Cloud

Characteristics

i) on demand : ~~Scalable Resources on demand~~
Self Service NO human Admin needed

ii) Broad Network : No geological Boundry

iii) Rapid Elasticity : Scalable Resources

iv) Resource pooling : Multiple Resources are shared with your multiple services

v) Mesured Service : pay what you use.

Service Models

i) IAAS : Infrastructure → HaaS : Hardware
ii) PAAS : platform
iii) SAAS : Software
other - - - - - - - - - -
i) XaaS : Everything
ii) CaaS : Communication
iii) NaaS : Network
iv) DBaaS : Database
v) SECaaS : Security
vi) HaaS : Healthcare
vii) TaaS : Transport
viii) BMaas : Bare Metal

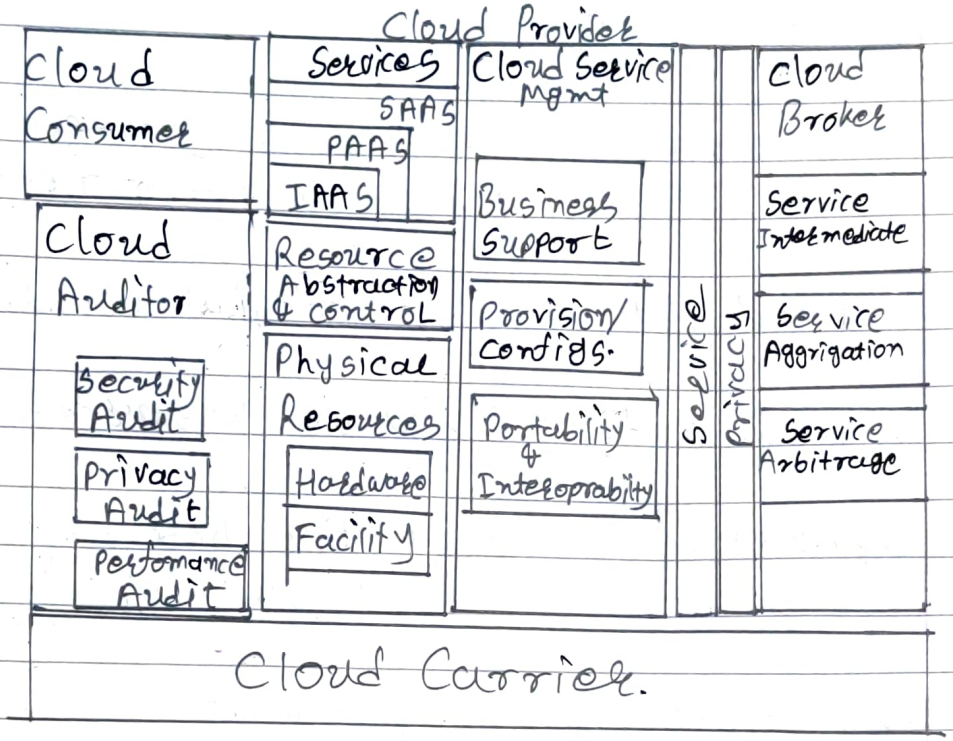
HaaS :
- Compute
- Storage
- Network
- Load balancers

PAAS Types
- AIpaaS : AI
- iPaaS : Intigration
- cPaas : communication
- mPaaS : Mobile

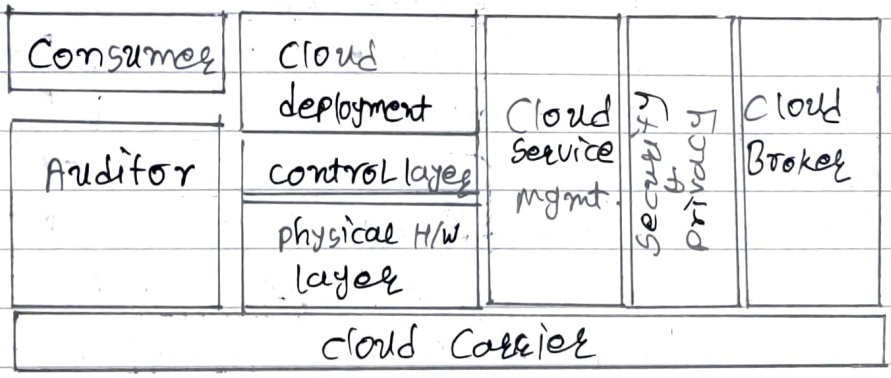
Deployment Model

i) Public
ii) Private
iii) Hybrid
iv) Community
v) Multi Cloud

⊛ Cloud Architecture.



Easy version



⊛ Stakeholders

i) Consumer : Customer
ii) Provider : AWS, Google, Linode
iii) Auditor : Assessment of cloud
iv) Broker : Manages Relation consumer to provider
v) Carrier : Provides connectivity & transport

Page No.:

Broker

- Service Intermediation
- Service Aggrigation
- Service Arbitrage.

