PERCIPIO 1-5

Thursday, December 1, 2022 7:11 PM

Aws cli configuring auth

- Aws configure(initialize-awsdefaults --region us-east1 in powershell)
- Enter access key ID
- Enter Secret Access key
- Default region name

Aws configure list for listing configurations

To query the list of buckets using cli
Aws s3api list-buckets --query "Buckets[].Name"

Ec2-user is the default username by ec2

AWS outposts:

- Running aws services on premises or edge locations
- Comes in outpost racks or server rack unit
- Used for local data processing,data residency restrictions,latency
- Supports ECS,EC2,EKS,EBS ETC..
- Resource actions: laucnh instance,create subnet,volume,create/manage s3bucket,

AWS Transfer:

- First create a role for giving access to the transfer api of aws to the full s3 bucket access
- Create a role and give the principal as the transfer.aws.com or something and change it in trust relationship(which we will use for adding user to this transfer family with ssh keys generated)
- SFTP,AS2,FTPS,FTP are the protocols we can enable
- To store and access data over the protocols we can select s3,efs
- The sftp files will be stored in s3 buckets

AWS Snow family:

- Aws snowcone
  - Smallest
  - SSD(14TB),HDD(8TB)
  - Data copied to nfs mount point
  - Offline device shipping data transfer or AWS datasync preinstalled
- Aws snowball
  - Edge compute optimized option to process data locally on premises using aws lambda code
  - Download the snowball edge client to unlock the device once it is powered up
  - Copy data to device using s3 gui or nfs mount point
- Aws OpsHub - a GUI app for managing snow family

devices or services

RDS proxy:

- Is a managed service
- Pools database connections to increase connections scalability
- Reduces failover time with automatic database failover

DynamoDB:

Should give partition , sort key

- Dynamodb satndard(frequently accessed data)
- Dynamodb standard-1A(infrequently acccessed data)

DocumentDB:=>mongodb

If we are going to create access policy we would need a arn(amazon resource name)

S3 BUCKETS:

- Region
- Object versioning
- Transfer acceleration
- ACL's
- Bucket policy
- Public access
- Bucket lock(which should be done while creating bucket which enables versioning if enabled) - legal hold, retention

Cli commands:

Aws s3api create-bucket --bucket name --regio us-east1 -> (for creating deleting listingetc..) s3api list-buckets
Aws s3 -> for managing objects(cp,rm,mv)
Aws s3 mb s3://name
Aws s3 cp c:/tmp s3://name --recursive --storage-class GLACIER
Aws s3 ls s3://name

Aws s3api create-bucket --bucket name --region eu-east1 --create-bucket-configration LocationConstraint-eu-west-1

Powrshell commands:
Get-module -list *aws* (to lost all modules which has the name aws in it in powershell)

Get-command -module awspowershell | more
Get-command *new*s3* (new(remove)-s3bucket -bucketname name, get-s3bucket or get-s3bucket | select bucketname)

Get-awscredential -listprofiledetail (to list all the auth's we have)

Initialize-awsdefaultconfiguration -profilename profilenamelistedinabovecmd (to initialize a auth)

Write-S3Object -bucketname name -file c:/tmp/fie.txt -CannedACLName Private
Get(remove)-s3objects -bucketname name foldername/

Storage tiers:

- Standard(more than once a month)
- Intelligent tier(unknown access pattern)
- Standard IA(once a month)
- One Zone IA(once a month in single AZ)
- Glacier Instant retrieval(once a quarter)
- Glacier flexible retrieval(once a year with minutes-hours retrieval time)
- Glacier Deep archive(less than oncena year retrieval in hours)
- Reduced redundancy(noncritical frequently accessed data)

Bucket policies:

We can use policy generator

- Type of policy:
  SQS queue policy,s3 bucket policy,VPC endpoint policy,IAM policy,SNS topic policy
- Choose s3 bucket policy
- Effect: allow/deny
- Principal:(*,accoutn no,user arn etc..)
- AWS service: (amazon s3,etc..)
- Actions: (createbukcet,craetejob, etc…)
- ARN: (bucket's ARN)
- Add statement,generate policy

Storage Gateway:

- On premises access to scalable cloud storage
- Use locally with low network latency

Types:

- S3 file gateway(on prem SMB/NFS access)
- Fsx file gateway(SMB,NTFS SUPPORT)
- Tape gateway
- Volume gateway(CONNECT USING Iscsi initiator)

Host platforms:

- VMWare ESXi
- Microsoft hyper-v
- Linux kvm
- Amazon ec2
- Hardware appliance

Steps will be provided to setup the host for gateway to contact aws

Encryption:

- the encryption settings are applied to newer uploads
- Need to edcit the serverside encryption for all existing objects

Set-s3bucketencryption -bucketname ….

