CHAPTER 3
METHODOLOGIES AND SYSTEM DESIGN

3.1. Foreword

This chapter covers the system design and methodologies for creating a Virtual Data Center (VDC) using AWS and Datadog. It emphasizes building a scalable, secure VDC, integrating cloud services, and applying best practices to ensure optimal performance and reliability. Additionally, it highlights the implementation of robust security measures to safeguard data, ensuring the integrity, availability, and resilience of the VDC against potential threats and vulnerabilities.

To configure a Virtual Data Center (VDC) on AWS, this system should use the following services:

- Identity and Access Management (IAM): For creating IAM users and roles and assign appropriate permissions using policies.
- Amazon VPC (Virtual Private Cloud): For creating a logically isolated network.
- Amazon EC2 (Elastic Compute Cloud): For scalable computing capacity.
- Amazon S3 (Simple Storage Service): For object storage.
- Amazon RDS (Relational Database Service): For managed relational databases.
- Amazon EBS (Elastic Block Store): For block storage.
- Elastic Load Balancing (ELB): For distributing incoming traffic.
- AWS Auto Scaling: For automatic scaling of resources.
- Datadog: For monitoring and logging.

3.2. AWS Identity and Access Management (IAM)

AWS Identity and Access Management (IAM) is a service that helps securely control access to AWS resources. It allows you to create and manage AWS users and groups and use permissions to allow and deny their access to AWS resources. IAM also

enables the implementation of security best practices, such as enforcing multi-factor authentication (MFA) and establishing fine-grained permissions to ensure only authorized users have access to specific resources. This service is critical for maintaining the security, governance, and compliance of cloud environments, thus ensuring organizational resilience and data integrity [10].

3.2.1. Key Features of IAM

- Granular Permissions: Define specific actions users can perform on AWS resources.
- User Management: Create and manage individual user accounts within your AWS account.
- Group Management: Create groups to manage permissions for multiple users at once.
- Roles: Create roles with specific permissions that can be assumed by users or services.
- Policies: Attach policies to users, groups, and roles to specify their permissions.
- Multi-Factor Authentication (MFA): Add an extra layer of security by requiring a second form of authentication.
- Secure Access: Provide secure access to AWS services and resources using access keys, passwords, and MFA.
- Federation: Allow users to access AWS resources using existing identity systems (e.g., Active Directory).
- Logging and Monitoring: Use AWS CloudTrail to monitor IAM operations and track changes.
- Least Privilege Principle: Apply the principle of least privilege by giving users the minimum permissions they need [10].

3.2.2. Benefits of IAM

- Enhanced Security: By managing permissions and applying MFA, IAM helps secure your AWS resources.
- Fine-Grained Access Control: Precise control over who can access which resources and how.

- Ease of Management: Simplifies the management of access for large teams through groups and roles.
- Scalability: Supports secure access management as your organization grows [10].

3.3. Amazon Virtual Private Cloud (VPC)

Amazon Virtual Private Cloud (VPC) allows to create a logically isolated section of the AWS cloud where launch AWS resources in a virtual network that define. It provides full control over virtual networking environment, including selection of IP address range, creation of subnets, and configuration of route tables and network gateways [7].

3.3.1. Key Features of VPC

- Subnets: Divide VPC's IP address range into multiple subnets, each residing in a specific Availability Zone.
- Route Tables: Control the routing of traffic within VPC, between subnets, and to/from the internet.
- Internet Gateway: Enable communication between instances in VPC and the internet.
- NAT Gateway: Allow instances in a private subnet to connect to the internet or other AWS services while preventing the internet from initiating connections with those instances.
- Elastic IP Addresses: Static IP addresses that can be associated with instances for consistent, reliable addressing.
- Network Access Control Lists (NACLs): Provide a stateless firewall to control inbound and outbound traffic at the subnet level.
- Security Groups: Provide a stateful firewall to control inbound and outbound traffic for instances.
- VPC Peering: Connect two VPCs privately using private IP addresses as if they were part of the same network.
- VPC Endpoints: Enable private connections between your VPC and supported AWS services without requiring an internet gateway or NAT device.

