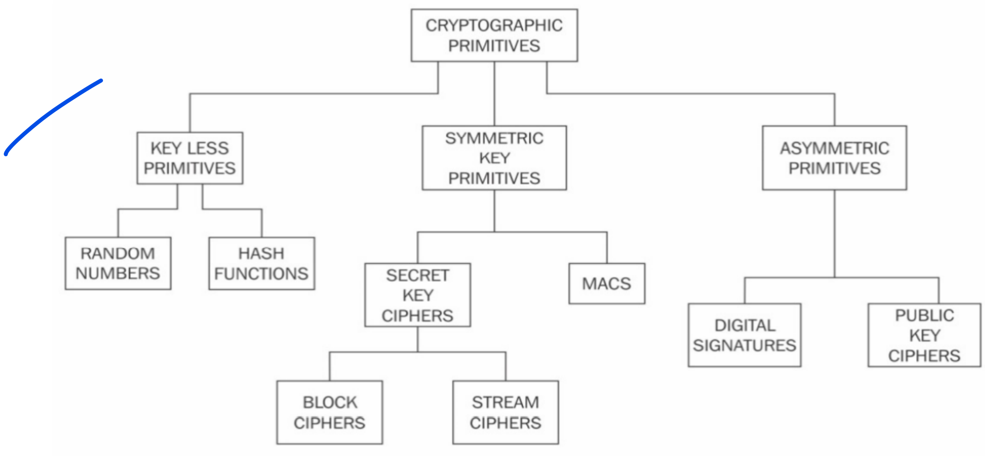
The primary requirements of a non-repudiation protocol are fairness, effectiveness, and timeliness.

Accountability:assurance which states that actions affecting security can be traced back to the responsible party. This is usually provided by logging and audit mechanisms in systems where a detailed audit is required due to the nature of the business, for example, in electronic trading systems.

Cryptographic Primitives

Cryptographic primitives are the basic building blocks of a security protocol or system. A security protocol is a set of steps taken to achieve the required security goals by utilizing appropriate security mechanisms. Various types of security protocols are in use, such as authentication protocols, non-repudiation protocols, and key management protocols.

The taxonomy of cryptographic primitives can be visualized as shown here:



Cryptographic primitives

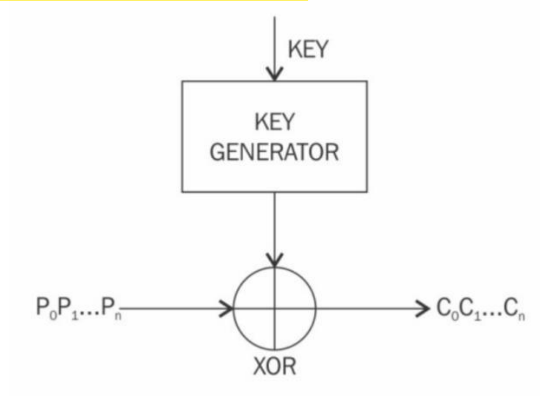
Symmetric cryptography or shared key cryptography :refers to a type of cryptography where the key that is used to encrypt the data is the same one that is used for decrypting the data. The key must be established or agreed upon before the data exchange occurs between the communicating parties. This is the reason it is also called secret key cryptography.

There are two types of symmetric ciphers: stream ciphers and block ciphers. Data Encryption Standard (DES) and Advanced Encryption Standard (AES) are typical examples of block ciphers, whereas RC4 and A5 are commonly used stream ciphers.

1. **Stream ciphers**: are encryption algorithms that apply encryption algorithms on a bit-by-bit basis (one bit at a time) to plaintext using a keystream.