Aws s3 cp s3://name/filename s3://name/filename --sse(serversideencryption) AES256

AWS Cloudfront cache behaviour settings:

- Path pattern and wildcards
- Http to https redirects
- Allowed http methods
- Object caching ttl

AWS Cloudfront:

- Origin of the content(s3,elastic load baalner etc..)
- Origin access identity to restrict public access and give it to only cloudfront
- Customize headers,paths
- Smooth streaming option that uses microsoft IIS
- Function associations(lambda etc..) based on viewer request,response,origin request,response
- Error pages
- Geographic restrictions

Distributed domain we use as url

VPC:

- Subnets
- DHCP options
- Network ACL for subnets
- Route table
- 50 tags

Aws ec2 create-vpc,create-tags --cidr-block 12.0.0.0/16
Aws ec2 describe-vpcs
Aws ec2 create-subnet --vpc-id name --cidr-block 12.0.1.0/23
Aws ec2 modify-subnet-attribute --subnet-id id --map-public-ip-on-launch

New-Ec2Vpc
Get-Ec2Vpc
New-Ec2Tag/Subnet/Tag

Elastic ip address <=> static ip
Can associate elastic ip to instance from elastic ip section and choose to reallocate in future

Aws ec2 allocate-address / describe-addresses / create-tags
Aws ec2 associate-address --instance-id id --allocation-id ipid

Powershell: new-ec2address/get-ec2address/new-ec2tag

(Un)Register-Ec2Address -instanceid id -allocationid ipid

VPC Peering:

- Link vpc's within or between aws acounts
- Within or between aws regions
- VPC1->VPC2->VPC3 here VPC1,VPC3 should not be connected/peered
- Accept connection in acceptor

- Adjust route table for becoming active/allowit(add the vpc cidr that is peered to the peering connnectionaname)

Aws ec2 create-vpc-peering-connection --vpc-id id --peer-vpc-id id

New-EC2VpcPeeringConnection -vpcid source -peervpcid target --peerownerid id --peerregion region

AWS service endpoints:

- Allow programmatic api/cli connectivity to service using a url
- Eg: https://dynamo.db.us-east-1.amazonaws.com
- Some services in specific regions support federal information processing standards(FIPS) endpoints which require atleast TLS 1.2 HTTP connecitons
- Eg: https://iam-fips.amazonaws.com
- VPC endpoints to allow private subnet resources access to aws services
- In case of vpc endpoints we don't need to bother about the route table and ip cidr overlap and allows direct connect with aws resources that have endpoints configured
- Endpoint types: - interface,gateway load balancer,gateway

VPN:

Site to site VPN

- Customer gateway -> customer side vpn appliance
- Virtual private gateway -> aws side vpn appliance
- Encrypted ipsec vpn tunnel
- IPV4,IPV6 traffic
- Onpre customer gateway <=encrpted ipsec vpn=> aws vpg
- Route propagation to yes(automatic route added by learning)

AWS client VPN

- VPN Client
- VPN Client endpoint in AWS(directory authentication/mutual certificate auth)
- Encrypted vpn tunnel
- Aws Certificate manager to request a public certificate from amazon or private certificate from our organizations CA

To generate certificates

```
Openssl req -newkey rsa:2048 -nodes -keyout myvpn.key -out myvpn,.csr
Openssl req -text -noout -verify -in myvpn.csr
Openssl x509 -signkey myvpn.key -in myvpn.csr -req -days 365 -out myvpn.crt
```

We can configure the acm certificate and create a client vpn endpoint with that certificate and provide a ip address and after downloading the configuration we can drag drop on the openvpn connect for windows or any other client . After importing into the app it will prefill and ask to cnnect without certificate or wth them

Direct Connect:

- Dedicated private network circuit(1gbps-100gbps)
- Hosted connection stems from an AWS Direct Connect Partner(50mbps-100gbps)

On prem router must be configured
OSI layer 2 data link layer point to point connection
Virtual interfaces
Split one circuit into multiple interfaces
LAG(logical interface aggregating group is used to aaggregate ,multiple dedicated connections ata single endpoint

AWS privatelink links specific vpc or resources privtely

Nat gateway is attached to a subnet
Type:public/private

Connections from internet will not traverse through nat gateway

Internet gateway for vice versa internet communications

Egress only internet gateway is a outbound only(ipv6)
Edit route table as ::/0 to eigw

Route 53:

- Health checks for app reachability
- Traffic policy rules (failover rule,weighted rule,etc..)
  Weighted rule directs traffic based on weights defined where the same app is hosted on multiple hosts
  Failover rule(active-passive failover)second resource accepts traffic if first one is not available
  Geolocation rule(based on origin of client dns queries)
  Latency rule (when sam service running multiple aws data centeres)
  Multivalue answer rule(can have upto 8 dns query results,response is a random selection based on an optional health check)
  Geoproximity rule-(Specify resource location region,lat,long, dns client query answers will route to resources nearest them)

DNS Firewall:

- Domain lists(aws managed,our own known attack domains)
- Rule group(dns exfiltration)
  Allow,block,alert
  Nodata,nxdomain,override)action
  Can give rule priority

PERCIPIO 6-10

Sunday, January 29, 2023 1:57 PM

```
Aws ec2 run-instances --image-id amiid --count 1 --instance-type t2.micro --key-name KeyPair1 --security-group-ids id --subnet-id id
```

```
New-Ec2Instance -imageid amiid -mincount 1 -maxcount 1 -keyname keyname -securitygroupid id -instancetype type -subnetid id
```

```
(get-ec2instance).instances.tags
```

Putty private key file is .ppk and for openssh .pem

For hibernation we need to enable at instance launch and set the volume as encrypted

```
Aws ec2 stop-instances --instance-ids id --hibernate $true
```

```
Stop-ec2instance -instanceid id -hibernate $true
```

```
Aws ec2 describe-instance-types --instance-types t2.micro
```

```
Aws ec2 describe-instance-atrribute --instance-id id --attribute instanceType
```

Aws auto scaling can be done for ec2 instances , elastic container services, amazon dynamodb,amazon aurora database replicas etc..

