International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 6 Issue 03, March-2017

Managing Infrastructure in Amazon using EC2, CloudWatch, EBS, IAM and CloudFront

Aishwarya Anand
Assistant Professor,
GD Goenka University
Sohna Road

***Abstract*— Amazon EC2 provides a centralized control over all the resources and an efficient computing environment. EC2 is very beneficial as it provides Auto-Scaling capability by which resources are scaled automatically when required without any friction. Amazon EC2 service level agreements commitment is 99.95%, which makes it highly reliable. AWS CloudWatch is a service provided by Amazon that basically reduces the time to detect and recover from issues. Amazon web services are mainly targeted at organizations that have multiple users or various systems that use AWS products such as EC2, Amazon S3 and others. IAM (Identity and access management) is used to manage all the users of AWS by having a central control, providing access keys, granting user permissions that control the access to various AWS resource.**

Keywords*— *Auto-scaling; centralized control; computing capabilities; simple service interface; service level agreement; Cloudwatch.

I. INTRODUCTION

The AWS management console provides a visual way to manage all the AWS tasks, such as working with Amazon storage buckets, launching EC2 instances, setting Amazon CloudWatch alarms and more. Each service has its own console that you can get to from the main console. The AWS Management Console also provides an easy way to get to information about your account and about billing. It's easy to use. You can sign in to your Amazon account and access management console and find all the services you want.

Amazon EC2 provides computing capabilities in the cloud. It has a very simple service interface through which user can obtain and configure capacity efficiently which ensures that the capacity is used in a predictable manner.

The EC2 instance is highly secure, it not only maintains the confidentiality of user data that is stored in the cloud but also places the user instances in a VPC (virtual private cloud) which has an IP range that is specified by the customer [2]. The customer can decide about the instances that he wants to keep private and the ones he wants to expose to the world. There are security groups that allow a customer to specify and control the inbound and the outbound traffic to and from his instance. Amazon EC2 is not very expensive as the customer pays on an hourly basis without any future commitment. It also provides Auto-Scaling, which lightens the customer from the burden of handling traffic spikes. Instances are available On-Demand and can also be reserved [2].

To use Amazon EC2, you simply: [2]

- Select a pre-configured, templated Amazon Machine Image (AMI) to get up and running immediately.
- Configure security and network access on your Amazon EC2 instance.
- Choose which instance type(s) you want, then start, terminate, and monitor as many instances of your AMI as needed.
- You can also attach persistent block storage to your instances.
- Pay only for the resources that you actually consume, like instance-hours or data transfer.

Amazon EC2 provides the following features: [5]

Instances: Virtual computing environments

AMI: Pre-configured templates for the instances, that package the bits you need for your server (including the operating system and additional software)

Instance Types: Various configurations of CPU, memory, storage, and networking capacity for your instances.

Key Pairs: Secure login information for your instances (AWS stores the public key, and you store the private key in a secure place)

Instance Store Volumes: Storage volumes for temporary data that's deleted when you stop or terminate your instance.

EBS volumes: Persistent storage volumes for your data using Amazon Elastic Block Store (Amazon EBS)

Regions and Availability zones: Multiple physical locations for your resources, such as instances and Amazon EBS volumes.

Security Groups: A firewall that enables you to specify the protocols, ports, and source IP ranges that can reach your instances.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 6 Issue 03, March-2017

Elastic IP addresses: Static IP addresses for dynamic cloud computing.

Tags: Metadata, that you can create and assign to your Amazon EC2 resources

Virtual private clouds (VPCs): Virtual networks that you can create, which are logically isolated from the rest of the AWS cloud, and that you can optionally connect to your own network.

