Journal of Modern Computing and Engineering Research
JMCER
Volume 2026, p. 01-11
https://jmcer.org

IMS Integration and Security with OpenStack Authentication Mechanisms

Abdallah Handoura
Computer department, Faculty of engineering and technology, Muscat University, Sultanate of Oman
Computer Department, IMT Atlantique France
ahandoura@muscatuniversity.edu.om

Received: January5, 2026 Revised: Feb 12, 2026 Accepted: March 10, 2026

Abstract:
Next Generation Networks (NGN) are packet-switched telecommunications architectures that rely on IP technology to support a broad range of services. In contrast, the IP Multimedia Subsystem (IMS) functions as the control and service framework that allows multimedia applications to operate over NGN. In simple terms, NGN provides the underlying network infrastructure, while IMS acts as the platform within that infrastructure that manages multimedia services such as voice and video communication for end users. Ensuring security for users, services, and providers is a critical priority. The increasing complexity of cyber threats, the wide variety of services, and the extensive use of distributed cloud environments make comprehensive security more important than ever. This paper proposes an integrated security mechanism aimed at strengthening protection for users, services, and providers within the IMS framework by tightly integrating IMS architecture with cloud computing environments. By leveraging multiple advanced security features, layered protection strategies, and control mechanisms inherent in cloud computing, the proposed approach enhances the system's overall reliability, confidentiality, and resilience. This paper presents an integration between two fields: telecommunication services and network concepts along with open technologies, aiming to provide an open service creation framework and to leverage the security mechanisms offered by cloud-based authentication, but not discuss the impact of the security mechanisms implementation to the QoS, latency, throughput.

Keywords: *IMS, Cloud Computing, OpenStack, Security, SIP.*

1 Introduction and Motivation

NGN (Next Generation Networks) support a broad spectrum of services, ranging from interactive multimedia applications such as audio and video telephony, conferencing, and instant messaging, to non-interactive services including push-based applications and multimedia streaming, as well as web-based services like e-commerce, e-learning, and e-health. One of the key benefits introduced by NGN is the shift from vertically integrated networks where data, voice, and video services are handled separately to a horizontally integrated architecture that relies on a unified network infrastructure.
The IP Multimedia Subsystem (IMS), developed by the 3GPP as an open, IP-based framework, (Dodd and al 2015, Mojka and al 2010), employs the Session Initiation Protocol (SIP),(Handoura 2018) as its signaling mechanism to enable service convergence across heterogeneous networks. IMS realizes the service convergence paradigm by allowing service providers to deliver advanced NGN services efficiently. As illustrated in Figure 1, IMS adopts a horizontally integrated architecture in which the service layer is clearly separated from the control layer.
Through IMS, service providers can independently design their service layers to support the development of innovative applications by combining reusable service components. IMS supplies standardized and fundamental service building blocks that can be shared across multiple application servers, enabling service reuse and significantly reducing time-to-market.
Cloud computing represents one of the most advanced paradigms in modern computing, offering cost-efficient data processing, high flexibility, and scalable resource management. Cloud platforms

Corresponding author:ahandoura@muscatuniversity.edu.om **ISSN: 2960-141X E-ISSN: 2958-0021**

deliver virtualized resources over the Internet through three main service models: Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). These models support various applications, including multimedia services, VoIP, and video streaming (Sasko and all 2014).

In this context, integrating IMS as a service delivery platform with cloud computing as a service support infrastructure is highly beneficial and as a solution to the growing challenges in modern telecommunication networks, such as scalability, high deployment costs, and system complexity (Mohamed Khalaf and all 2016). Such integration simplifies the development of value-added services, enables efficient deployment of required software and infrastructure, and leverages cloud-based security services to protect users, services, and providers (Giuseppe and all 2014).

A comprehensive review of related research is provided, focusing on cloud-based NGN and IMS implementations, virtualization of network functions, resource optimization, quality of service (QoS), and self-organizing and elastic IMS architectures. Several studies demonstrate how cloud technologies can improve performance, availability, and efficient resource utilization in telecom networks. ((Mohamed Khalaf and all 2016, Giuseppe and all 2014, Wei and all 2013), but not to the security area of data transmitted into the both technologies IMS and cloud and how to exploit the different security mechanisms offered by these technologies to secure services and users.\cite{deep2025} propose to use mTLS (mutual Transport Layer Security) and SRTP (Secure Real-Time transport protocol) to secure only the data communication between IMS end points.

Today, one of the most critical challenges in distributed and multi-environment systems is securing data transmission within cloud services and making security requirements a top priority. The European Union Agency for Cybersecurity (ENISA) has identified major cloud-related risks and provided recommendations and best practices (Enisa 2015), highlighting threats such as data breaches, malicious insiders, loss of governance, and insecure data handling.