Launch configuration <=> instance template
Auto scaling groups <=> instances from this template

AWS serverless compute for automatic instance scaling and management

AWS batch for batch jobs scheduling etc..

Application containers for running app code and its dependencies packaged into one container
Application migration using App2Container which is a cli tool that converts a java or .NET App into a container.

AWS fargate is a serverless compute engine for app containers(works with ECS,EKS)

AWS elastic mapreduce is managed cluster platform that allows distributed processing(based on hadoop)

AWS elasticache is a cache engine with memcached(key,value storage),redis (complex datatypes eg: hashes,lists,bitmaps etc..
Build in replication is supported in redis not in memchached

AMAZON athena ansi sql querying service for data stored in s3

Amazon quicksight is a BI service and support querying

Amazon kinesis is a realtime processing streaming data

Aws glue is a serverless ETL service

Aws lake formation services are datalakes

Amazon simple queue service to send store read messages(like pubsub) , enables app component decoupling , provides standard(atleast once delivery) and FIFO (exactly once delivery)queues
Dead letter queue to receive undelivered messages

AWS lambda is a serverless cloud service tht runs code (java,python,c# ,etc..) , upload .zip file or container image or edit in console , triggered by an event , has code signing

Amazon api gateway is a scalable serverless solution that can process API CALLS

Amazon cognito to allow developers to add user signup and sign in functionality to app , providers like google,amazon,fb,apple etc..

Web application firewall intercepts requests to http based apps , api's , protects from sql injection,cross site scripting etc.. , uses OWASP top 10 rule sets
It can be placed in ec2 instacnes, amazon api gateway,cloudfront distribution,app load baalncer

- Web traffic filtering(ip address,http headers,body)
- Rate limits and blocking bot traffic
- Fraud control,prevent login page attacks
- Security metrics monitoring using cloudwatch

AWS firewall manager ensure compliance with security rules,centrally manage multiple WAF deployment with or within aws accounts

AWS secrets manager database credentials,api keys,oauth tokens can be stored in this.
Types:

- Amazon rds db credentials
- Amazon documentdb credentials
- Amazon redshift credentials
- Other db
- Api key,oauth token,other

Can rotate secrets automatically using lambda,schedule
Resource permissions,replication across regions

Amazon elastic container service is runningcontainers in AWS, amazon fargate intergaration , amazon ecs anywhere , elastic container registry , container version updates , can create ecs cluser for high availablity and ecs task for running specifictasks and it can have public ip

We can create application type as service,tasks

ECR is container registry

```
Aws ecr get-login-password --region us-east1 | docker login --username AWS --password-stdin 231312313.dkr.ecr.region.aazonaws.com
```

```
Docker build -t .
Docker tag ssdd ecrimagename:latest
```

EKS is kubernetes managed service
Can use cloudformation templates for creating resources like this

```
Aws eks create-cluster --region region --name clutsername --role-arn awsrolearn --resources-vpc-config subnetIds=subnetid,subnetid,subnetid,securityGroupIds=sgid
```

Aws eks desribe-cluster --region region --name name --query cluster.status

Aws eks update-kubeconfig --region region --name clustername

After this we can create node groups with a new role which has ec2role,ecrreadrole

AWS step functions are like dags which have step by step instructions like if this is true execute this step then execute this in parallel wait for these parallel execution to execute the other one etc..
We can create statemachines to design workflow visually or in code

AWS speech services:

- Voice to text
- Web app customer service chatbots
- Text sentiment analysis,spam detection

Amazon comprehend is to derive insights from various text sources, PII detection,redaction,multi spoken language support

Amazon polly is a cloud based text to speech service, includes male,female voices

AWS app runner is used for running web app with deployment types to manual,automatic(deploys a new versdion of service for each image push) (ecr image/source code repo) with autoscaling,health checks etc..

Aws launch templates is like auto scaling groups laucnh config but have versioning of templates or create spot fleet and both are same

Encryption using server side encryption using s3managed keys,kms, rds aes encryption pracle,microsoft sql server tde

Aws shield is used for ddos protection , integrates with AWS WAF,EC2,CLOUDFRONT,ROUTE53 and has standard,advanced tiers

IPV4 is 32 bit address, dotted decimal notation between octets(collection of eight binary bits which falls in between 0 and 255), since 1970s in layer 3 network
Reserved ipv4 privat ranges:

- 10.0.0.0 to 10.255.255.255
- 172.16.0.0 to 172.31.255.255
- 192.168.0.0 to 192.168.255.255

IPV6 eight hextets seperated by colons 128 bit addresses
Each hextet value falls between 0 and F
Tcp:
22 - ssh
80 - http
443 - https
3389 - rdp
25 - smtp

UDP:
53 - client dns query
67 - dhcp server listering port
68 - dhcp client listeing port
161 - snmp

We can mirror traffic using mirror sessions , targets,filters(inbound,outbound rules)
Source are only network interfaces,targets are interfaces,nlb,glb

Amazon guard duty will give findings it found in resources , protects s3,kubernetes etc..