- VPN Connections: Establish secure connections between on-premises network and AWS VPC.
- AWS Direct Connect: Establish a dedicated network connection from premises to AWS.
- Flow Logs: Capture detailed information about the IP traffic going to and from network interfaces in VPC [7].

3.3.2. Benefits of VPC

- Enhanced Security: Control over IP addresses, subnets, and routing tables improves security.
- Customizability: Fully customizable network configurations to fit specific requirements.
- Scalability: Easily scalable to meet the needs of application workloads.
- Reliability: High availability and fault tolerance using multiple subnets and Availability Zones.
- Cost-Efficiency: Only pay for what you use, with the ability to optimize resource utilization [7].

3.4. Amazon Elastic Compute Cloud (EC2)

Amazon Elastic Compute Cloud (EC2) is a web service that provides resizable compute capacity in the cloud. It allows users to run virtual servers, known as instances, to host applications and perform various computing tasks. EC2 is designed to make web-scale cloud computing easier for developers by providing secure, resizable compute capacity on demand. EC2 provides a flexible platform where users can select instance types that best fit their application needs, ensuring cost-efficiency and optimal performance [2].

3.4.1. Key Features of EC2

- Scalability: Easily scale instances up or down based on demand.
- Variety of Instance Types: Offers a wide range of instance types optimized for different workloads, including general-purpose, compute-optimized, memory-optimized, storage-optimized, and GPU instances.

- Elastic Load Balancing: Automatically distributes incoming application traffic across multiple instances to ensure high availability and reliability.
- Auto Scaling: Automatically adjusts the number of EC2 instances to maintain performance and manage costs.
- Elastic Block Store (EBS): Provides persistent block storage volumes for use with EC2 instances.
- Security: Offers various security features, including security groups, network access control lists (NACLs), and key pairs for secure SSH access.
- Flexible Pricing Options: Includes On-Demand, Reserved, and Spot instances to optimize costs based on usage patterns.
- Integration with Other AWS Services: Seamlessly integrates with services like S3, RDS, IAM, VPC, CloudWatch, and more [2].

3.4.2. Benefits of EC2

- Cost-Effective: Pay only for the compute capacity you actually use.
- High Availability: Built on a reliable infrastructure across multiple Availability Zones.
- Flexibility: Choose from a wide variety of instance types and configurations to match your workload requirements.
- Control: Full control over your instances, including root access and the ability to stop, start, and terminate them [2].

3.4.3. EC2 Instance Types and Use Cases

Table 3.1. EC2 Instance Types and Use Cases [2]

Instance Families	Use cases
General Purpose Eg. A1, T2, T3, T3a, T4a, M6g, M7g, …	• Low-traffic websites and web application • Small database and midzide databases
Compute optimized Eg. C5, C6g, C6a, C7g, ….	• High-performance web servers • Video encoding
Memory optimized Eg. R7g, R6g, X2gd, X2i, z1d,…	• High-performance databases • Distributed memory caches
Storage optimized Eg. I4g, I3, D3, H1, …	• Data warehousing • Log or data processing applications
Accelerated computing Eg. P5, P4, Inf2, G6, G5g,Trn1, …..	• 3D visualizations • Machine learning

3.5. Amazon Simple Storage Service (S3)

Amazon Simple Storage Service (S3) is a scalable, high-speed, web-based cloud storage service designed for online backup and archiving of data and applications. It offers a simple web services interface to store and retrieve any amount of data at any time from anywhere on the web [3].

3.5.1. Key Features of Amazon S3

- Scalability: Store and retrieve unlimited amounts of data.
- Durability: Data is automatically distributed across a minimum of three physical facilities.
- Security: Supports data encryption and integrates with AWS Identity and Access Management (IAM) for secure access.
- Cost-Effectiveness: Pay only for the storage you use, with no minimum fee.
- Accessibility: Access data from anywhere via the web [3].
- Integration: Integrates with other AWS services like EC2, RDS, and Lambda.