II. AWS CLOUDWATCH

AWS CloudWatch is a service provided by Amazon, that collects measurements of your AWS resources so that you can easily access those measurements, receive timely notifications and more. So, it basically reduces the time to detect and recover from issues.
AWS CloudWatch enables you to monitor your AWS resources in real time without installing additional software. AWS resources that can be monitored in CloudWatch are: [11]

EC2 instance
EBS volume
Auto Scaling groups
Elastic Load Balancer and more

Cloud Watch is chosen because, it:

Lowers the operational overhead
Works extremely well with AWS as there is no configuration necessary and alarms can be set for different situations

You can get many things for free, like:

CPU stats
I/O stats and more

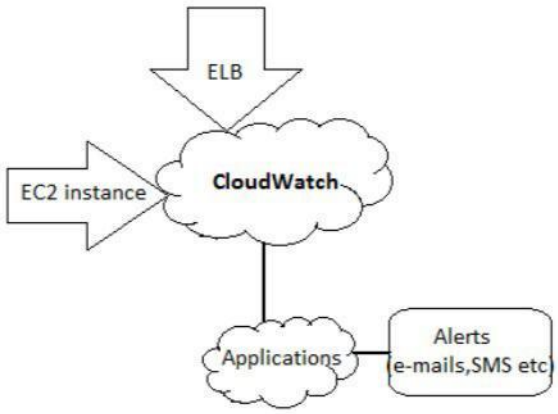

Fig 1: Basic CloudWatch Architecture

For e.g. you can use CloudWatch to monitor the CPU and disk reads of your EC2 instances and then use this data to determine whether you should launch additional instances to handle that load or not. [12]

Alarms

An alarm watches a single metric over a time period you specify, and performs one or more actions based on the value of the metric relative to a given threshold over a number of time periods. The action is a notification sent to an Amazon Simple Notification Service or Auto Scaling policy.
The following steps are followed:

- Choose a metric in CloudWatch
- Set a threshold value
- Assign an action to take when the threshold breaches. For e.g. Email, SMS and more.

III. EBS (AMAZON ELASTIC BLOCK STORE)

It is essentially a virtual hard drive i.e. storage volume that you can attach to your EC2 instance, it is: [6]

- Raw Storage
- Limits 1GB to 1000GB
- Attached to EC2 instance in an availability zone
- Billed on storage space and I/o operations(e.g. 10 cents per Gb)

Amazon Elastic Block Store (EBS) offers persistent storage for Amazon EC2 instances. It allows you to create storage volumes from 1 GB to 1 TB. You can create file systems on top of Amazon EBS volumes or use them in any other way (like hard disk). Amazon EBS volumes are network-attached, and persist independently from the life of an instance. Amazon EBS volumes are highly available, highly reliable volumes that can be attached to any running instance that is in the same Availability Zone, it is leveraged as an Amazon EC2 instance's boot partition or attached to a running Amazon EC2 instance as a standard block device. Amazon EBS volumes that are attached to an Amazon EC2 instance are exposed as storage volumes that persist independently from the life of the instance.

Amazon EBS volumes offer greatly improved durability over local Amazon EC2 instance stores, as Amazon EBS volumes are automatically replicated on the backend (in a single Availability Zone). It has an annual failure rate of 0.1% to 0.4%, which is 10 times less than normal hard drives.

For those wanting even more durability, Amazon EBS provides the ability to create point-in-time consistent snapshots of your volumes that are then stored in Amazon S3, and automatically replicated across multiple Availability Zones. These snapshots can be used as the starting point for new Amazon EBS volumes, and can protect your data for long-term durability. You can also easily share these snapshots with co-workers and other AWS developers. Amazon EBS provides two volume types: Standard volumes and Provisioned IOPS volumes. Standard volumes are designed for applications with moderate I/O requirements. They are also well suited for use as boot volumes or applications where I/O can be bursty. Provisioned IOPS volumes offer storage with consistent and low-latency performance, and are designed for applications with I/O-intensive workloads such as databases. [2]

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 6 Issue 03, March-2017

Amazon EBS volumes

An Amazon EBS volume is a durable, block-level storage device that you can attach to a single EC2 instance. You can use Amazon EBS volumes as primary storage for data that requires frequent updates, such as the system drive for an instance or storage for a database application. Amazon EBS volumes persist independently from the running life of an EC2 instance. After a volume is attached to an instance, you can use it like any other physical hard drive. [7]