Amazon macie is for data classification in s3 bukets that will discovr in schedule as well with data identifiers(include,exclude)

Security(cryptography,pki):

- Symmetric encryption - one key
- Asymmetric- public,private key
- Hashing - integrity

Cryptography,storage:

- Self-encrypting drive
- Os encryption-windows EFS,bitlocker
- Aws s3 bucket encryption,ecr encryption
- Rds database encryption

AWS certificate manager,AWS KMS, S3 bucket encryption using aes 256,vpn encrypted tunnels,https encrypted connections

Public key infrastrcture(PKI):

- Hierarchy of digital security certs issued by CA
- Public trusted CA's,private internal untrusted CA's
- PKI cert Issuance for users,devices,software etc..
- Certificate issuance like wildcard certificates(*.domanin.com),extended domain validation,code signing certs

Lifecycle:

- Certificate signing request(CSR)
- Issuance
- Use
- Revocation
- Renewal/expiry - then again CSR and so

Pki cert contentss:

- Subjectname
- Issuing ca,ca sign
- Public key(private in os secret store)
- Certificate usage
- Issue and expiry date
- Other attributes

Revoked options are given by CRL(certificate revocatio lists) , online certificate status protocol(OCSP)

In **AWS KMS** we can give adminstrative permissions for adminstrating kms for a user,role or etc., usage permissions for users etc.. , key policy will be created which can be edited later

For custom secret store we can use cloud hsm which is specialized one for managing cryptographic operations

Aws managed,customer managed,custom secret store(hsm)

We will have the option to rotate the keys in only the encrypt and decrypt option for both symmetric,asymmetric

Aws kms get-key-rotation-status/disable-key-rotation --key-id id

AWS certificate manager(acm)

- Private CA-issued certificates are not trusted by default
- We can request a public cert by amazon or import or create private CA

- Easy cert provisioning where no key generation,csr submission invloved
- Built in cert management
- Create,import ,renew certs

CREATING A PRIVATE CA:

- Type: root CA,subordinate CA
- Encryption algo
- Revocation standard
- Configure CA root cert

When requesting private certificate(SSL/TLS) from our own ca we have to give domain name when requesting and create a passphrase for exporting the cert(we can use this cert for cloud front ssl cert)

When requesting public cert we need to give a validation type as dns validation or email validatio to validate the domain

We can attach https listener target group and attahc cert

Percipio 11-16

Monday, January 30, 2023 12:32 PM

IAM:

- Authentication
  - Proof of identity
  - Single factor - user,pass
  - Multi factor - user,pass,smartcard
- Authorization
  - Controlled access to apps and aws resources
  - Policies are collections of permissions
  - Custom or built in policies

For users console password,mfa device signed certs,access keys etc are present

Password policy can be set for giving set of rules for setting a password for a user

When creating users we can give them access key,security key access or console password access or both

```
Aws iam create-user --user-name name
```

```
Aws iam get-user --user-name name
```

```
Aws iam list-users
```

```
Aws iam create-login-profile --user-name name --password pass --password-reset-required
```

Aws iam change-password --old-password oldpass--new-password newpass (this will work only if the user is logged in to the cli)

```
Powershell: new-iamuser -username uname
            Get-iamuser
            Get-iamusers
```

IAM groups cant be nested like group inside a group
Access advisor will show the services we can access using the group permissions
Can use policy simulator for simulatin the access

```
Aws iam create-groups --group-name name
Aws iam list-groups --query "Groups[*].GroupName"
Aws iam get-group --group-name aaname
Aws iam add-user-to-group --user-name name --group-name addadmins
```

```
Aws iam list-groups-for-user --user-name username'
Aws iam delete-group --group-name name
```

Aws iam remove-user-from-group --group-name name --user-name name(1st to do befoe deleting a group)

```
Powershell: new-iamgroup -groupname app1
            Get-iamgroups
Add-iamusertogroup -username mbishop -groupname name
Get-iamgroup -groupname name
```

```
Write-IAMGroupPlicy -groupname name -policyname 'AllAccessExceptIAM' -policydocument
$policy({"veriosn:"L"","statement":[{"effect""":"allow","notaction:"iam.*","resource":"*"}]}
```

IAMPOLICY:

- Permissions
- Policy usage like where it is used
- Policy versions (changes)
- Access advisor

IAMROLE:

- Service access to be provided
- Entity types:
  - Aws service
  - Aws account
  - Web identity
  - SAML 2.0 federation
  - Custom trust policy

