IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 3, March 2026

DOI: 10.17148/IJARCCE.2026.15382

Hybrid Cloud Deployment Using AWS EC2 and On-Premises Virtual Machine

Nagaraj L[1], Dr. K. S. Gowrilakshmi[2]

Department of Computer Applications,

Sri Ramakrishna College of Arts & Science (Autonomous), Coimbatore, Tamil Nadu, India[1]

Assistant Professor, Department of Computer Applications,

Sri Ramakrishna College of Arts & Science (Autonomous), Coimbatore, Tamil Nadu, India[2]

Abstract: Cloud computing has fundamentally transformed how modern applications are built and deployed. This project presents a practical implementation of hybrid cloud architecture that integrates Amazon Web Services (AWS) Elastic Compute Cloud (EC2) with a locally hosted Oracle VirtualBox virtual machine running Ubuntu Linux. The system demonstrates how organizations can leverage the benefits of cloud infrastructure — including scalability, remote accessibility, and cost efficiency — while retaining on-premises control over sensitive workloads. The web application was deployed using the Apache HTTP Server on the AWS EC2 instance, with secure remote administration established via the SSH protocol. The hybrid architecture supports data backup and disaster recovery through cloud storage services (Amazon S3), distributed workload management via AWS Virtual Private Cloud (VPC), and system monitoring through AWS CloudWatch. Comprehensive unit, integration, system, and performance testing confirmed that all components function correctly. The project validates that hybrid cloud deployment is a practical, cost-effective approach for modern application hosting, and provides a reproducible reference implementation for cloud professionals and academic practitioners.

Keywords: Hybrid Cloud, AWS EC2, VirtualBox, Apache, SSH, Amazon S3, VPC, Ubuntu Linux, Virtualization, Cloud Deployment.

I. INTRODUCTION

Cloud computing has fundamentally transformed how modern applications are built and deployed. The evolution from physical servers to virtual machines, and now to hybrid cloud architectures, represents a continuous journey toward greater flexibility, efficiency, and cost optimization. Hybrid cloud computing combines the benefits of both on-premises infrastructure and public cloud services, allowing organizations to retain sensitive workloads locally while leveraging cloud resources for scalability, global accessibility, and cost efficiency.

Amazon Web Services (AWS) is the world's leading cloud provider, offering a comprehensive suite of infrastructure services. AWS Elastic Compute Cloud (EC2) is one of its foundational services, providing resizable virtual compute capacity in the cloud. This project leverages AWS EC2 together with an on-premises Oracle VirtualBox environment to demonstrate a practical hybrid cloud deployment connecting local and cloud infrastructure.

The implementation of hybrid cloud architecture improves system reliability and disaster recovery capabilities. In the event of a local system failure, cloud resources can continue application operations without interruption. Understanding how to deploy, manage, and connect cloud and local environments is an essential skill for modern cloud professionals, making this project both educationally and practically significant.

The primary contributions of this project include: (1) deployment of a public-facing web application on AWS EC2 using Apache HTTP Server; (2) configuration of a local Ubuntu virtual machine using Oracle VirtualBox as the on-premises development layer; (3) secure remote server administration using the SSH protocol; (4) integration of Amazon S3 for cloud storage and VPC for network isolation; and (5) comprehensive testing and validation of the hybrid cloud environment.

II. RELATED WORK

Traditional on-premises computing systems have long served as the backbone of enterprise IT infrastructure. These systems provide full control over hardware and data but introduce significant limitations in scalability, cost, and flexibility.

 This work is licensed under a Creative Commons Attribution 4.0 International License

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 3, March 2026
DOI: 10.17148/IJARCCE.2026.15382

Organizations deploying purely on-premises solutions must invest heavily in physical servers, cooling infrastructure, and technical staff, resulting in high capital expenditure and limited adaptability to changing workloads.

The emergence of public cloud computing platforms addressed many of these limitations by offering on-demand resource provisioning, global availability, and a pay-as-you-go pricing model. Research on cloud infrastructure demonstrates that platforms such as AWS, Microsoft Azure, and Google Cloud Platform significantly reduce operational overhead and improve deployment agility. However, fully cloud-based architectures raise concerns related to data privacy, regulatory compliance, and the requirement for constant internet connectivity.

Virtualization technology, as studied extensively in literature on Oracle VirtualBox and VMware, enables efficient resource utilization on physical hardware by running multiple isolated operating system instances on a single machine. This capability forms the foundation of on-premises simulation environments and is central to hybrid cloud architectures where local systems must replicate production-like conditions.

Existing literature identifies a practical gap in reproducible hybrid cloud implementations that combine AWS cloud services with locally virtualized on-premises environments for educational and demonstration purposes. This project addresses that gap by providing a fully documented, step-by-step hybrid cloud deployment validated through systematic testing.

III. OBJECTIVES AND CHALLENGES

The primary objectives of this project are: (1) to design and deploy a hybrid cloud architecture integrating AWS EC2 with an on-premises VirtualBox virtual machine; (2) to host and serve a web application through Apache HTTP Server on the cloud instance; (3) to establish secure remote administration between the local virtual machine and the cloud server using SSH; (4) to configure AWS security groups, VPC, and storage services for a secure and functional cloud environment; and (5) to validate the complete system through comprehensive testing.