Similarly, the Cloud Security Alliance (CSA) has proposed frameworks for building effective cloud application security architectures that ensure control, visibility, and remediation while promoting security best practices (Security Group 2025). In the same context, the National Institute of Standards and Technology (NIST) has defined key recommendations to guide organizations in selecting appropriate cloud deployment models based on their security objectives.

In IMS networks the security remains a significant challenge due to the wide range of supported services, protocols, and architectural components (Abdullah Shaikh and all 2024). This inherent complexity increases the number of potential vulnerabilities and risks affecting IMS users, services, and service providers.

The reminder of this paper is structured as follow: section 2 presents the different entities and components related to the IMS architecture and OpenStack – cloud open source. Section 3 discusses the integration between IMS and OpenStack. Section 4 presents the security mechanisms offer by IMS and offer by OpenStack and the different algorithms used to implement and exploit these mechanisms to secure user, service and provider. and finally, 5 concludes the paper.

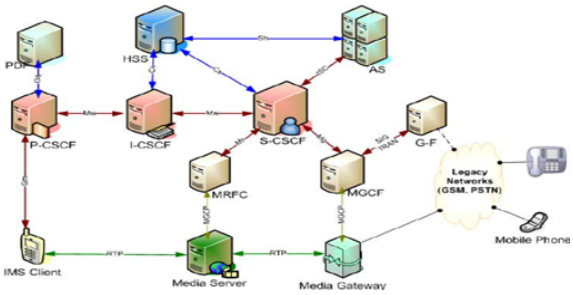

Figure 1: IMS architecture

2 IMS and Cloud

The rapid expansion of the Internet has led to significant advancements in information technology, affecting both hardware and software domains. The IP Multimedia Subsystem (IMS) is designed to support heterogeneous network environments and to enforce policies that enable efficient service delivery to end users. At the same time, cloud computing has emerged as a key technology, offering large-scale data storage capabilities along with multi-infrastructure support and open platforms for users.

2.1 IP Multimedia Subsystem (IMS)

The IP Multimedia Subsystem (IMS) serves as a reference architecture within Next Generation Networks (NGN), providing a global, open service delivery platform that enables the provisioning of converged multimedia services. IMS aims to

JMCER

standardize an overlay platform capable of delivering integrated multimedia services to users independently of their location, access technology, or time of use. To achieve this goal, IMS combines technologies from telecommunications, the Internet, and real-time multimedia systems into a unified heterogeneous framework.

In accordance with NGN design principles, the IMS architecture is structured into three main layers, as illustrated in Figure 2: the Service Layer, the Control (IMS) Layer, and the Transport Layer.

- Service Layer (Application Layer): Hosts and executes the services offered to users.
- Control Layer: Manages session control and signaling while regulating traffic between the transport and service layers.
- Transport Layer: Supports the core network by providing connectivity between access networks and IP-based core networks.

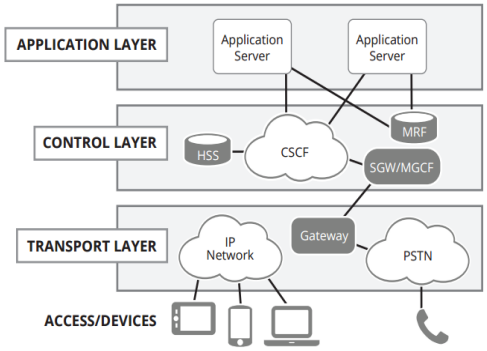

Figure 2: The IMS three layers

Every telecommunications system, regardless of its level of advancement, relies on a set of core functions such as session management, service control, media processing, and interworking capabilities to support essential communications. Within the IP Multimedia Subsystem (IMS), these functions are assigned to specific functional entities. The roles and capabilities of these entities are briefly outlined in the following sections (Mojka 2010).

Call Session Control Function (CSCF)

The Call Session Control Function (CSCF) comprises the primary control components in IMS, providing mechanisms for user authentication and authorization, session routing, and service management. A complete IMS architecture implements three distinct CSCF entities that operate together: the Serving-CSCF (S-CSCF), the Proxy-CSCF (P-CSCF), and the Interrogating-CSCF (I-CSCF).

Serving-CSCF (S-CSCF)

The Serving-CSCF is responsible for managing sessions, services, and charging functions for users. It communicates with the Home Subscriber Server (HSS) via the Diameter protocol over the Cx interface to obtain authentication data and retrieve user profiles, which are used to trigger appropriate application servers.

Proxy-CSCF (P-CSCF)

The Proxy-CSCF is located at the boundary of the IMS network and serves as the main entry point into the operator's domain. It is the first contact point for signaling messages originating from or destined for user equipment. When users are connected through a visited network, they communicate with the P-CSCF, which then forwards session signaling to the user's assigned S-CSCF in the home network.