2 types :
Synchronous stream ciphers are those where the keystream is dependent only on the key
Asynchronous stream ciphers have a keystream that is also dependent on the encrypted data

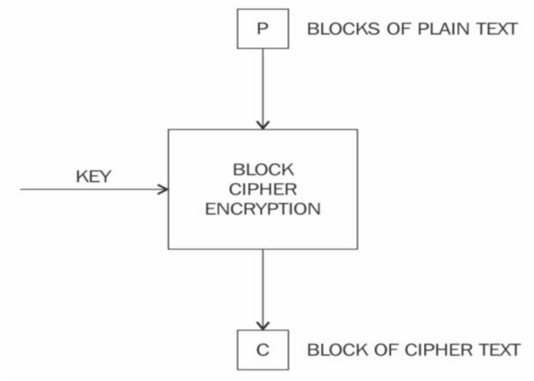
In stream ciphers, encryption and decryption are the same function because they are simple modulo-2 additions or XOR operations. The fundamental requirement in stream ciphers is the security and randomness of keystreams.



Operation of a stream cipher

2. **Block ciphers**: are encryption algorithms that break up the text to be encrypted (plaintext) into blocks of a fixed length and apply the encryption block-by-block. Block ciphers, such as AES (Rijndael) have been built using a combination of substitution and permutation called a Substitution-Permutation Network (SPN).
Block ciphers are built using a design strategy known as a Feistel cipher, based on the Feistel network, developed by Horst Feistel. This structure is based on the idea of combining multiple rounds of repeated operations to achieve desirable cryptographic properties known as confusion and difusion. Feistel networks operate by dividing data into two blocks (left and right) and processing these blocks via keyed round functions in iterations to provide sufficient pseudorandom permutation.
A in plaintext is replaced by X in encrypted text. Substitution is performed using lookup tables called S-boxes. The diffusion property spreads the plaintext statistically over the encrypted data. This ensures that even if a single bit is changed in the input text, it results in changing at least half (on average) of the bits in the ciphertext. Confusion is required to make finding the encryption key very difficult. This is achieved by transposition or permutation.

Advantage of Feistelcipher : encryption and decryption operations are almost identical and only require a reversal of the encryption process to achieve decryption. DES is a prime example of Feistel-based ciphers.



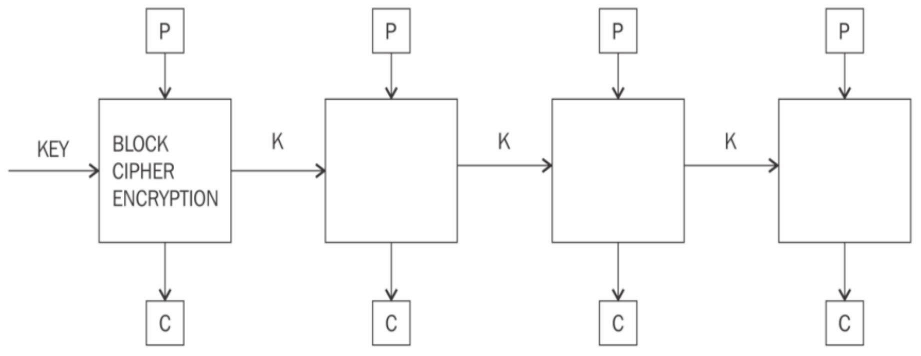
Simplified operation of a block cipher

Modes of operation for block ciphers
Electronic Code Book (ECB)
Cipher Block Chaining (CBC)
Output Feedback (OFB) mode
Counter (CTR) mode.

2.1 Block encryption mode:

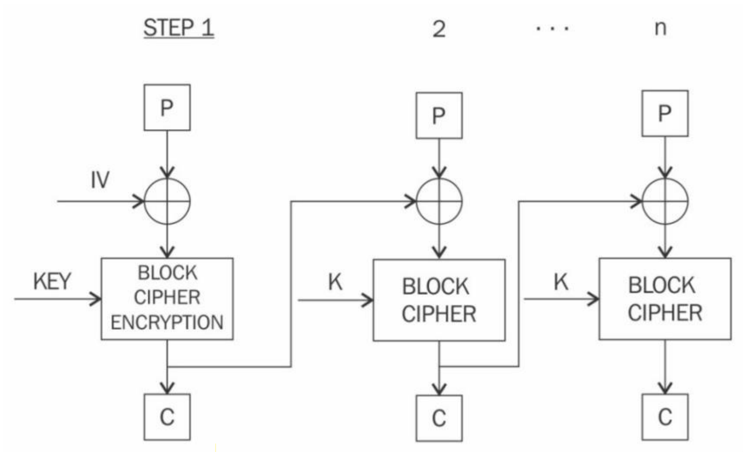
In block encryption mode, the plaintext is divided into blocks of fixed length depending on the type of cipher used.