Carrier

- provides Network, telecomm., Network Access devices

Auditor

- performance
- Security.
- Controls
- Privacy

Cloud Advantage vs Disadvantage

Advantage	Disadvantage
• Bkp & Restore	• Net Connection
• collaboration	• vendor lockin
• 24/7 available	• limited control
• Low maintanance	• Security
• pay as you go	• Loss of data
• unlimited storage.	

Challanges in cloud

i) DoS
ii) Security & privacy
iii) Lack of standards
iv) Reliability
v) Semi-Trusted Servers.

Cloud Security

i) Vulnerabilities
   a) Data threats
   b) API Vulnerabilities
   c) Malicious Insiders
   d) Shared technology vulnerability
   e) Weak cryptography

ii) Attack vectors
   a) Vulnerabilities / 0-day
   b) Weak credentials
   c) Weak Access control
   d) Malicious Insiders
   e) Bruteforce
   f) Phishing

iii) Cloud Threats
   a) Malware injection Attacks
      - XSS, SQL
   b) Abuse of cloud service
   c) DDoS
   d) Side channel attack
   e) Wrapping attack
   f) man in the cloud attack
   g) Insider
   h) hijacking Account/service
      - Spyware, cookie jacking.
   i) APT
   j) 0-day
   k) Insecure API

Mechanisms for Accountability In Cloud.

i) Preventive controls
   a) Risk Analysis & decision support tools
   b) policy enforcing
   c) Data obfuscation
   d) Identity Management

ii) Detective Control
   a) Intrusion detection
   b) Transaction logs
   c) Lang. framework for security properties
   d) verification tools

iii) Corrective Controls
   a) Incident mgmt plan.
   b) dispute Resolution methods
   c) other Remidiation

SLA (Service Level Agreement)

Service Provider ←—— SLA ——→ Consumer

SLA defines the services standards

Types of SLA

i) Customer
ii) Internal
iii) Multilevel

⑧ key Components of SLA

i) Agreement overview
ii) Discription of Service
iii) Exclusions
iv) Service performance
v) Redressing (compansation if failed to deliver on terms of SLA.
vi) Stakeholders
vii) Security
viii) Risk management & Disaster Recovery
ix) Service tracking & Reporting
x) Reviews
xi) termination of process
xii) Signatures.

⑨ Encryption.

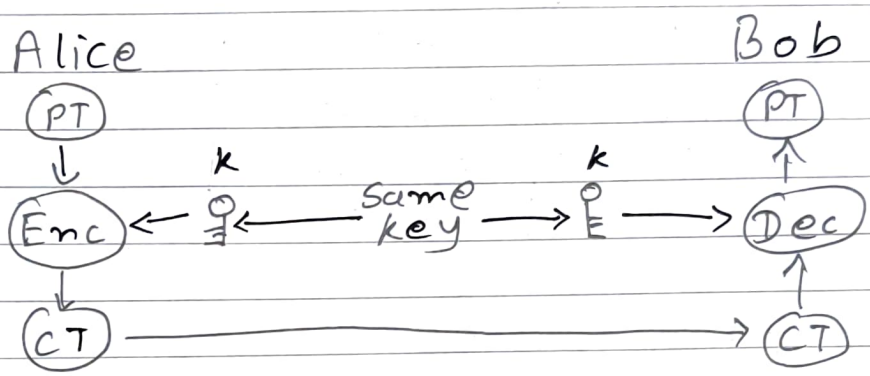
i) Types of Encryption operation
→ Substitution : Re-mapping characters
→ Transposition : Re-arranging cha
→ Product : combination of both

ii) No. of keys
→ 1 key (private/symmetric)
→ 2 key (public/symmetric)

iii) Processing plaintext
→ block
→ stream.

⋆ No. of keys

i) Symmetric



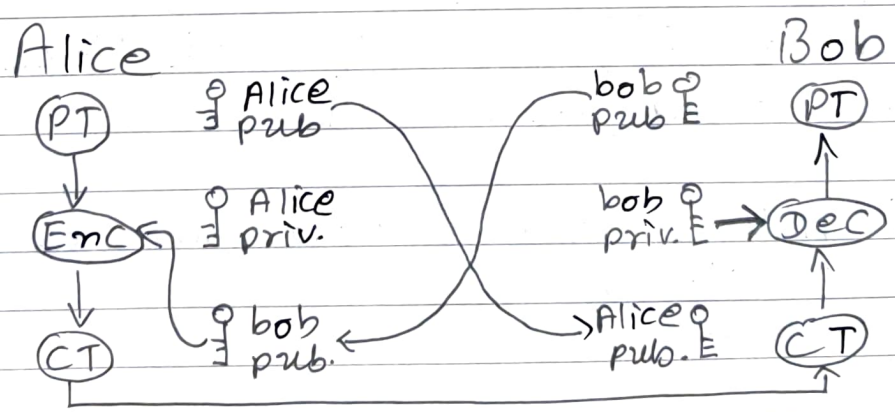
PT = plain text , CT = cipher text , k = key