Interrogating-CSCF (I-CSCF)

The Interrogating-CSCF also operates at the edge of the IMS network and functions as the initial contact point for incoming requests from external IMS domains. It queries the Home Subscriber Server (HSS) and, based on the response, directs signaling messages to the appropriate S-CSCF. If the HSS does not specify an S-CSCF, the I-CSCF selects one.

Application Server (AS)

The Application Server layer comprises multiple types of servers that implement the logic required to deliver various multimedia services. In IMS, application servers are categorized into three types: SIP Application Servers (SIP AS), IP Multimedia Service Switching Function Application Servers (IM-SSF AS), and Open Service Access–Service Capability Server Application Servers (OSA-SCS AS).

SIP Application Server (SIP AS)

The SIP Application Server is the most common type of AS in IMS and is responsible for hosting and executing service logic for end-user applications. It is

based on SIP technology and supports next-generation services such as instant messaging and presence.

Home Subscriber Server (HSS)

The Home Subscriber Server (HSS) functions as a centralized database and an enhanced Authentication, Authorization, and Accounting (AAA) server within the IMS network. It stores information essential for session and service control, including addressing and numbering data, security credentials, user identities and locations, user profiles, and service profiles. Additionally, the HSS generates security parameters required for authentication procedures, integrity protection, and encryption.

2.1 OpenStack- private Cloud Computing

Cloud computing provides users with on-demand access to applications and data from anywhere and at any time. It offers several advantages, including flexible billing models, intuitive web-based interfaces, scalability, and independence from specific user devices (Mizani 2017). Cloud services are typically delivered through three main models: Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS).
Cloud platforms can be classified as either open-source or public. Open-source cloud solutions include OpenStack, OpenNebula, Eucalyptus, Heroku, and Cloud9 IDE, while public cloud providers include Amazon Web Services (AWS), Microsoft Azure, and Alibaba Cloud. For this project, OpenStack was deployed on a CentOS 7 environment. As an open-source cloud platform, OpenStack supports IaaS and PaaS service models and offers extensive functionality through its modular architecture.
The OpenStack platform aims to define a global cloud computing standard while providing a flexible software framework that enables continuous innovation in cloud technologies for both providers and users. The first OpenStack release, named Austin, was introduced in October 2010 and included only two core components: Nova and Swift. By October 2025, OpenStack had reached its eleventh major release, expanding to encompass 36 integrated modules.
OpenStack consists of several essential components, figure 3, including Horizon for the graphical dashboard, Keystone for identity management and access control, Nova for compute resource management, Cinder and Swift for block and object storage, Glance for image services, Neutron for software-defined networking (SDN) and virtual network management, Ceilometer for resource monitoring, and Heat for orchestration, which automates the deployment of complex application stacks. All OpenStack services are accessed and managed through standardized APIs with consistent authentication mechanisms. Table 1, shows the description for the different OpenStack components and its code name.

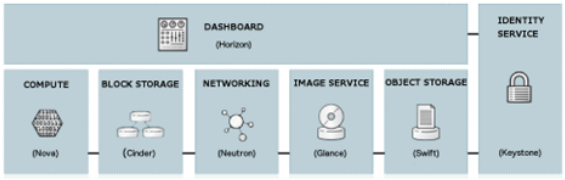

Figure 3: OpenStack Components

OpenStack provides an additional component, the OpenStack API, which allows developers to design scripts and applications that automate the provisioning and management of hardware resources through system administration and configuration processes. Using the OpenStack API, developers can implement automation for containers, application services, web server stacks, elastic cloud orchestration, database services, network traffic management, and security mechanisms.
The OpenStack dashboard interface is shown in Figure 4.

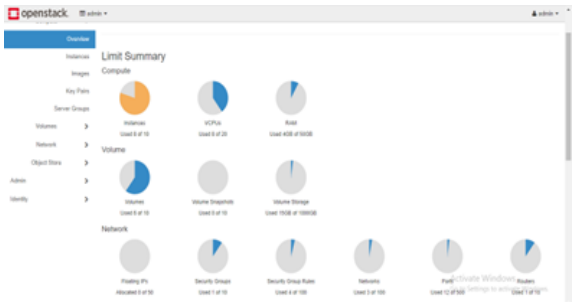

Figure 4 : OpenStack dashboard

OpenStack faces potential security threats not only from external hackers but also from internal actors, such as co-tenants or the service provider. Security evaluations were performed on the server node, virtual machine instances running Windows and Linux, and the Horizon dashboard web interface.
For this setup, OpenStack was installed on a single CentOS 7 host.

JMCER

Table 1: OpenStack Components.