2.1.1 Electronic Code Book: is a basic mode of operation in which the encrypted data is produced as a result of applying the encryption algorithm one-by-one to each block of plaintext.



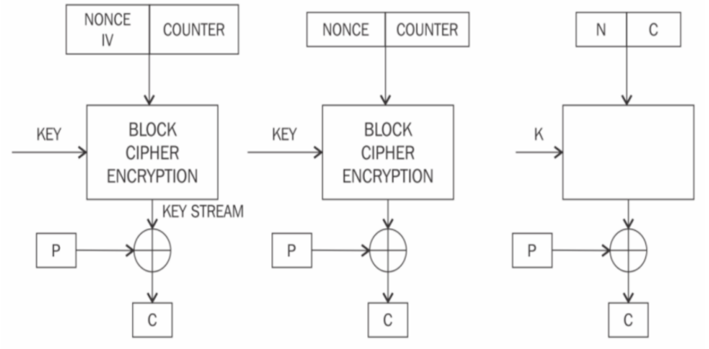
Electronic Code Book mode for block ciphers

2.1.2 Cipher Block Chaining: In Cipher Block Chaining (CBC) mode, each block of plaintext is XOR'd with the previously-encrypted block. CBC mode uses the Initialization Vector (IV) to encrypt the first block.



Cipher block chaining mode

2.1.3 Counter mode The Counter (CTR) mode effectively uses a block cipher as a stream cipher. In this case, a unique nonce is supplied that is concatenated with the counter value to produce a keystream.



Counter mode

2.1.4 Keystream generation mode Inkeystream generation mode, the encryption function generates a keystream that is then XOR'd with the plaintext stream to achieve encryption.

2.1.5 Message authentication mode In message authentication mode, a Message Authentication Code (MAC) results from an encryption function. The MAC is a cryptographic checksum that provides an integrity service. Method to generate a MAC using block ciphers is CBC-MAC, where a part of the last block of the chain is used as a MAC. For example, a MAC can be used to ensure that if a message is modified by an unauthorized entity.

2.1.6 Cryptographic hash mode Hash functions are primarily used to compress a message to a fixed-length digest. In cryptographic hash mode, block ciphers are used as a compression function to produce a hash of plaintext.

Data Encryption Standard (DES)

-introduced by the U.S. National Institute of Standards and Technology (NIST) as a standard algorithm for encryption, during the 1980s and 1990s.
-not resistant to brute force attacks
-In 1998, Electronic Frontier Foundation (EFF) broke DES using a special-purpose machine called EFF DES cracker (or Deep Crack).
-DES uses a key of only 56 bits

-Triple DES (3DES), uses 168-bit key by means of three 56-bit keys and the same number of executions of the DES algorithm, thus making brute force attacks almost impossible.

Advanced Encryption Standard(AES)

-In 2001, after an open competition, an encryption algorithm named Rijndael invented by cryptographers Joan Daemen and Vincent Rijmen was standardized as Advanced Encryption Standard (AES)

-original version of Rijndael permits different key and block sizes of 128-bit, 192-bit, and 256-bits. In the AES standard, however, only a 128-bit block size is allowed