→ Enc $C = E_k(P)$

→ Dec $P = D_k(C)$

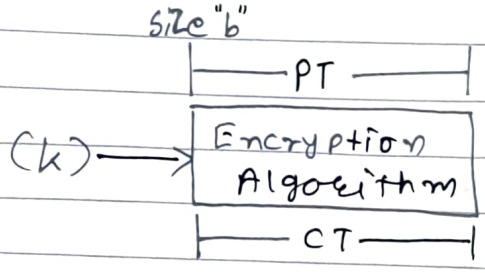
$$D_k(E_k(x)) = E_k(D_k(x)) = x$$

ii) Asymmetric

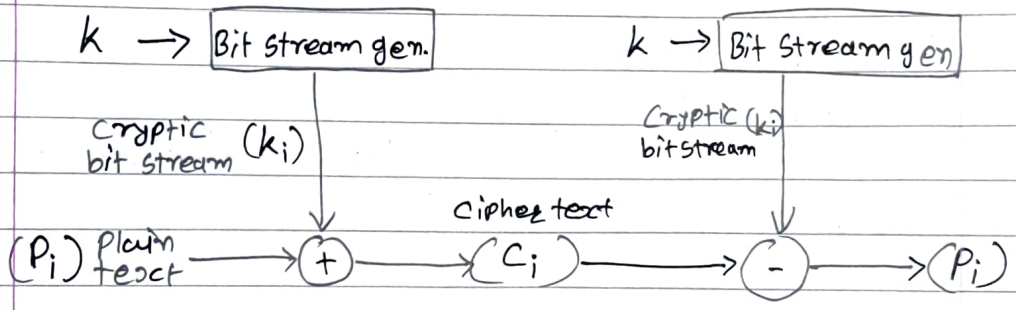


⭐ Processing of plain text

→ i) Block Cipher

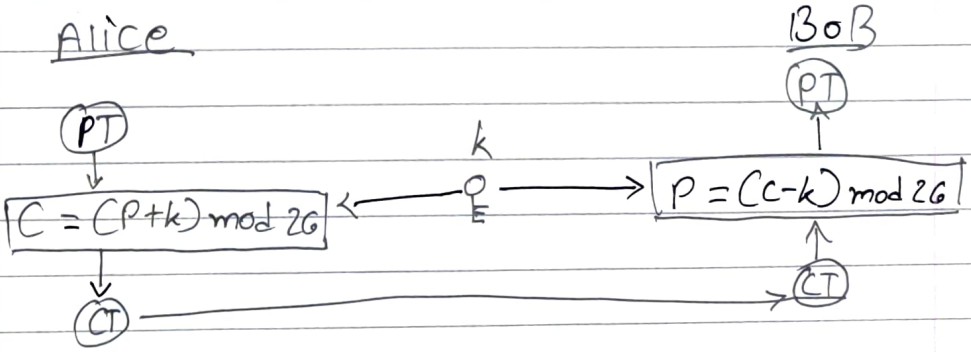


→ ii) Stream cipher



$$PT = PT_1, PT_2 \ldots PT_n$$
$$CT = CT_1, CT_2 \ldots CT_n$$
$$K = k_1, k_2 \ldots k_n$$
$$CT = E_{k_1}(PT_1) \ldots CT_n = E_{k_n}(PT_n)$$

→ iii) Additive Cipher (Monoalphabetic cipher)
or Shift cipher
or Ceaser Cipher.