3.6. Amazon Relational Database Service (RDS)

Amazon Relational Database Service (RDS) is a managed database service that simplifies setting up, operating, and scaling relational databases in the cloud. It automates time-consuming administrative tasks such as hardware provisioning, database configuration, patching, and backups, facilitating efficient management and allowing you to focus on your applications. RDS supports multiple database engines, offering flexibility and compatibility with various applications and enabling seamless integration with other AWS services [11].

3.6.1. Key Features of Amazon RDS

- Automated Backups: Regular backups and point-in-time recovery.
- Scalability: Easily scale database compute and storage resources with just a few clicks.
- High Availability: Multi-AZ deployments for automatic failover and increased availability.
- Security: Data encryption at rest and in transit, and integration with AWS IAM for access control.

- Managed updates and maintenance: AWS handles routine database management tasks.
- Database Engines: Supports multiple database engines including MySQL, PostgreSQL, MariaDB, Oracle, SQL Server, and Amazon Aurora [11].

3.7. Amazon Elastic Block Store (EBS)

Amazon Elastic Block Store (EBS) is a cloud-based storage service that provides persistent block storage for use with Amazon EC2 instances. It offers high performance and reliability for critical applications and supports a wide range of workloads [3].

3.7.1. Key Features of Amazon EBS

- Persistent Storage: Data persists independently of the life of the instance.
- High Performance: Designed for applications that require high IOPS (Input/Output Operations Per Second).
- Scalability: Easily scale storage size up or down.
- Snapshots: Create point-in-time snapshots of volumes for backup or replication.
- Encryption: Provides encryption for data at rest and in transit.
- Flexibility: Offers a variety of volume types, such as General Purpose (SSD), Provisioned IOPS (SSD), Throughput Optimized (HDD), and Cold (HDD), to suit different performance needs [3].

3.8. Elastic Load Balancing (ElB)

Elastic Load Balancing (ELB) is a service that automatically distributes incoming application traffic across multiple targets, such as Amazon EC2 instances, containers, and IP addresses, in one or more Availability Zones. It helps ensure high availability, fault tolerance, and scalability of your applications, optimizing resource utilization and enhancing performance [21].

3.8.1. Key Features of Elastic Load Balancing

- Automatic Traffic Distribution: Balances incoming traffic across multiple targets to ensure no single server is overwhelmed.

- High Availability: Spreads traffic across multiple Availability Zones for increased reliability.
- Health Monitoring:Continuously monitors the health of registered targets and only routes traffic to healthy instances.
- Security: Integrates with AWS Identity and Access Management (IAM), SSL/TLS termination, and encryption.
- Three Types of Load Balancers
    - Application Load Balancer (ALB): For HTTP/HTTPS traffic, providing advanced routing features.
    - Network Load Balancer (NLB): For TCP/UDP traffic, offering ultra-high performance and static IP addresses.
    - Classic Load Balancer (CLB): For applications built within the EC2-Classic network [21].

3.9. Amazon CloudWatch

Amazon CloudWatch is a comprehensive monitoring and management service for AWS cloud resources and applications. It provides data and actionable insights to systematically monitor your applications, understand and respond to system-wide performance fluctuations, optimize resource utilization, and obtain a unified view of operational health, facilitating data-driven decision-making and ensuring robust system integrity [5].

3.9.1. Key Features of Amazon CloudWatch

- Metrics Collection: Collects and tracks metrics, logs, and events from AWS resources and custom sources.
- Alarms: Set alarms to automatically initiate actions based on specified thresholds.
- Dashboards: Create customizable dashboards to visualize your cloud resources and applications in real-time.
- Logs Monitoring: Collect, monitor, and analyze log files from EC2 instances, CloudTrail, and other sources.
- Event Notifications: Receive notifications based on event patterns and set up automated responses.