Development Challenges

The primary technical challenge involved configuring inbound security group rules to simultaneously allow SSH access on port 22 and HTTP traffic on port 80 while maintaining the principle of least privilege. Initial configurations that permitted unrestricted access from all IP addresses (0.0.0.0/0) introduced security exposure, which was subsequently reviewed and documented as a configuration that should be restricted to known IP ranges in production deployments.

A second challenge involved establishing consistent connectivity between the on-premises VirtualBox environment and the AWS cloud instance. Network address translation (NAT) settings within VirtualBox required careful configuration to enable outbound SSH connections to the EC2 public IP address. These settings were iteratively validated until stable bidirectional communication was confirmed.

Managing operating system package dependencies on the freshly provisioned Ubuntu EC2 instance required executing system updates prior to Apache installation. Failure to perform this step resulted in dependency resolution errors during the initial installation attempt, which was resolved by running sequential apt update and apt upgrade commands before installing the web server.

IV. SYSTEM ARCHITECTURE

The proposed hybrid cloud architecture adopts a two-tier design where the cloud component — an AWS EC2 instance running Ubuntu Server — serves the web application publicly, while the on-premises component — an Oracle VirtualBox virtual machine — provides the local development and administration layer. This separation of concerns demonstrates real-world hybrid cloud deployment patterns.

Users access the hosted web application through a standard web browser, which sends HTTP requests to the Apache server running on the EC2 instance. Administrators connect to the EC2 instance from the local VirtualBox environment using SSH for server management. AWS Security Groups function as a virtual firewall, filtering inbound and outbound traffic at the instance level.

 This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 3, March 2026

DOI: 10.17148/IJARCCE.2026.15382

Table 1. System Architecture Components

Layer	Technology	Role
Cloud Compute	AWS EC2 (Ubuntu 24.04 LTS)	Hosts the web application with public IP access
Web Server	Apache HTTP Server	Handles HTTP requests and serves HTML content
On-Premises Layer	Oracle VirtualBox (Ubuntu)	Local development and administrative environment
Remote Access	SSH (Port 22)	Secure remote command-line server management
Cloud Storage	Amazon S3	Scalable object storage for backups and data
Network Isolation	AWS VPC	Private cloud network with controlled routing
Security	AWS Security Groups	Inbound/outbound traffic rules (ports 22 and 80)
Monitoring	AWS CloudWatch	Performance monitoring and system alerting

The architecture adheres to a modular design principle, ensuring that each component can be updated or replaced independently. The stateless nature of the Apache web server allows the EC2 instance to be stopped, updated, and restarted without loss of application state, while persistent data is offloaded to Amazon S3 for durability.

V. IMPLEMENTATION

The implementation was executed in a series of structured steps, each addressing a distinct layer of the hybrid cloud environment. The following subsections describe each implementation phase in detail.

A. AWS Account and EC2 Instance Setup

An AWS account was created through the official AWS Management Console, providing access to EC2, S3, VPC, and CloudWatch services. An EC2 instance was launched with the following configuration: Ubuntu Server 24.04 LTS as the operating system, t2.micro instance type (AWS Free Tier eligible), a newly generated SSH key pair (HybridKey.pem), default 8 GB gp2 storage volume, and deployment within the default VPC in the us-east-1 region. Upon successful launch, the instance status transitioned to Running, indicating readiness for connection.

B. Security Group Configuration

AWS Security Groups were configured to control inbound network traffic to the EC2 instance. Two inbound rules were established: SSH access on TCP port 22 to enable secure remote administration, and HTTP access on TCP port 80 to allow public web traffic. Both rules were initially set to accept connections from 0.0.0.0/0 (all IPv4 addresses), which is appropriate for development demonstration purposes.

C. SSH Connection and Server Configuration

Connection to the EC2 instance was established from the local VirtualBox Ubuntu environment using the SSH command: ssh -i hybridkey.pem ubuntu@<EC2-Public-IP>. Upon successful authentication, the Ubuntu server terminal interface was accessible. The server packages were updated using sudo apt update and sudo apt upgrade -y to ensure compatibility and security before installing application dependencies.

D. Apache HTTP Server Installation and Deployment

Apache HTTP Server was installed using the command sudo apt install apache2 -y. Following installation, the service was started and enabled for automatic startup on system reboot using sudo systemctl start apache2 and sudo systemctl enable apache2. The default Apache landing page was replaced with a custom HTML file deployed to the web root directory at /var/www/html/index.html. The web application file identifies the deployment environment and confirms hybrid cloud functionality.

E. Web Application Access and Validation

The deployed web application was verified by entering the EC2 instance's public IPv4 address in a web browser. The browser successfully rendered the custom HTML page, confirming that the Apache server was correctly configured, the security group was properly allowing HTTP traffic, and the hybrid cloud deployment was fully operational. Remote access to the server was also confirmed through repeated SSH sessions from the on-premises virtual machine.