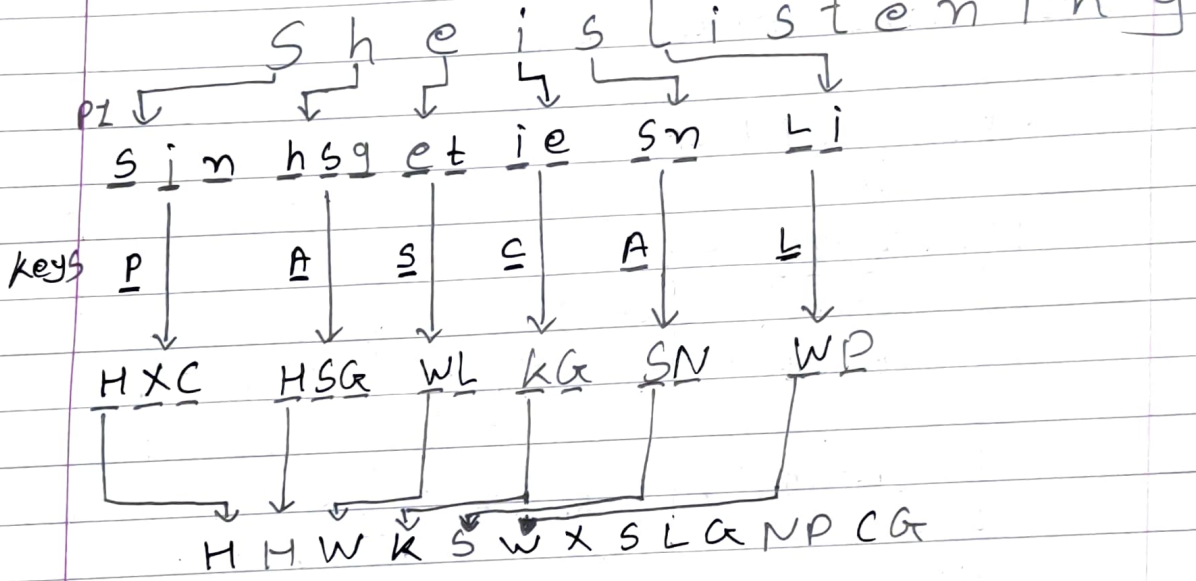


Encryption is adding key value(k) To plaintext that returns a value from Z_{26} to replace

Ex = HELLO, key = 13 → H→U L→Y E→R O→B, cipher = URYYB

iv) Vigenere Cipher (polyalphabetic cipher)

Ex: "She is listening", key = "PASCAL"



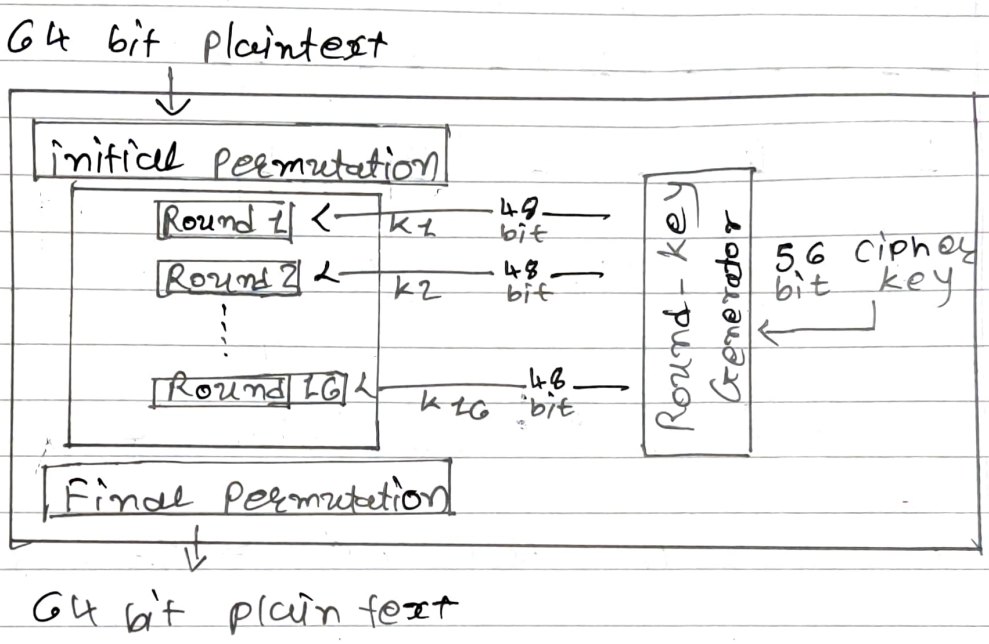
→ key "PASCAL" keystream = (15, 0, 18, 2, 0, 11)

plaintext →	s	h	e	i	s	L	i	s	t	e	n	i	n	g
P's val. →	18	07	04	08	18	11	08	18	19	04	13	08	13	06
key stream →	15	00	18	02	00	11	15	00	18	02	00	11	15	00
C's val. →	07	07	22	10	18	22	23	18	11	06	13	19	02	06
Ciphertext →	H	H	W	K	S	W	X	S	L	G	N	T	C	G

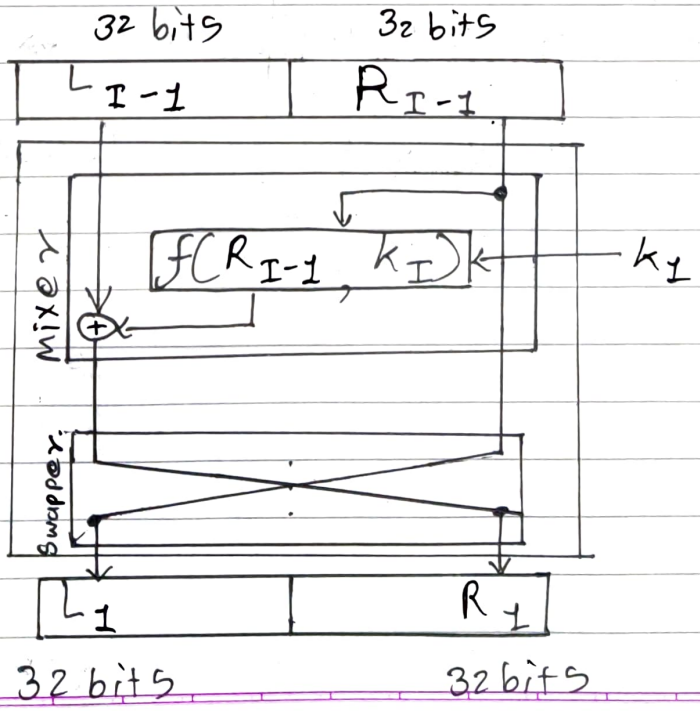
⊛ Substitution or transposition.