- Automated Actions: Automatically react to changes in your AWS environment by triggering automated responses [5].

3.10. AWS Auto Scaling

AWS Auto Scaling is a service that automatically adjusts the number of Amazon EC2 instances or other resources in a group based on the current demand. It helps ensure that you have the right amount of resources available to handle the load for your application, thereby optimizing performance and cost. This service helps maintain application availability by automatically scaling your resources in response to varying traffic patterns, ensuring that you only pay for what you need [21].

3.9.1. Key Features of Auto Scaling

- Dynamic Scaling: Automatically adjusts resource capacity in response to changing demand.
- Scheduled Scaling: Scales resources based on a schedule to anticipate predictable changes in demand.
- Predictive Scaling: Uses machine learning to predict future traffic and scale resources accordingly.
- High Availability: Ensures applications are available by maintaining a specified number of running instances.
- Cost Efficiency: Optimizes costs by only using the resources needed to handle current workloads [21].

3.11. AWS Lambda

AWS Lambda is a serverless computing service that enables you to execute code without provisioning or managing servers. It automatically triggers and runs your code in response to events, such as modifications to data in an S3 bucket or updates to a DynamoDB table. With AWS Lambda, you are charged solely for the compute time you consume, and there are no costs incurred when your code is idle, ensuring cost-efficiency and scalability in dynamic computing environments [6].

3.11.1. Key Features of AWS Lambda

- Serverless Execution: Run code without managing servers.

- Event-Driven: Execute code in response to events from AWS services or HTTP requests via Amazon API Gateway.
- Automatic Scaling: Automatically scales based on the number of incoming requests.
- Pay-per-Use: Charge only for the compute time your code consumes.
- Integrated Security: Integrates with AWS Identity and Access Management (IAM) for secure access control [6].

3.12. Datadog Monitoring Framework

The Datadog Monitoring Framework is a cloud-based platform designed for IT infrastructure, operations, and development teams. It offers comprehensive monitoring and analytics tools that track the performance of applications, servers, databases, and services. By integrating data from various sources, Datadog provides real-time insights and alerts, helping to maintain system performance and reliability. Furthermore, its advanced data correlation and visualization capabilities facilitate in-depth analysis and informed decision-making, enhancing operational efficiency and strategic planning, thereby optimizing overall system performance [1].

3.12.1. Key Features of Datadog

- Real-time Monitoring: Collects and visualizes metrics and logs from various sources in real-time.
- Integrated Dashboards: Provides customizable dashboards to monitor metrics across your infrastructure.
- Alerting and Notifications: Set up alerts to notify teams of performance issues and anomalies.
- APM (Application Performance Monitoring): Tracks application performance and helps diagnose bottlenecks.
- Log Management: Aggregates, searches, and analyzes log data.
- Security Monitoring: Detects and responds to security threats in real-time
- Resource Utilization Analysis: Evaluates and reports on the usage of computing resources .
- Automated Scaling: Adjusts resource allocation dynamically based on predefined performance thresholds and metrics [1].

3.13. System Design Flow Charts

3.13.1. Design Phase

Figure 3.1 shows design phase of the virtual data center.