Amazon EBS Snapshots

The process of moving an EBS volume to S3 is by taking a snapshot, which is used for back up and cloning. An Amazon EBS snapshot is a backup copy of an Amazon EBS volume that is stored in Amazon S3. Snapshots are incremental backups, which mean that only the blocks on the device that have changed after your most recent snapshot are saved. When you delete a snapshot, only the data exclusive to that snapshot is removed. Active snapshots contain all of the information needed to restore your data (from the time the snapshot was taken) to a new Amazon EBS volume. [7]

IV. AUTOSCALING

Auto Scaling grows and shrinks the number of instances attached to your application in case of an event. It is a very attractive service provided by Amazon which allows you to automatically scale your Amazon EC2 capacity up or down according to your requirement. With Auto Scaling, you can ensure that the number of Amazon EC2 instances you're using scales up seamlessly during demand spikes to maintain performance, and scales down automatically when the demand is low to minimize costs. Auto Scaling is particularly well suited for applications that experience variability in usage. [8]

You can use Auto Scaling to:

- Manage Amazon EC2 capacity automatically
- Maintain the right number of instances for your application
- Operate a healthy group of instances, and
- Scale instances according to your needs

Auto Scaling frees you from having to predict huge traffic spikes accurately and plan for provisioning resources in advance of them. With Auto Scaling, you can build a fully scalable and affordable infrastructure on the cloud. Consider a common web application scenario, you run multiple copies of your application simultaneously to serve the incoming customer traffic. These multiple copies of your application are hosted on identical Amazon EC2 instances each of which is handling customer requests. Auto Scaling manages the launch and termination of these EC2 instances on your behalf. [10]

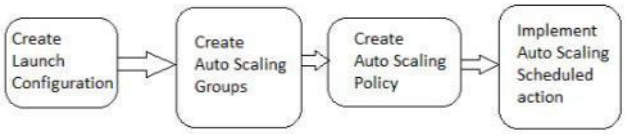

Fig 2: Implementing Auto Scaling [8]

When you use Auto Scaling, your EC2 instances are categorized into Auto Scaling groups. You create Auto Scaling groups by defining the minimum, maximum, and, optionally, the desired number of running EC2 instances the group must have at any point in time, for the purposes of instance scaling and management.

Your Auto Scaling group uses a launch configuration to launch EC2 instances. You create the launch configuration by providing information about the image you want Auto Scaling to use to launch EC2 instances i.e. the size of your image, instance type and more. In addition to creating a launch configuration and an Auto Scaling group, you must also create a scaling plan for your Auto Scaling group. A scaling plan tells Auto scaling when and how to scale. You can create a scaling plan based on the occurrence of specified conditions (dynamic scaling) or you can create a plan based on a specific schedule.

Auto Scaling helps you make efficient use of your compute resources by automatically doing the work of scaling for you. This automatic scaling is the core value of the Auto Scaling service

V. IAM (IDENTITY AND ACCESS MANAGEMENT)

Amazon web services are mainly targeted at organizations that have multiple users or various systems that use AWS products such as EC2, Amazon S3 and others. IAM is used to manage all the users of AWS by having a central control, providing access keys, granting user permissions that control the access to various AWS resource.

The basic entities in IAM that comprise AWS account are users and groups. Permissions are granted to these entities that enable them to access AWS resources. IAM manages users and their permissions by using IAM query API with which the customer can make direct calls to the IAM web service. Whenever a request is sent to AWS, it must include authentication information so that AWS can verify the authenticity of the user request. AWS uses this information to recreate the customer signature, which it compares with the one that is sent, if they match, the user is allowed to access AWS.

It supports GET and POST requests for all actions. GET requests are browser dependent and have a limitation of URL (Uniform resource locator) size, whereas POST does not have any such limitation and is used for URLs of larger size. IAM users can sign their requests with access key ID and a secret key or use the AWS security token service that generates temporary security credentials. These security credentials not only authenticate a user but also determine whether he has permission to access those resources [13].

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 6 Issue 03, March-2017

IAM is very beneficial, as it not only manages all the users and their permissions centrally but also provides security credentials for AWS account like access keys, user permissions and more to authenticate every user. However, without IAM, companies will have multiple AWS accounts and will have to manage them separately, every account will have different billing and there will be no controls over the tasks of a particular user. An organization can restrict its employees, AWS access, based on their job duties.