→ to resist Exhaustive - Search Attack.
both Substitution & transpotition is needed

→ DES Enc. Structure

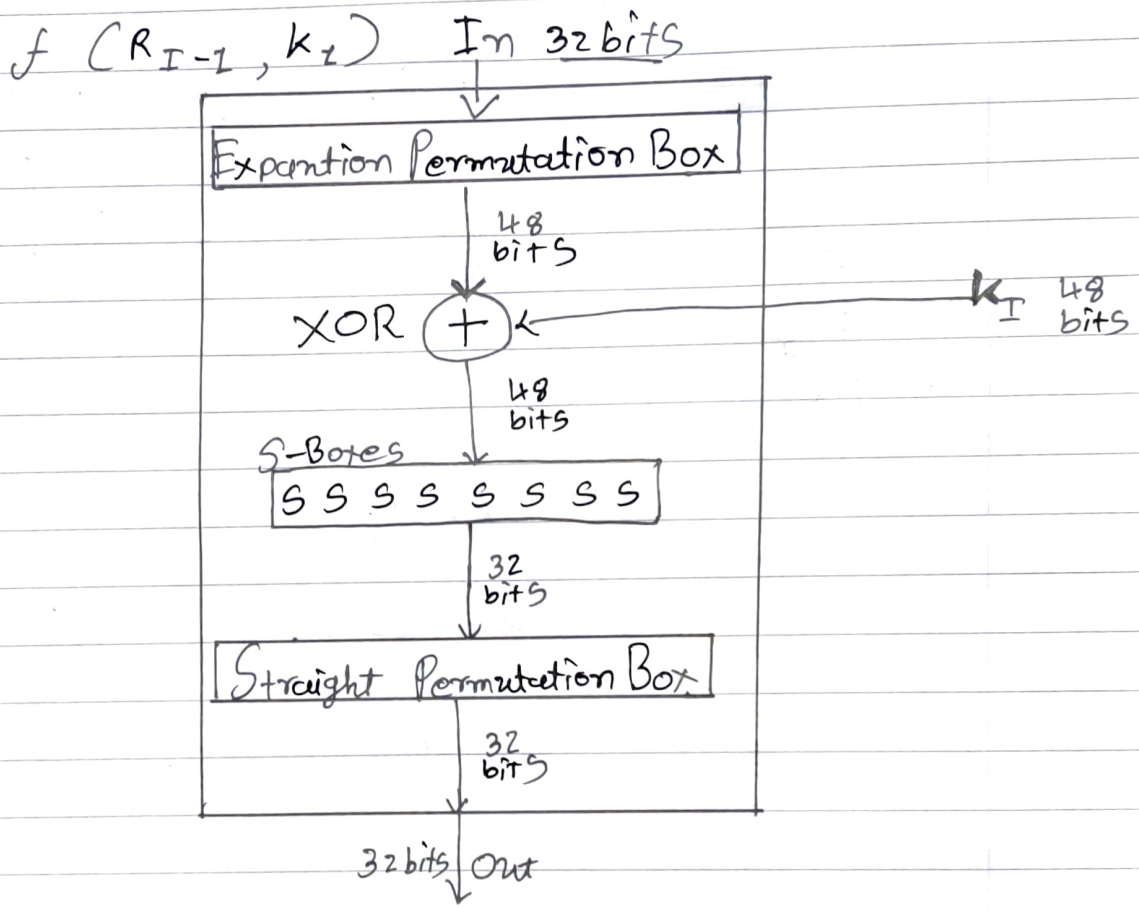


→ Each Round is Feistel Cipher of 16 Rounds



DES Function -

→ applied to 48-bit key to rightmost 32 bits to produce 32 bit output

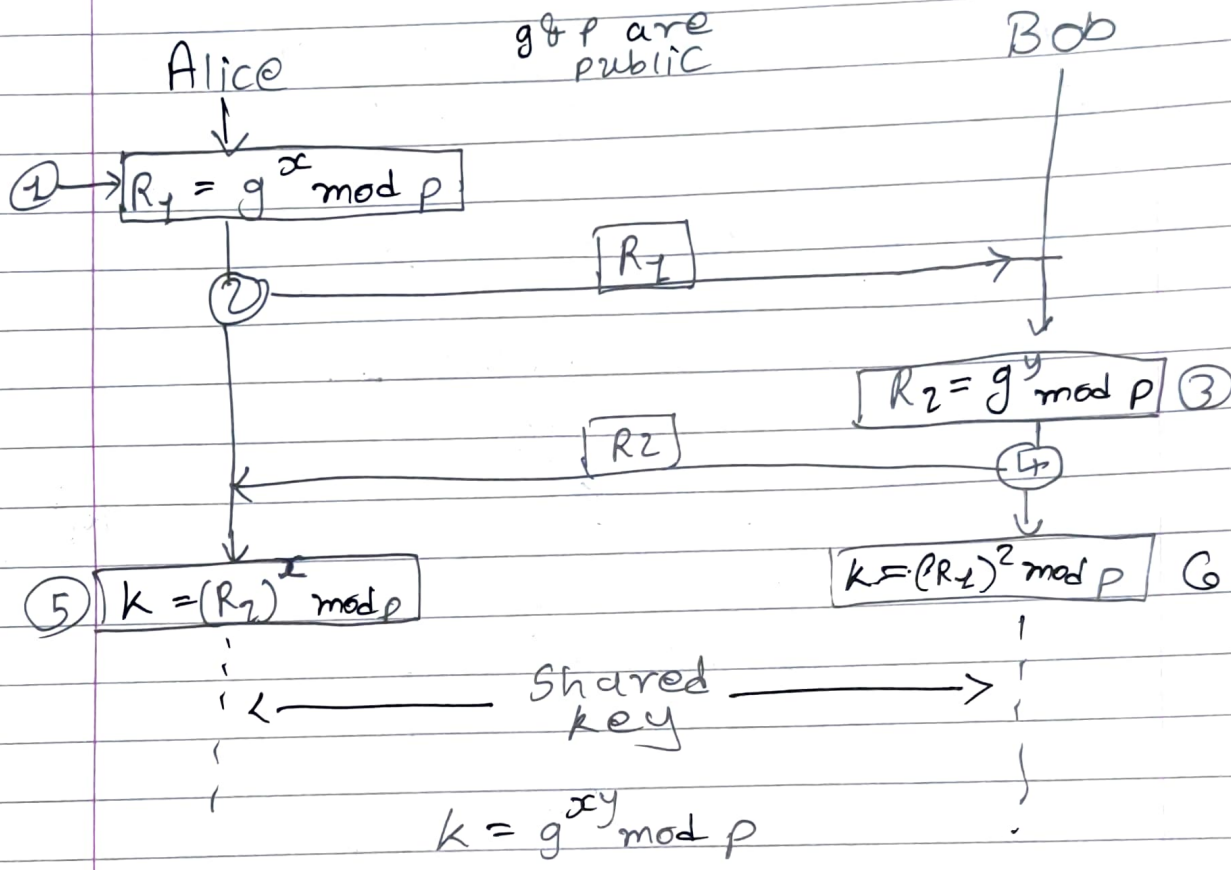


Symmetric key Agreement

→ Session key to Avoid using key Distribution Center

i) Diffie-Hellman key Agreement
ii) Station-to-Station key Agreement

i) Diffie-Hellman Method



Cloud Storage. (from 3rd party vendor, Accessed via API's, scalable on demand)

benifits

i) total cost of ownership.
ii) Fast deployment
iii) Informative management

Requirements

i) durability : Stored Redundently Across multiple facilities to ensure no data loss
ii) Availability:
iii) Security : Encrypted in both rest & in transit, permission & Access control Applied

⊛ Types of cloud Storage.

i) object Storage: stored as "objects" in scalable buckets for unstructured big data & Archiving
ii) File Storage: stored as files in hierarchically nested folders for documents, images, Audio, video
iii) Block storage: stored as fixed sized "blocks" in a rigid arrangement for Enterprise databases.

→ i) object Storage

→ object based Storage. for large unstructured data
→ for archiving, backup & static contents

workings

→ No folder, directory, files or hierarchis
→ All data is stored in flat data Environment as obj.
→ each obj. contains data files with metddata & ID No.

object Storage database

→ uses object's metadata to locate files
→ has 2 tables, 1 → obj. diretory, 2 → object storage.

Indexes

- Time stamp of creation
- Name ID.
- Pending Action
- obj. Name & collection Name ID.

Amazon S3 Storage

→ for applications that needs scale & flexibility
→ can be used for data store Analytics, backups, Archive

→ Accessed through API & managed through dashboards

benifits

- Greater data Analytics
- infinite Scalability
- Faster data interval
- Reduction of cost
- optimized for Resources

usecases

- Data Recovery & Backup
- Analytics
- Static Content

→ ii) File Storage.

→ organized in Folder with metadata
→ Files named, tagged with metadata, size, Creation date, last modified, hirarchical path

EFS : Elastic File System.

→ Accessing shared files
→ NAS : Network Attached Storage server.