Keyfob -> security devices plugged in for tokens

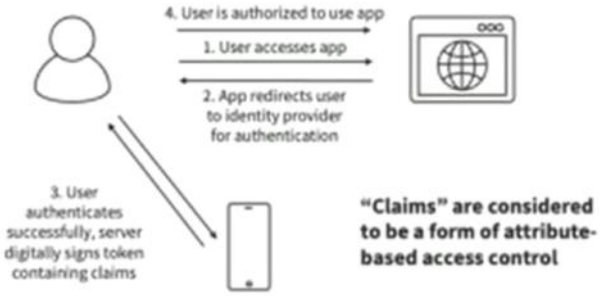
AWS DIRECTORY SERVICES:

- AWS simple AD
  - Its not true microsoft AD instead uses samba 4 as ds
  - Supports small(upto 2000 objects),large(upto20000)
  - Does not support MFA , domain trusts,powershell,ad recycle bin
- AWS MANAGED microsoft AD
  - Is microsoft AD DS but hosted in aws
  - Join ec2,rds instances to the domain
  - Standard(1gb storage,30000 objects),enterprise(17gb storage,500000 objects) editions
- AD connector
  - Links existing ad to aws ad
  - One connector for each on prem AD domain
- Amazon cognito user pools

When creating the ec2 instance we can provide the domain join directory with the created directory and the instance profile(iam role)
When connecting to windows vm through rdp we can specify the username from the directory netbios name from the directory like corp(netbios)/Admin(username)

We can login to the admin windows server instance and goto windows adminstrative services and go to ad service and create a user in that

An identity federation claims are assertions about a user or device, are contained within a digitally signed secuirty token and are consumed by apps


4. User is authorized to use app
1. User accesses app
2. App redirects user to identity provider for authentication
3. User authenticates successfully, server digitally signs token containing claims
"Claims" are considered to be a form of attribute-based access control

AWS identity solutions:

- Single sign on
- Aws security token service

Network ACL:(layer4)

- Packet filtering firewall solution
- Assoiated with subnet

```
Aws ec2 describe-vpcs
Aws ec2 create-network-acl --vpc-id id
Aws ec2 create-network-acl-entry  --network-acl-id id --ingress --rule-number 100 --protocol tcp --
port-range from=3389,to=3389 --cidr-block 0.0.0.0/0 --rule-action allow
Aws ec2 replace-network-acl-association --association-id id --network-acl-id id
```

```
Powershell: New-EC@NetworkACl -vpcid id
New-Ec2NetwrorkAclEntry -networkaclid id -egress $false -rulenumber 100 -protocol 6 -
portrange_from 443 -portrange_to 443 -cidrblock block -ruleaction allow
```

Security Groups:

- Apply to ec2 insatnce not subnets
- Can specify allow rules only
- Have stateful firewall rules(changes to incoming rule apply to outgoing rule as well)
- No priority number
- Checked after acl when both are in place

Can use reachability analyzer to test the connectivity is correct or not

```
Aws ec2 create-security-group --group-name name  --vpc-id id --descripiton desc
```

```
Aws ec2 authorize-security-group-ingress --group-id sqid --protocol tcp --port 3389 --cidr 12.0.0.0/0
```

```
Powershell: New-Ec2SecurityGroup -groupname groupname -description desc -vpcid id
```

```
$rule = @{IpProtocol="tcp"; FromPort="22"; ToPort="22"; IpRanges="199.126.129.0.24"}
```

```
Grant-Ec2SecurityGroupIngress -groupid id -ippermission $rule
```

AWS network firewall:

- Firewall
- Firewall policies
- Firewall rule groups(S-tuple(source,target ip,port etc.),Domain lists,Suricata compatible IPS rules)
- Traffic direction (any,forward)
- Action(pass,drop,alert)
- One or rule groups to firewall policy

Trusted Advisor:

- Summary of health checks in our aws account / services
- Security/cost optimization/performance/fault tolerance
- For cost optimization we need to upgrade the support plan

AWS organizations => aws accounts
Can create organizational units under root and move the accounts to particular OU's , also can invute throgh email or using existing aws account

Policies for aws organizations
Type: all services optout policies
Backup policies
Service control policies (centralized access ppermissions control)
Tag policies

AWS systems manager manages all resources in aws like gain operational insight and take action
Config types:

- Host management(configures iam roles and enables commonly used systems manager

capabilities to securely manage ec2)
- Config recording(tracking and recording of changes to aws resources types you choose)
- Conformance packs(collecitons of aws config rules and remediation actions that can e deployed ads a single entity)
- Change manager(configures the IAM role needed for change manager to incoke management operations across org)
- Devops guru(improve apps operational peformace and availabiity)
- Distributor(distribution of software packages agents to ec2 instaces)

We can schedule and give targets on which It should act on

We can create incident manager to respond to incidents with incident plans

We can create change manager like service now for creating crq's

Can have fleet of instance to manage

AWS CONFIG is a inventory tracking service(conformance packs)
Complaintn and non complaint resources listed here
We can also disable the inventory recording in the settings , can query the resource details like nonn complaont resources etc..