IAM also allows central control over all the data and thus no data is lost when an employee leaves the organization. IAM also allows control over the network by which the employees can access AWS resources only from the organization's corporate network using SSL (Secure sockets layer). It also lets you receive a single consolidated bill for multiple AWS accounts. An organization can have multiple AWS accounts out of which one account becomes the paying account that pays for its own charges and the charges of the linked AWS accounts.

IAM includes the following:

- Users – Create individual users
- Groups – Manage permissions with groups
- Permissions – Grant least privilege
- Password – Configure a strong password policy
- MFA – Enable MFA for privileged users
- Roles – Use IAM roles for EC2 instances
- Sharing – Use IAM roles to share access
- Rotate – Rotate security credentials regularly
- Conditions – Restrict privileged access further with conditions
- Root – Reduce/remove use of root

IAM creates multiple users and assign them different permissions. It also grants temporary access to some users by assigning them temporary security credentials. IAM also provides federated user access to those who do not have an AWS account.

IAM GROUP

IAM group is a collection of IAM users. It allows you to specify a set of permissions for different users in the group. An IAM user can belong to multiple groups and permissions can be assigned to new users by adding them to appropriate groups. When a user is moved to another department in the organization, you can just remove him from his old group and add him to his new group.

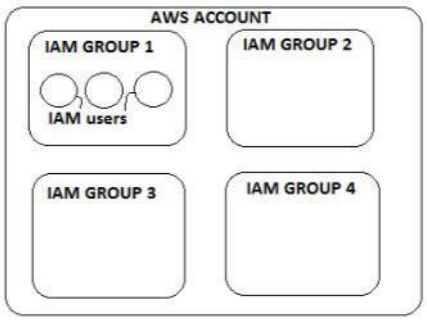

Fig 3: IAM Groups and Users

IAM USERS

IAM user is the entity that is created to interact with AWS. IAM users provide identity to various employees of an organization with which they can sign into their AWS management console and access all the services they are allowed to. IAM user is not always a person, it is an identity which is associated with some permissions. IAM users are provided with their username and password that enables them to uniquely identify themselves to AWS.

IAM username and password: An organization has a larger number of users that access the AWS account. AWS IAM lets you create multiple unique user identities for all the users that are accessing the account. This enables all the users in an organization to access the account with their unique ID and password. It also stratify its users by granting them different permissions like some users get administrative level permissions where as some get read only permissions and so on.

Security of user's credentials can be enhanced by enabling Multi-factor Authentication for the IAM users.

Multifactor authentication: AWS allows only authenticated users to access the AWS account with the help of a username and password. To provide an extra layer of security an authentication code is generated from a MFA (Multifactor authentication) device, this additional code needs to be entered by the user along with the username and password to sign into the AWS website. It can be used for both, the root account as well as IAM account.

IAM ROLES

A role lets you define a set of permissions to access the resources that a user or service needs, but these permissions are not attached to an IAM user or group. Instead, many applications and AWS services like Amazon EC2 can assume roles during runtime. When a role is assumed, AWS returns temporary security credentials that the user or application can use to make programmatic request to AWS. It temporarily delegates access to users or services that do not have access to AWS resources. Like, a user in an AWS account might need access to resources in another account.

When you create a role, you specify two policies, i.e.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 6 Issue 03, March-2017

Trust policy: It specifies, who is allowed to assume the role i.e. the principal.

Access policy: It defines what resources the principal is allowed to access.

IAM PERMISSIONS

There are a number of permissions that are granted to the customer that allows him to access one or more resources. These permissions are basically of two types: User-Based and Resource-Based.

- User-Based permissions specify the permissions associated with a specific user.
- Resource-Based permissions determine the entities that have access to particular resources.

IAM POLICIES

To assign permission to a user, group or resource, a policy is created. A policy is a document that explicitly lists permissions. It lets you specify the following:

- Action: The actions you will be allowed.
- Resources: The resources on which the action will be allowed.
- Effect: The effect when user requests access i.e. allow

or deny.

IAM policies control access regardless of the interface. For example, you could provide a user with a password to access the AWS Management Console, and the policies for that user would control what the user can do in the AWS Management console.