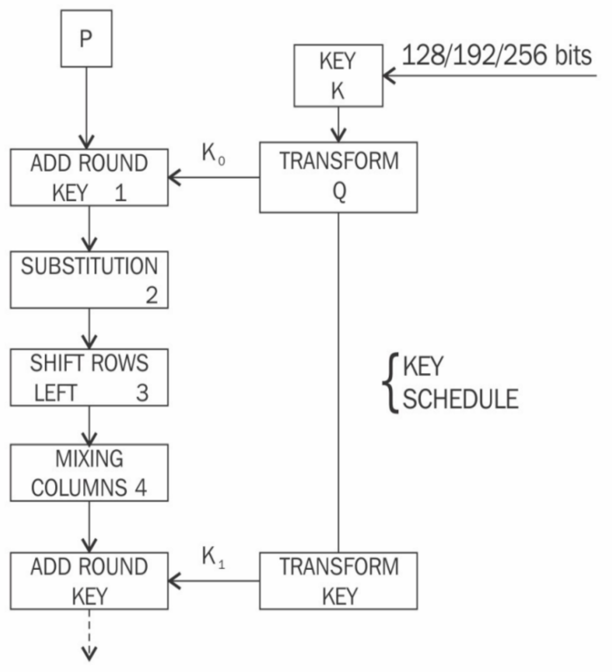
-During AES algorithm processing, a 4 x 4 array of bytes known as the state is modified using multiple rounds. Full encryption requires 10 to 14 rounds, depending on the size of the key.

-

Key Size	Number of rounds required
128-bit	**10 rounds**
192-bit	**12 rounds**
256-bit	**14 rounds**

Four operations are performed in four stages to encrypt the input:

1. In the **AddRoundKey** step, the state array is XOR'd with a subkey, which is derived from the master key
2. **SubBytes** is the substitution step where a lookup table (S-box) is used to replace all bytes of the state array
3. The **ShiftRows** step is used to shift each row to the left, except for the first one, in the state array to the left in a cyclic and incremental manner
4. Finally, all bytes are mixed in the **MixColumns** step in a linear fashion, column-wise



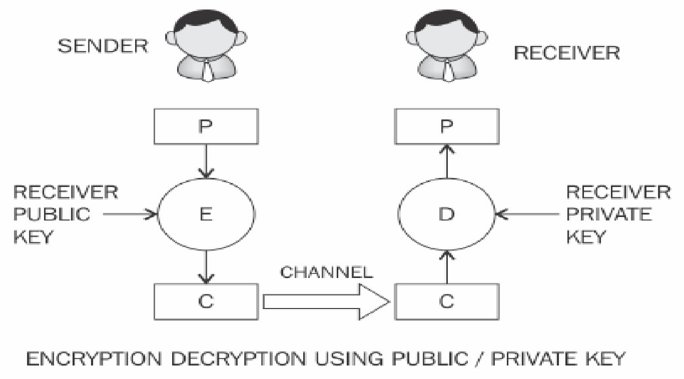
AES block diagram, showing the first round of AES encryption. In the last round, the mixing step is not performed

Asymmetric cryptography

Public Key Cryptography

Asymmetric cryptography

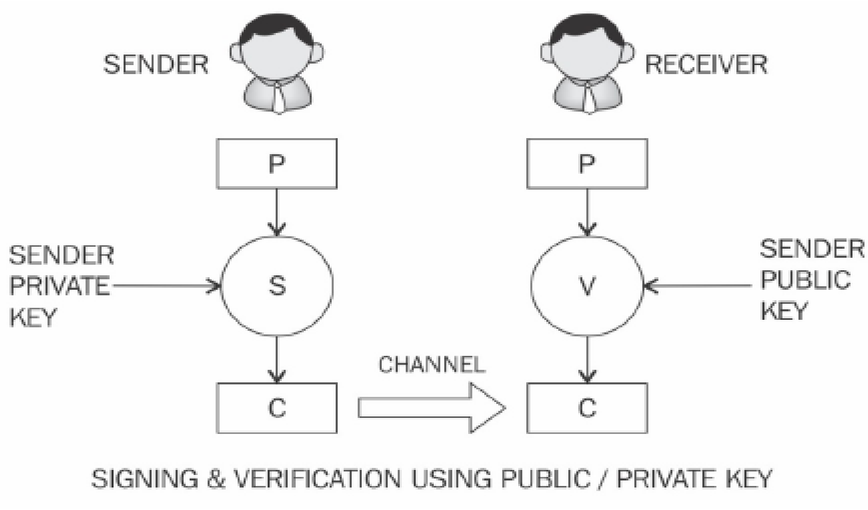
Asymmetric cryptography refers to a type of cryptography where the key that is used to encrypt the data is different from the key that is used to decrypt the data. This is also known as public key cryptography.
- uses both public and private keys to encrypt and decrypt data, respectively.
- examples are RSA, DSA, and ElGammal.