AWS batch can have fargate intergration,spot,gargate spot,spot,on demand , can schdule queue etc..
BASH command on contianer image or json type for complex command to run on image

In **AWS CLOUDFORMAITON TEMPLATES** is usd to template rsource creation like also used in design to visually create templates , tempates are known as stacks and satcksets

Savings plan are 1-3 year commitment , reserved instances are like per region instance type basis

Instance scheduler to start ad stop instances in schdeules based on tagging

AWS budgets => cost budget,usage budget,savings plan budget,reservation budget

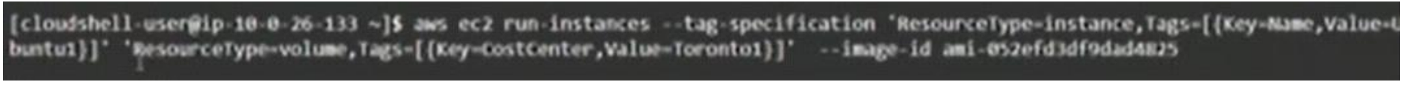
With requester pays option in s3 bucket we can save the in and out transfer to the objects from the s3 bucket

Aws s3api get-object --requester=pays(to confirm t is enabled)

We can set a maximum price for the spot instance when enabling it, request type=> one time ,persistent

We can resell the reserved instances if we backoff

In the billing console we can dd a cost allocation tags(user-defined(a key that will be used everywhere to trackcosst),aws generated tags)



```
[cloudshell-user@ip-10-0-26-133 ~]$ aws ec2 run-instances --tag-specification 'ResourceType=instance,Tags=[{Key=Name,Value=U
buntu1}]' 'ResourceType=volume,Tags=[{Key=CostCenter,Value=Toronto1}]'  --image-id ami-052efd3df9dad4825
```

RTO(recovery time objective)(maximum allowable downtime

RPO(recovery point objective)(maximum tolerable amount of data loss)

AWS elastic disaster recovery(DRS) is based on continous re[plication , launching recovery instances,supports failover,failback

In aws amazon s3 glacier we have vault that serves as the containers for the archives and we can set vault policies, data retrieval settings
(can only upload thorugh cli)

```
Aws glacier upload-archive --account-id id --vault-name vault1 --body filename
Aws glacier initiate-job --account-id id --vault-name vault1 --job-paramters "{"Type":"inventory-retrival"}"
Aws glacier describe-job aacount-id id --vault-name vault1 --job-id id
Aws glacier get-job-output --account-id id --vault-name name --job-id id output.json
```

In S3 we can set replication rules for replicating objects into another bucket in any account and should be versioned and can choose destination storage class,can choose to replicate certain filtered objects or existing objects (all)

If bucket versioning is turned on and we go to the veriosning settings we can suspend the verisoning of new objects by preserve existing objects

Load balancer types:

- Application load balancer(OSI layer 7,HTTP,HTTPS)
- Gateway Load balaner : OSI layers 3,4, IP
- Network load balaner: OSI layer 4 , TCP,UDP

AWS backup:

- Build a centralized cloud backup using console,cli
- Stored in a backup vault
- Integrated with S3 and works with following:
  - EC2 instances
  - EBS volumes
  - Amazon RDS databases
  - DynamoDB tables
  - EFS file systems
  - Storage gateway volumes
- Backup scheduling and retention
- Backup lifecycle management

STEPS:

- Create a backup plan(using json ,tempkate etc..)
- Assign resources to the plan(can use tags to select resources)(all resource types or specific resource types)
- Create a new backup vault (optional)

Protected resources-> on demand backup

We can restore fullly or item -level , with the file's iam role or no iam role
Can export mysql instance(rds) into s3 and only can create snapshot at cluster leevl not instance level

EXTRA PERCIPIO

Thursday, February 2, 2023 3:08 PM

```
Aws ec2 create-vpc --cidr-block 12.0.0.0/16
Aws ec2 create-subnet --vpc-id id --cidr-block cidr
```

```
New-EC2VPC -cidrblock 13.0.0.0/16
New-EC2Subnet -vpcid id -cidrblock cidr
New-EC2Tag -resourceid id -tag @{Key="Name"; Value="voc"}
```

```
Aws ec2 describe-vpcs --output table(to output as a table)
```

```
Aws ec2 create-dhcp-options -dhcp-configurations "{/"Key"/":/"domain-name/",/"Values/":[/"sample.com/"]}""{/"Key/":/"domain-name-servers/",/":[/"192.168.1.5/"]}" --region us-east1
```

Route 53 DNS (how record routing options):

- Simple
- Weighted
- Latency
- Failover(primary,secondary records)
- Geolocation

```
Aws ec2 create-security-group --group-name name --vpc-id id
```

EC2 PLACEMENTS GROUPS:

- Vm placement on hardware
- Consider vm workload
- Vm can be in only one placement group
- Specify when launching ec2 instance
  Placement strategy:
  - Cluster
  - Spread
  - Partition

SQS standard queue)

- Default queue type
- Send,receive,delete message
- Atleast once message delivery
- Order of messages(sometimes come out of order,can give some info in messages to reorder after receiving)

SQS dead letter queue:

- Debugging appps or messaging system
- Standard or fifo queue(if required ordering)
- Redrive policy is a somthign like a condition on which the source system messages are pushed to dead letter queue(cannot process)
- Must be in same region that uses this dead letter queue

SQS FIFO queue:

- With or without server side encryption
- Exactly once processing
- Batching - upto 3000 transactions per second
- Without batching - 300 api callsper sec
  Message deduplication ID