VI. CLOUDFRONT

Amazon CloudFront is a content delivery network that delivers web content with lowest latency. It quickly distributes static as well as dynamic web content like image files, html pages and more. CloudFront delivers the content through the various edge locations that Amazon has all around the world like in Chennai, Mumbai and more. It delivers contents immediately and provides best possible performance. [14] There is a time period that is defined, for which the content stays in the edge location. After the object has been in the edge location for the specified time, CloudFront retrieves it from an Amazon S3 bucket or a web server which is the source of that content when the next request for the object is made. It checks whether the version in the edge location is the latest, if so CloudFront delivers it to the user. If the version is not latest, then the source sends the latest version to CloudFront and CloudFront delivers the object to the user.

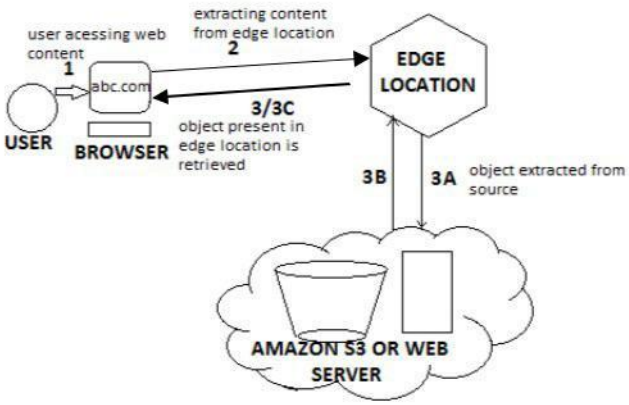

Fig 4: Delivering content using CloudFront

In the above figure a user access your website i.e. it requests for one or more objects. The request is then routed to the CloudFront edge location that can serve the user's request. If the CloudFront cache contains the latest version of the object that the user has requested, it returns it to the user. In case, the latest version is not present in the edge location, the edge location extracts the files from the source which can be an Amazon S3 bucket or a web server located at Amazon's data centre. This object is then sent to the user.

You create a CloudFront distribution, which tells CloudFront which origin servers to get your files from when users request the files through your web site or application. It basically contains all the configuration settings like, Amazon S3 bucket information, access control information, whether HTTPS is required to access the content, whether access logs are to be created and more.

Distributions are of two types:

Web Distributions: Web distributions are used to serve web content like html pages, image files and more over HTTP or HTTPS, on demand multimedia content, a live event like a concert in real time and more.

RTMP Distributions: RTMP distributions use Amazon S3 bucket as origin and streams media files using Adobe media server and Adobe real-time messaging protocol (RTMP)

An original server stores all your files like web pages, images and more these are known as objects. You can make objects in your Amazon S3 bucket visible to public such that anyone who is aware of the CloudFront URLs for the object can access them. You also have the option of keeping the objects private and controlling access to them.

CloudFront provides optional log files which inform about all the user requests that have been made. Whenever the end user makes a request for your application, a request is routed by CloudFront to the appropriate edge location. CloudFront writes data about each request to a log file. CloudFront periodically puts this log file in an Amazon S3 bucket that you specify.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 6 Issue 03, March-2017

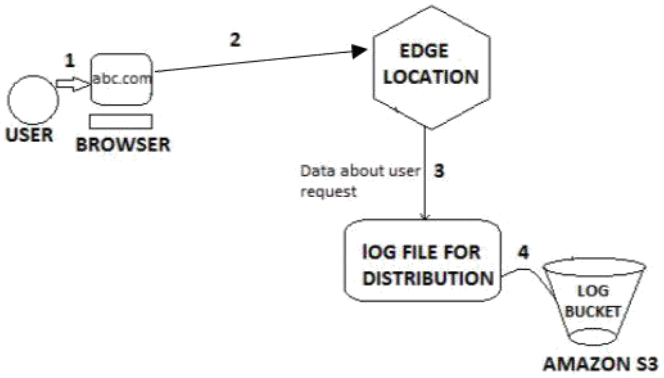

Fig 5: Access Logs In CloudFront

There are a number of organizations that distribute their content via internet and want to restrict access to various documents, media files and others. They can serve their private content securely using CloudFront i.e. users can use special signed URLs to access private content. You can restrict access to object in CloudFront edge caches as well as in Amazon S3 buckets. You can create signed URLs for your objects by specifying an ending date and time, after which the URL is not valid and IP address of computer that can access the content and more and distribute to the users. A part of the signed URL is hashed and signed using private key form the key pair.