Encryption/decryption using public/private keys

-sender encrypts data P using the recipient's public key and encryption function E and producing an output encrypted data C
-C is then transmitted over the network to the receiver.
- receiver decrypts using the receiver's private key by giving C into function D giving output plaintext P
- private key remains on the receiver's side, and no need to share keys in order to perform encryption and decryption.

The following diagram shows how the receiver uses public key cryptography to verify the integrity of the received message.



Model of a public-key cryptography signature scheme

-sender signs the data using their private key and transmits the message across to the receiver.
-Once the message is received, it is verified for integrity by the sender's public key.
-sender digitally signs the plaintext P with his private key using signing function S and produces data C sent to the receiver
-receiver verifies C using sender public key and function V to ensure the message has come from the sender.
-Security mechanisms offered by public key cryptosystems include key establishment, digital signatures, identification, encryption, and decryption.

Key establishment mechanisms are refers to the setting up of keys over an insecure channel.
Non-repudiation services is a property provided using digital signatures.
It is important to authenticate a user and to identify the entity involved in a transaction.
-achieved by a combination of digital signatures and challenge-response protocols.
-encryption mechanism to provide confidentiality can also be obtained using public key cryptosystems, such as RSA, ECC, and ElGammal.

Public key algorithms are slower in terms of computation than symmetric key algorithms.
Therefore, they are not commonly used in the encryption of large files or the actual data that requires encryption. They are usually used to exchange keys for symmetric algorithm.
Once the keys are established securely, symmetric key algorithms can be used to encrypt the data.

Public key cryptography algorithms are based on mathematical functions discussed below.

1. **Integer factorization schemes** are based on the fact that large integers are very hard to factor.

RSA is the prime example of this type of algorithm.

2. **Discrete logarithm scheme** is based on a problem in modular arithmetic. It is easy to calculate the result of modulo function, but it is computationally impractical to find the exponent of the generator.
3. **Elliptic curves algorithm** is based on the discrete logarithm problem. An elliptic curve is an algebraic cubic curve over a field, defined using the equation

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

a and b are integers whose values are elements of the field on which the elliptic curve is defined. Elliptic curves can be defined over real numbers, rational numbers, complex numbers, or finite fields.
-Elliptic Curve Digital Signature Algorithm (ECDSA) and the Elliptic Curve Diffie-Hellman (ECDH) key exchange are examples

Public and private keys

A **private key** is a randomly generated number that is kept secret and held privately by its users.
-need to be protected and no unauthorized access should be granted to that key
-can be of various lengths depending on the type and class of algorithms used.
-For example, in RSA,a key of 1024-bits or 2048-bits is used.

A **public key** is freely available and published by the private key owner.Anyone who want to send the publisher of the public key an encrypted message can do so by encrypting the message using the published public key and sending it to the holder of the private key.
No one else is able to decrypt the message because the corresponding private key is held securely by the intended recipient.

Once the public key encrypted message is received, the recipient can decrypt the message using the private key. These include authenticity and identification of the publisher of the public keys.

RSA is the first implementation of public key cryptography whereas ECC is used extensively in blockchain technology.