Message group ID
Receive request attempt ID
Sequence number

Standard queue cannot be converted to FIFO queue
(max 10 messages we can get_,30 sec default visibility timeout,14 days max retention period)
String,number,binary are the message attributes available in sending messages,custom

Polling is the subscriber receving messages in their side

We can create a dead letter queue by creating two standard queuees and editing one of them and enable dead letter queue and give its arn with maximum receives

Aws sqs create-queue --queu-name name --attributes file://create-queue.json

Aws sqs list-queues

Aws sqs send-message --queue-url url --region region --message-body message

Aws sqs receive-message --queue-url url --wait-time-seconds 25 --visibility-timeout 30 --max-number-of-messages 5 --region region

AWS KINESIS:
Kinesis data streams,kinesis data firehose(stream content to specific taregt), kinesis data analytics(process stream data),kinesis video streams

- Aws kinesis list-streams
- Aws kinesis list-shards --stream-name name
- Aws kinesis put-records --stream-name name --records Data=blobdata1,PartitionKey=partitionkey1 Data=blobdata2,PartitionKey=partitionkey2 (puts a sequence number ,shard id for the records)
- Aws kinesis describe-stream --stream-name name
- Aws kinesis get-shard-iterator --stream-name name --shard-id id --shard-iterator-type TRIM_HORIZON(read oldest data record in the shard)
- Aws kinesis get-records --shard-iterator itr
- For best effort scale the autoscaling group should be with cpu load on consumers rather than putting on shards
- The KCL(kinesis client library) is used for connecting stream , pull,push records,process records etc..

In amazon simple notification service subcriptions we have http,https,email,emailjson,amazon sqs,lambda,platform app endpoint,sms(need to confirm subscription through email)

AWS step functions :

- Visual workflow(State machine ex)
- Call for a service(request response,run a job,wait for callback)

Create a state machine,start a new execution,update a state machine

AWS simple workflow service(SWS)(build applications,coordinate tasks,scheduling,implement workers in ec2)

- AWS SDK'S
- AWS FLOW FRAMEWORK(JAVA,RUBY)
- HTTP SERVICE API
- DEV ENV

Amazon MQ(based on apache mq mmessage broker service)(active/standby broker,singleinstacne broker)

- Wire-level protocols

    - AMQP
    - MQTT
    - OPENWIRE
    - STOMP
- Atributes
    - Name
    - Id
    - Arn
    - Activemq web console url
    - Wire-level protocol endpoints
- State
    - Creationfailed
    - Creation is progress
    - Deletion in progress
    - Reboot in progress
    - Running

Elastic transcoder:

- Media file manipulation
- Playback on a wide array of devices
- Size, type,Quality
- Target formats:
    - Mp4
    - Flac audio
    - Animated gif
    - Mp3
    - h.264 audio and video
- Process
    - S3 bucket containing the contents
    - Transcoding
    - S3 bucket containing transcoded content

CloudFormation

- IaC
- Aws resource deployment,updating
- Templates(yaml,json)
- Parameters
- Reusability
- Automation
- Can use desingner to build cloudformation stacks

Change sets (see how changesmight impact live resources)
Drift occurs when an unexpected configuration differs from the actual configurations(manually edited after applying template)
Can be detected on entire stack,individual resources,
DRIFT DETECTION:

- CREATE_COMPLETE
- UPDATE_COMPLETE
- UPDATE_ROLLBACK_COMPLETE
- UPDATE_ROLLBACK_FAILED

OPERATION STATUS:

- DETECTION_COMPLETE
- DETECTION_FAILED
- DETECTION_IN_PROGRESS

DRIFT STATUS:

- DRIFTED
- NOT_CHECKED

- IN_SYNC

State manager,powershell DSC

Security Hub:

- Security alerts
- Compliance reports
- Data sources
  - Cloudwatch
  - Cloudtrail
  - Aws config
  - Amazon guard duty
  - Etc..
- Custom actions
  - Helpdesk ticket
  - Chat session
  - Email notifications
  - SIEM forwarding

Cloudwatch:

- Onprem(agent)
- Aws

- Dashboards
- Event routing
- Detect suspicious activity
- Resource metric alrms
- Export log data to s3

CloudTrail:

- Auditing solution
- Resource changetracking
- Detect suspicious activity
- Event history(90 days)
- CloudTrail Trail(archiving to s3)
- Legal and regulatory complaince

AWS TRUSTED ADVISOR:

- Cost optimization
- Performance
- Security
- Fault tolerance
- Service limits

IMP

Friday, February 3, 2023 7:29 PM

Amazon AppFlow:

- Secure data exchange between saas apps and aws services
- Data sync
- Data aggregation
- Data tracking
- Secure data
- Private data(privatelink(not traverse through pubic internet))

Flows:

- Flow
- Data mapping
- Filter
- Trigger

1000 flows limit

Amazon Athena:

- Interactive query service(data analysis in s3)
- Serverless
- Standard sql
- Athena federated query

Components

- Tables
- Databased
- Data catalog(through aws glue)

Amazon cloudsearch:

- Create search domain
- Upload and index data
- Submit search requests

Facet is a index field
Searching:

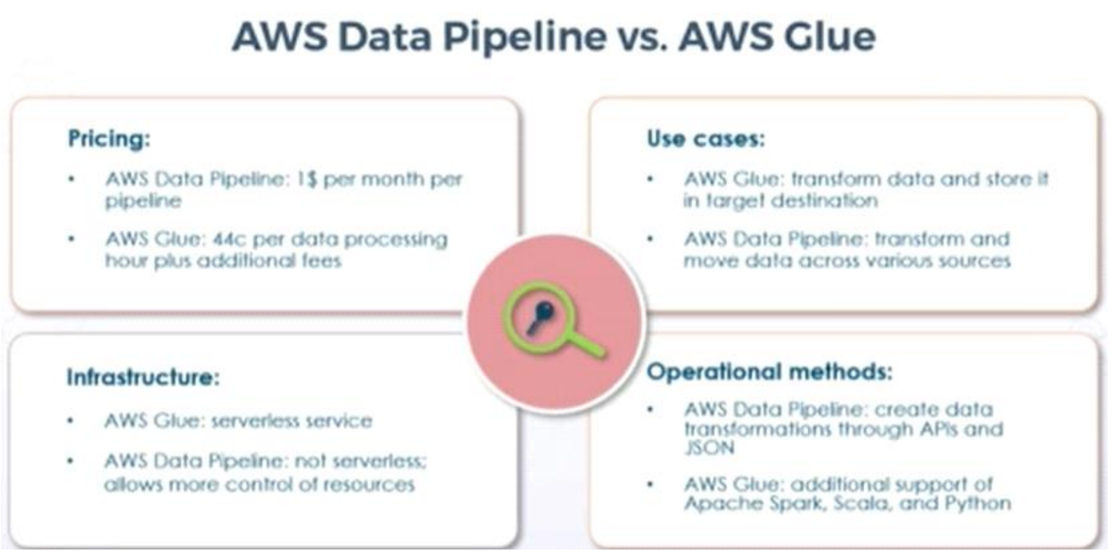
- Simple
- Structured
- Lucene
- Dismax

Elasticsearch is an open source database for search and analytics
Elastic mapreduce (hadoop,spark)

- Submit work
- Cluster lifecycle
- Emr notebooks

AWS glue

- Build data warehouses,datalakes
- Generate output streams
- Api operations
- Aws glue console
- Jobs created using data catalog
  Data targets:
    - S3
    - Rds
    - 3rd party jdbc dbs
    - Mongodb,amazon document db



AWS Data Pipeline vs. AWS Glue

Pricing:

- AWS Data Pipeline: 1$ per month per pipeline
- AWS Glue: 44c per data processing hour plus additional fees

Use cases:

- AWS Glue: transform data and store it in target destination
- AWS Data Pipeline: transform and move data across various sources

Infrastructure:

- AWS Glue: serverless service
- AWS Data Pipeline: not serverless; allows more control of resources

Operational methods:

- AWS Data Pipeline: create data transformations through APIs and JSON
- AWS Glue: additional support of Apache Spark, Scala, and Python

Amazon rekognition can perform video analysis , identity objects,people,text,scenes generating an amazon kinesis streams

Amazon eventbridge is a serverless bus service

- Realtime data stream from/to saas,aws services
- Event driven serverless microarchitecture

Functionality

- Receives an event
- Applies rul to target
- Rules match events to targets by event pattern or schedule

Bus is a pipeline to receive events(upto 300 rules per event bus)
Rules:

- Send events to multiple targets
- Veent pattern or schedule
- Managed rules
- Iam policy

Events:

- Env change
- Schdduled events
- Json objects

Event schema can be deployed using openapi and jsonschema draft4
Create archive of events and can replay them later

See the events on ec2 autoscaling instances terminate or start and choose a target for it as lambda function(for logging these events etc..)

Amazon comprehend is a natural language processing
Amazon comprehend medical to identity complex medical information

- Built in models
- Custom models with endpoint

Amazon kendra is a search service with ml

- Create an index
- Add data sources like sharepointms3 etc..
- Test and deploy

It will crawl through all documents and create an index for searching like(in your company how many customers are there)

Amazon transcribe isa automatic speech recognition service(speech to text), has realtime trasncription, it also has medical side for the same

Amazon polly is a text to lifelike speech service

Amazon rekognition is a visual analysis service like images,videos

Amazon personalize is a personalization/recommendation service

Amazon forecast is used to deliver highly accurate forecasts based on our datasets(retail,custom,inventory pkanning,ec2 capacity,work force,web traffic,metrics)

Amazon textract is used to extract text and data from documents

Amazon panorama brings computer vision into on premcameras for local predictions

Amazon fraud detector(fraud in sign ups , transactions et..)

Amazon lex is a voice and text conversion addgin feature into an app(converts speech to text and uses natural language processing to recognize text intentions

Amazon connect records calls

Amazon keyspaces Is a apache cassandra service(serverless)

Amazon neptune is a graph database service(apache gremlin,sparql)

Amazon quantum ledger database is a managed ledger database(blockchain ledger)

Amazon timestream is a IOT db service(time series data)