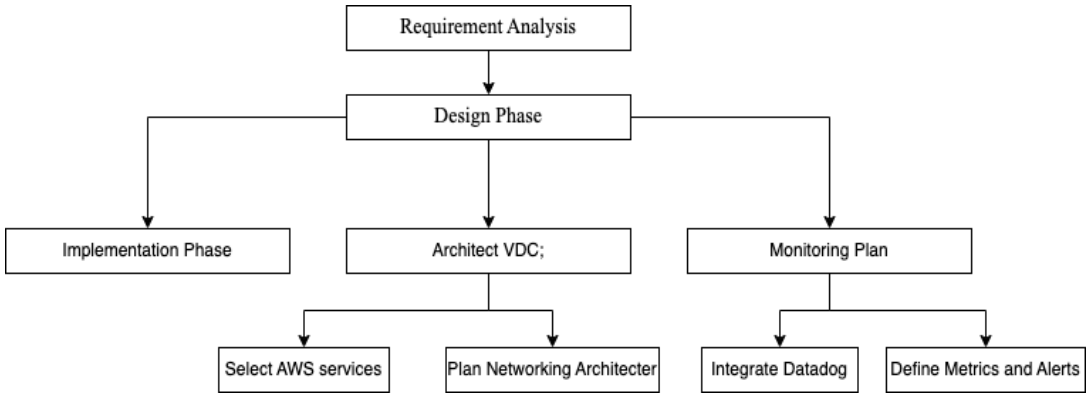

Figure 3.1. Design Phase

3.13.2. Implementation Phase

Figure 3.2 presents implementation phase of the virtual data center.

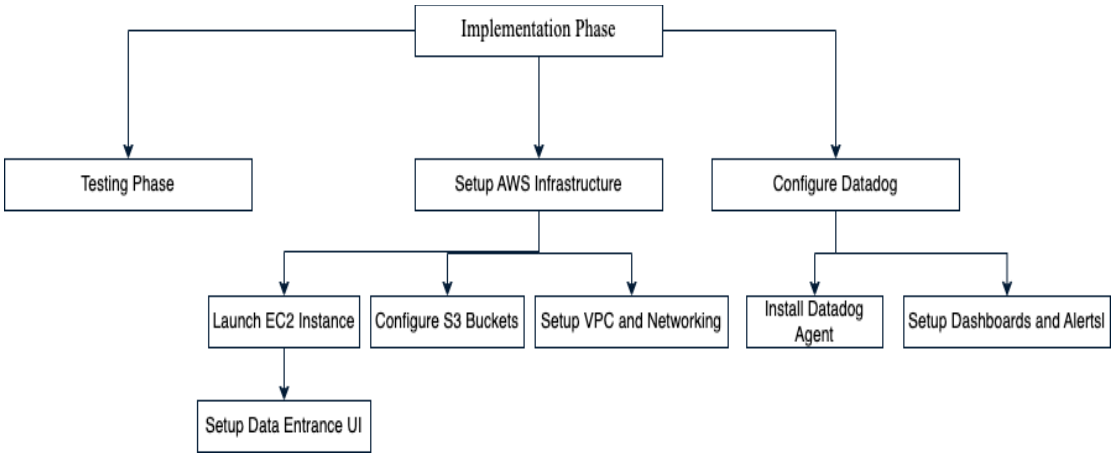

Figure 3.2. Implementation Phase

3.13.3. Testing Phase

Figure 3.2 demonstrats testing phase of virtual data center.

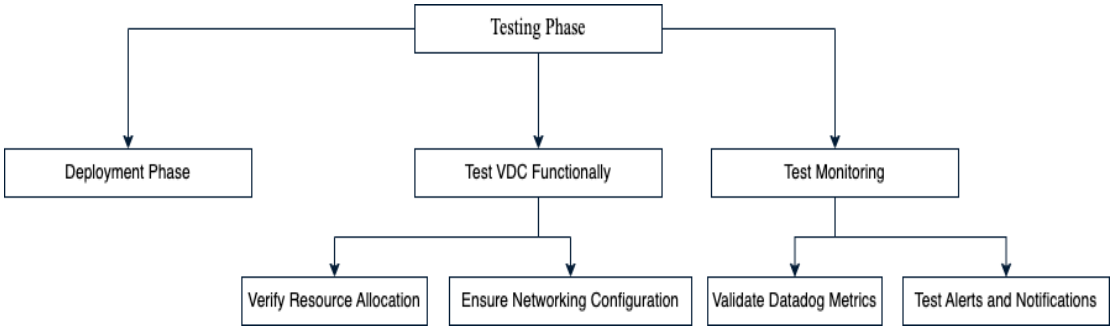

Figure 3.3. Testing Phase

3.13.4. Deployment Phase

Figure 3.4 shows deployment phase of the virtual data center.