→ iii) Block Storage

→ Alternative to file storage.
→ Stores files into equally-sized Chunks
→ To access it os uses unique Address to pull all the blocks & Assembles it into the file
→ effecient & no need to Navigate through directories
→ best for business apps, databases & VM for low latency with performance

EBS : Elastic Block Store.

→ Analoges to DAS : Direct attached Storage or SAN : Storage Area Network

Securing Network in cloud.

Attacks on OSI layers

A) • Network & Filesystem bugs
  • FTP, EMail & 0-day.

P)

S) • Remote procedure Call Worms
  • Port Mapper Exploits

T) • Routing info. protocol Attack.
  • Syn flooding,
  • SEquence No. prediction

N) • IP Smurfing.
  • other Address spoofing Attacks

D) • wired Equivalent privacy Attacks

A P S T N D P
Soft. T Hard
Heart

Page No.
Date

OSI

A
- Produces data, user interaction
- Browser, Mail Client,

P
- Translation (ASCII → EBCDIC)
- Enc./decr.
- Compression

S
- Session establish / maintain / terminate
- Syncronization.
- dialogue control (duplex full/half)

T
- Segmentation & Reassembly
- Service point Addressing
- Connectionful & connectionless service

N
- Routing
- logical Addressing (Sender & Reciever's IP)
- Packets
- Routers

D
- Framing
- Physical Addressing (Sender & Recievers MAC)
- Err control
- flow control
- Access control
- Switch & Bridge

P
- Bit Synchronization
- Bit Rate control (b/sec)
- Physical topologies
- transmission mode (duplex half/full)

⊛ ARP Spoofing/poisoning

ARP

→ Address Resolution protocol is used to reach/discover devices on Network

→ binds IP & MAC Address. for device Identification, in ARP cache.

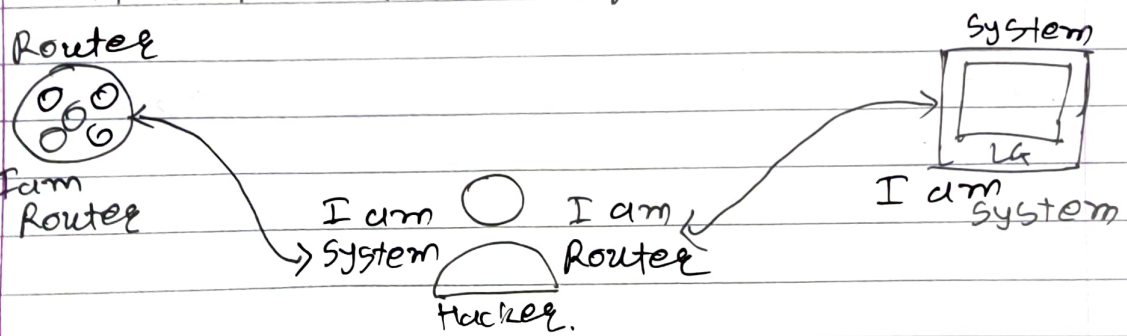
→ if MAC is not known for destination, it sends broadcasts ARP Packet Requests

- No verification Security
- No Authorization.
- host may Accept unknown Response without sending request.

→ Works on 32-bit IP, IPv4, IPv6 uses NDP: Neighbour discovery protocol

Spoofing Attack

→ Mitm attack where attacker intercepts communication betn devices.
→ Attacker must be in ~~Network~~ Network
→ Tools like ARP spoof, driftnet sends forged ARP Responses.
→ Attacker will send forged Responses for Routers or workstation with it's own IP & Mac. & cache gets updated.



Page No.

⊛ detection

→ windows has 'arp -a' command to see ARP table
→ if two IP have same Mac.
→ wireshark

⊛ prevention

→ using VPN
→ using static ARP
→ using packet Filtering.
→ Pentesting

⊛ CAM overflow

→ hubs broadcasts packets from source(A) to all devices, all devices Except destination(B) will drop it.
→ in switch packets are sent to only destination using Content Address Memory (cam) that has a table of mac Addresses of devices.

→ Attacker generates Fake mac Adresses & sends it to the switch, switch will store them into CAM table, Eventually Memory will be full & no legitimate device can be added to network

→ After overflowing CAM table it will start Acting as hub broadcasting all packets so the attacker can capture the packets & Read it.

→ Tool : Ettercap.

Recovery : Shutdown, No shutdown
Prevention : Change defaut port.
: Set limit on cam table
: violation Action Shutdown

⑧ DHCP Starvation

→ Assigns IP automatically on Network devices
→ uses DORA : Discover, offer, Request & Acknowledge
→ DHCP Starvation can be DoS or Mitm attack

→ Attacker sends forged DHCP requests with spoofed MAC & DHCP will assigns IPs eventually Running out of IP to assign. so new legitimate device can not enter the Network.

→ Attacker also may setup Rogue DHCP to assign IP to users, this can provide middleman gateway to legitimate Routers & DNS. that way all traffic can be routed via attacker's Machine. it's MITM attack.