You can also secure the content in Amazon S3 bucket by using CloudFront URLs for access as shown in the figure below. This prevents any user from bypassing CloudFront and using Amazon S3 URL to access the content.

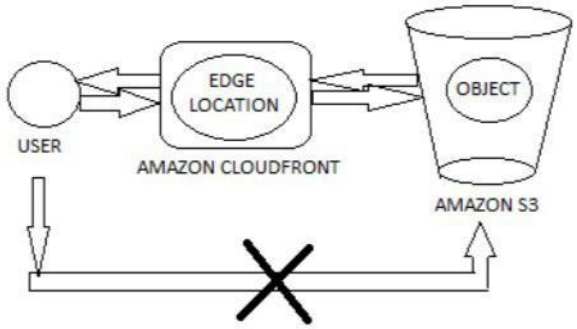

Fig 6: Secured access using CloudFront [14]

CONCLUSION

There are a number of cloud providers in the market today, including Amazon, Microsoft, AT&T, Cisco and many more. Out of these, our focus is on Amazon that is the most popular service provider today and has guaranteed security through its Amazon elastic compute cloud (EC2) architecture. The beauty of EC2 is that it provides capabilities like load balancing, Auto-Scaling and many more.

Amazon not only provides automated and standardized services to its customers but also supports different security schemes like IAM. Amazon maintains the confidentiality of the data by IAM, MFA and Access keys; it also maintains the integrity of the data by HMAC (Hash-Based message authentication code) and S3 server side encryption. Amazon is also known for its EC2's availability i.e. 99.95%. It has IAM that manages all the users and their access permissions. It centrally controls the users, their security credentials and their resources. There are different types of security credentials that are provided to the user, like access keys, X.509 certificate, MFA and many more.

REFERENCES

[1] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, James Broberg, Ivona Brandic: Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility *Future Generation Computer Systems*, Volume 25, Issue 6, Pages 599-616
[2] Aishwarya Anand, Infrastructure management in Amazon-EC2 instance, International Journal of Applied Engineering and Research (IJAER), Volume-10, Number 35, 2015.
[3] AWS documentation; Auto scaling, http://aws.amazon.com/autoscaling/
[4] AWS management console, Getting started guide (version 1.0), http://docs.aws.amazon.com/awsconsolehelpdocs/latest/gsg/getting-started.html
[5] AWS Elastic Compute Cloud User Guide (API Version 2014-02-01), http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/concepts.html
[6] CBT Nuggets, 07-AWS Storage-Elastic Block Storage (EBS)
[7] Amazon Elastic Compute Cloud, User Guide (API Version 2014-02-01), Amazon Elastic Block Storage, http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AmazonEBS.html
[8] CBT Nuggets, 06-AWS Elasticity- Components of Auto scaling
[9] Amazon Elastic Compute Cloud, User Guide (API Version 2014-02-01), Amazon Elastic Block Storage, http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AmazonEBS.html
[10] AWS Documentation, Auto Scaling Docs, Developer Guide,http://docs.aws.amazon.com/AutoScaling/latest/DeveloperGuide/WhatIsAutoScaling.html
[11] AWS re-invent RMG 203-"Cloud Infrastructure and application"
[12] Amazon CloudWatch, Developer Guide (API Version 2010-08-01),
[13] http://docs.aws.amazon.com/AmazonCloudWatch/latest/
[14] DeveloperGuide/WhatIsCloudWatch.html
[15] AWS Identity and Access Management, Using IAM, API Version2010-05-08, http://awsdocs.s3.amazonaws.com/IAM/latest/iam-ug.pdf
[16] Amazon CloudFront, Developer Guide (API Version 2014-01-31), http://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/Introduction.html