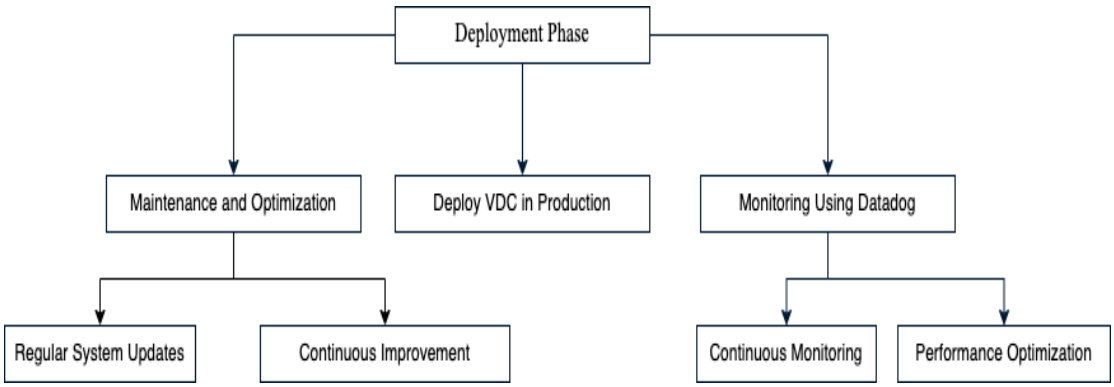

Figure 3.4. Deployment Phase

3.13.5. Step-by-Step Guide to Creating a Virtual Data Center on AWS

Figure 3.5 demonstrates step-by-step guite to creating virtual data center on aws.

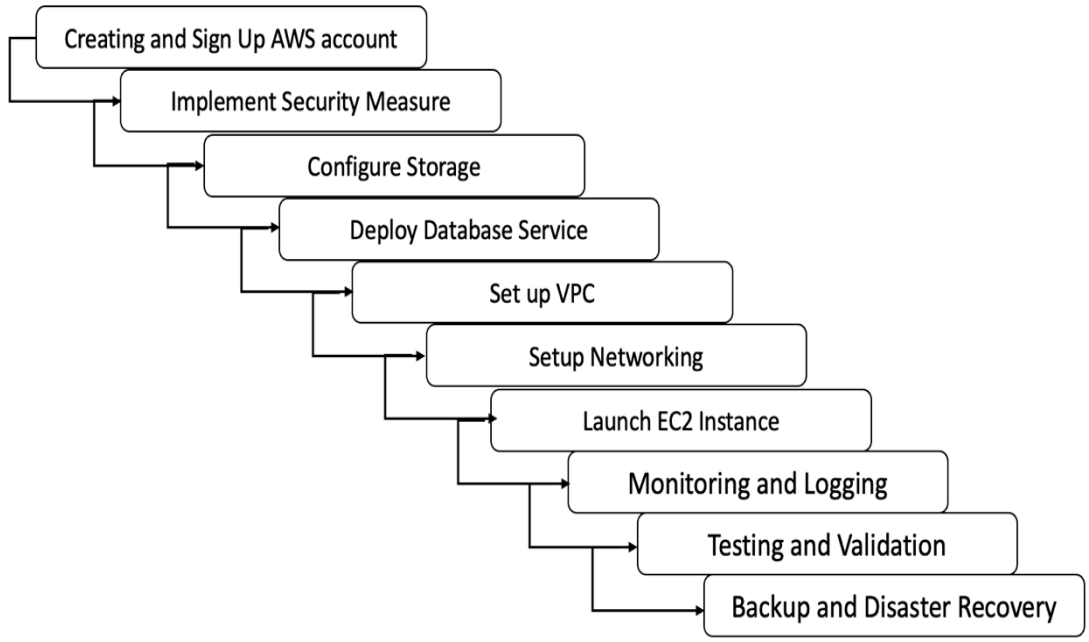

Figure 3.5. Guide to Creating a VDC on AWS