⑧ CDP attack

→ Cisco discovery protocol.
→ for discovering other cisco devices. for Auto Configuration & simplifies connectivity.

→ CDP has data like IP, Native vlan, software version, platform version etc.
→ that may be used for Exploitation, OSINT & may carried out as DoS attack.

(*) Virtualization.

→ techniques

i) Para - virtualization
ii) Full - virtualization
iii) OS - virtualization.

i) Para virtualization

→ does not implement complete Isolation, ~~partial~~
→ Partial approach.
→ Alters OS kernal to subsitute hypercalls
→ hypercalls interacts with virt layer hypervizors.
→ Reduces virtualization overheads.
→ less compatible & portable
→ does not support unmodified OS.

ii) Full - virtualization

→ each OS is independent device.
→ multiple OS in single host.
→ Fully isolated.
→ OS calls translated by virt. manager & Executes

iii) OS - virtualization

→ Just virtualizing OS stating as real device
→ ~~Partial~~ No direct HW Access.
→ Shared storage or devices can be used.

⊛ Types of Virtualization.

i) Application Virt.
ii) Network Virt.
iii) Storage. Virt.
iv) Desktop. Virt.
v) Data Virt.
vi) Server virt.

⊛ Hypervisor

→ abstracts, partitions & Isolates OS & application from real hardware.

→ Manages virtualized components & allows multiple guests at a time.

Features

i) Partitions / Partitioning.
ii) Resource Allocation

⊛ Types of Hypervisor

i) Type 1 : Bare Metal (Esxi, Xen, Hyper-V)
ii) Type 2 : Runs on OS (Vmware, virtual box)
iii) KVM. : intigrated with linux kernel

Docker ~~technolo~~ terminologies.

i) Image : Set of instructions to create container
ii) Container : runtime instanse of image.
iii) Docer file : text. doc. with Commands to Ex. Image.
iv) Engine : main seever that runs containers
- Seever : daemon tat manages Everything
- Rest API : for communication
- client : Cli interface for user interaction

v) hub : official online Repository

Hypervisor	Docker
OS → OS specific	→ Supports only linux
Boot → Slower Boot	→ very Fast Boot.
Security → Extra layer of security with 2 OS	→ depends on container No Extra security
Resource → higher demand	→ low demand.

Multi Factor Authentication

→ two or more verification process

→ Access control Models

i) MAC : Mandatory Access control
ii) DAC : Discrationary Access control
iii) RBAC : Rolebased Access control
iv) LBAC : Lattice-Based Access Control.

→ Policies

- DAC : based on identity of requestor.
- MAC : based on comparing security lable with clearance
- RBAC : based on user roles
- LBAC : based on attributes of user & current Env.

→ Requirement

- Reliable Input : Mechanism to Authenticate.
- Specification : Regulate access at varying lable
- Least privilage : min. Authorization to do it's work
- Saperation of duty : divide steps among diff. individuals
- open/close policies : access specific access exept prohibited
- policy combinations & conflict resolution
- Administrative policies : who can add, delete, modify rules

→ Elements

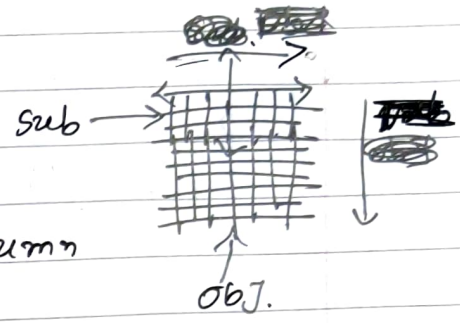
- Subject : entity that can access object
  - → Process representing user/application
  - → 3 classes owner, group, world

- Object : access controlled Resource
  - → file, Records, Programs
  - → number/type depend on Environment

- Access right : the way which subject Accesses object
  - → Read, write, Execute, delete, Create, search

① DAC

→ Uses Access Matrix
→ Subject in rows
→ object in columns
→ Can decomposed by row or column



Structure

→ Access control lists (column)
→ Capability tickets (Rows)

Ex

Subjects	Objects F1	F2	F3
U1	RWX	RW	R
U2	RW	R	RX
U3	RWX	RWX	RX

② MAC

→ Bell la Padula Model
  → Security levels
    - Top Secret
    - Secret
    - Confidential
    - Unclassified

→ Reading Info.

- flows up, Not down
  - → Read up ⇒ disallowd
  - → read down ⇒ allowd

→ Writing Info

- flows up, not down
  - → write up ⇒ allowd
  - → write down ⇒ disallowd

(*) Constraints : Conditions on role(s)

→ Manualy Exclusive
→ Cardinality (Max users allowd)
→ Prerequisite role : based on role

(*) Attribute based Access control

→ Types of Attribute

i) Subjective.
- Name
- org.
- title.

ii) objective
- Title
- Author
- date.

iii) Environmental.
- specific date
- Current virus/hacker Activities
- Security level