RSA

RSA was invented in 1977 by Ron Rivest, Adi Shamir, and Leonard Adelman, hence the name Rivest-Shamir-Adleman (RSA).
-is a public key cryptography is based on the integer factorization problem, where the multiplication of two large prime numbers is easy, but it is difficult to factor it back.
An RSA key pair is generated by performing the following steps:

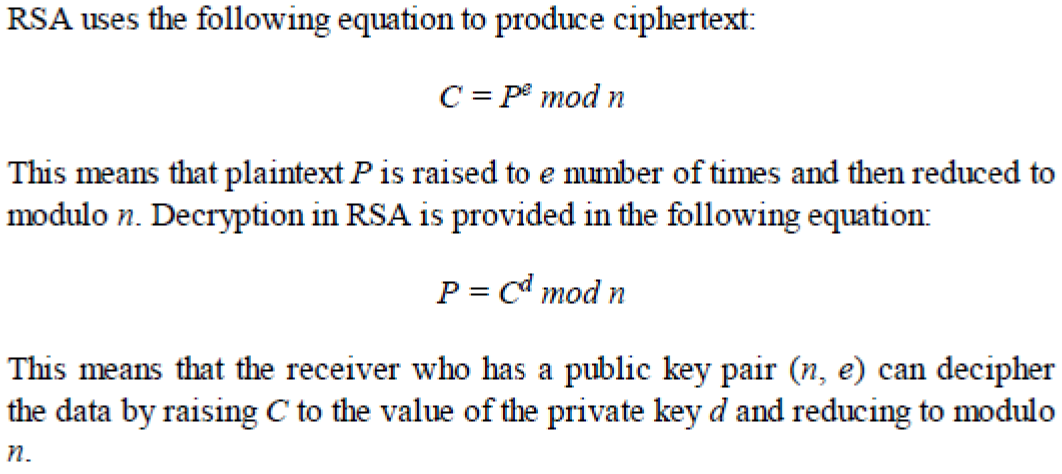
1. Modulus generation:
   - Select p and q, which are very large prime numbers
   - Multiply p and q, n=p.q to generate modulus n
2. Generate co-prime:
   - Assume a number called e.
   - e should satisfy a certain condition; it should be greater than 1 and less than (p-1) (q-1). e must be a number such that no number other than 1 can divide e and (p-1) (q-1). e is the co-prime of (p-1) (q-1).
3. Generate the public key:

Modulus generated in step 1 and co-prime e generated in step 2 is a pair together that is a public key. This part is the public part that can be shared with anyone; however, p and q need to be kept secret.

4. Generate the private key:
   Private key, called d is calculated from p, q, and e. Private key is the inverse of e modulo (p-1) (q-1).

$$ed = 1 \bmod (p-1)\ (q-1)$$

Encryption and decryption using RSA



RSA uses the following equation to produce ciphertext:

$$C = P^e \bmod n$$

This means that plaintext P is raised to e number of times and then reduced to modulo n. Decryption in RSA is provided in the following equation:

$$P = C^d \bmod n$$

This means that the receiver who has a public key pair (n, e) can decipher the data by raising C to the value of the private key d and reducing to modulo n.

Elliptic Curve Cryptography

Elliptic Curve Cryptography (ECC) is based on the discrete logarithm problem founded upon elliptic curves over finite fields (Galois fields).
-main benefit of ECC is it requires a smaller key size while providing the same level of security as RSA.
-Two notable schemes that originate from ECC are ECCH for key exchange and DCC□A for digital signatures.
-ECC can also be used for encryption
-ECC is used for key exchange and digital signatures commonly.
-ECC needs less space to operate, it is becoming very popular
-same level of security can be achieved with ECC only using 256-bit operands as compared to 3072-bits in RSA.

—An elliptic curve is defined in the following equation:

$$y^2 = x^3 + Ax + B \; mod \; P$$

A and B belong to a finite field Zp or Fp (prime finite field) along with a special value called the point of infinity.

-The point of infinity (∞) is used to provide identity operations for points on the curve