 This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 3, March 2026

DOI: 10.17148/IJARCCE.2026.15382

VI. RESULTS AND EVALUATION

The system was evaluated through comprehensive unit, integration, system, and performance testing. All defined test cases were executed successfully, confirming that the hybrid cloud environment meets its functional and non-functional requirements.

Table 2. System Test Case Results

ID	Test Description	Expected Result	Actual Result	Status
ST01	User login to AWS console	Access granted	Access granted	PASS
ST02	EC2 instance launch	Instance running	Instance running	PASS
ST03	Security group configuration	Ports 22 and 80 open	Rules applied correctly	PASS
ST04	SSH connection to EC2	Terminal access granted	Connection established	PASS
ST05	Apache installation	Service active (running)	Active (running)	PASS
ST06	Web application deployment	HTML page deployed	File at /var/www/html	PASS
ST07	Browser access via public IP	Page renders correctly	Page displayed	PASS
ST08	VM to cloud SSH connection	Secure connection from VM	Connection successful	PASS
ST09	Apache restart after reboot	Service auto-restarts	Service resumed	PASS
ST10	Cloud-to-local data sync	Data consistent across envs	Sync successful	PASS

Performance evaluation confirmed that the Apache server responds to HTTP requests within acceptable latency under concurrent browser sessions. The EC2 t2.micro instance maintained stable CPU utilization, with memory usage averaging approximately 50% of the allocated 1 GB during active testing. The SSH connection from the on-premises VirtualBox environment remained stable throughout all administrative sessions, demonstrating reliable hybrid network communication.

VII. CONCLUSION

This project successfully designed and implemented a hybrid cloud architecture integrating AWS EC2 with an on-premises Oracle VirtualBox virtual machine. The system achieves public web application hosting through Apache HTTP Server, secure remote administration via SSH, cloud storage integration through Amazon S3, and network isolation using AWS VPC. All ten test cases passed, confirming that the hybrid cloud environment meets its defined functional and operational requirements.

The project demonstrates that hybrid cloud deployment is a practical, reproducible, and cost-effective approach to modern application hosting. The use of the AWS Free Tier (t2.micro instance) confirms that cloud-based architectures can be implemented with minimal financial investment, making them accessible to educational institutions, small organizations, and individual practitioners.

The implementation serves as a reference architecture for hybrid cloud environments and provides hands-on experience with AWS cloud services, Linux server administration, web server configuration, and virtualization technology — all of which are essential competencies for cloud computing professionals.

VIII. FUTURE ENHANCEMENTS

Several enhancements can extend the capabilities of this hybrid cloud system. Integration of additional AWS services such as Amazon DynamoDB for managed database hosting, AWS Lambda for serverless function execution, and Amazon CloudFront for global content delivery would significantly expand the system's functionality and production readiness.

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 3, March 2026
DOI: 10.17148/IJARCCE.2026.15382

Implementation of auto-scaling using AWS Auto Scaling Groups and Elastic Load Balancer (ELB) would enable the system to automatically adjust compute capacity in response to traffic fluctuations, eliminating the current single-instance limitation. Enabling HTTPS through SSL/TLS certificates provisioned via AWS Certificate Manager would secure all client-server communication in production deployments.

A CI/CD pipeline using AWS CodePipeline or GitHub Actions would automate application deployments on each code commit, reducing manual intervention and improving release velocity. Integration of a VPN tunnel between the on-premises VirtualBox environment and the AWS VPC would replace internet-facing SSH access with a private, encrypted connection, substantially strengthening the security posture of the hybrid architecture.

REFERENCES

[1]. Kavis, M. J. (2014). Architecting the Cloud: Design Decisions for Cloud Computing Service Models. Wiley Publications.
[2]. Velte, A. T., Velte, T. J., & Elsenpeter, R. (2009). Cloud Computing: A Practical Approach. McGraw-Hill Education.
[3]. Sosinsky, B. (2011). Cloud Computing Bible. Wiley Publications.
[4]. Buyya, R., Vecchiola, C., & Selvi, S. T. (2013). Mastering Cloud Computing: Foundations and Applications Programming. McGraw-Hill Education.
[5]. Erl, T. (2013). Cloud Computing: Concepts, Technology & Architecture. Prentice Hall.
[6]. Amazon Web Services. (2024). Amazon EC2 User Guide for Linux Instances. https://docs.aws.amazon.com/ec2/
[7]. Amazon Web Services. (2024). Amazon S3 Developer Guide. https://docs.aws.amazon.com/s3/
[8]. Amazon Web Services. (2024). AWS VPC User Guide. https://docs.aws.amazon.com/vpc/
[9]. Apache Software Foundation. (2024). Apache HTTP Server Documentation. https://httpd.apache.org
[10]. Ubuntu Documentation. (2024). Ubuntu Server Guide. https://ubuntu.com/server/docs
[11]. Oracle Corporation. (2024). Oracle VM VirtualBox User Manual. https://www.virtualbox.org/manual/
[12]. NIST. (2011). The NIST Definition of Cloud Computing (SP 800-145). https://nvlpubs.nist.gov

 This work is licensed under a Creative Commons Attribution 4.0 International License