Component	Code Name	Description
Compute	Nova	Instances (KVM)
Image Service	Glance	Disk(Server)
Object Storage	Swift	Scalable
Identity	Keystone	Authentication
Networking	Neutron	IP Network

3 Integration between IMS and Cloud

The IP Multimedia Subsystem (IMS) introduces a variety of new services and applications, driving increased interest from the telecommunications industry. However, traditional IMS infrastructures are facing challenges due to the growing demand for IMS services. Conventional IMS relies on a collection of SIP servers, such as CSCF entities, each dedicated to a specific role. Moreover, the scalability of front-end distributors depends on costly, specialized hardware, which limits the overall scalability of SIP-based systems. As a result, traditional IMS often underperforms compared to cloud-based IT environments, which offer high scalability and availability.

Cloud computing enables rapid and efficient deployment of diverse applications by pooling hardware and software resources. It also provides substantial computing power and virtually unlimited storage capacity. Despite these advantages, technical limitations across different cloud providers have prevented the technology from fully meeting consumer expectations. The main obstacles include:

- Lack of comprehensive signaling control: Most cloud platforms rely on web interfaces, which cannot fully support advanced mechanisms such as granular service access control and flexible pricing models. This limitation makes it difficult for telecom providers to offer profitable public cloud services.
- Quality of Service (QoS) concerns: Cloud services, being primarily Internet-based, cannot consistently guarantee QoS. Regardless of service type, cloud systems place significant demands on network bandwidth.
- Fragmented user experience: Users face platform-specific requirements and cannot easily switch between cloud providers, leading to a disjointed experience.

To address these challenges, the 3rd Generation Partnership Project (3GPP) introduced IMS, which has become the primary signaling framework for Next Generation Networks (NGN) and is widely adopted by telecom operators. IMS provides standardized signaling control and configurable QoS for IP services, which is its key advantage (Handoura 2018).

Nevertheless, IMS adoption has been limited due to a lack of innovative services. Integrating cloud computing into IMS is therefore crucial. In this combined architecture, cloud services serve as core IMS applications, while IMS provides an open and standardized service platform. By combining the capabilities of IMS and cloud computing, both technologies benefit from enhanced functionality. User devices can leverage cloud computing for internet access, audio/video decoding, and interactive processing, enabling rapid growth of IMS value-added services.

IMS's open signaling control enables advanced service access management, including digital rights protection, charging, and security. It can manage IP multimedia sessions with negotiated QoS not only at session setup but also throughout the session by interacting with network components handling application flows. Additionally, IMS can offer customized services based on user profiles, locations, access networks, and device types (Giuseppe 2014).

Through standardized interfaces, existing IMS services, such as presence, group management, authentication, and capability negotiation can be extended to cloud services. Complex cloud services can also access fundamental IMS services. By adopting IMS-based standardized cloud interfaces, service interoperability and cloud computing standardization are promoted.

The functional architecture of cloud computing integrated with IMS is illustrated in Figure 5. It updates traditional IMS specifications to meet the requirements of cloud services. Cloud functions are divided into two categories: cloud service functions and cloud interaction functions. The IMS core manages all SIP signaling for cloud session control and service notifications, while actual data flows between the User Equipment (UE) and the cloud platform bypass the IMS core, facilitating multi-provider deployments.

The UE interacts with the cloud platform through multiple interfaces: the Gm interface (via IMS core)

for session management and service control, the Ut interface for user profile configuration, and the Xd interface for accessing cloud services. These interfaces conform to 3GPP IMS standards.
User data for cloud services is split into IMS profiles and cloud-specific profiles.
IMS profiles store information required to establish IMS sessions and access services via application servers, whereas cloud-specific profiles contain data necessary to manage cloud services, such as enrolled services. IMS profiles are maintained in the Home Subscriber Server (HSS), while cloud-specific profiles may reside in dedicated databases, application servers, or the HSS. In environments with multiple HSS instances, the Subscription Locator Function (SLF) helps the IMS core and cloud service functions locate the appropriate HSS at Dh and Dx reference points.

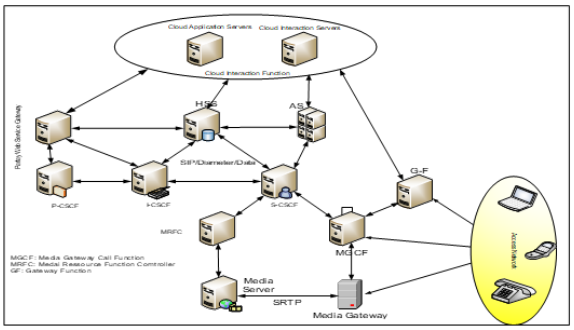

Figure 5: Integration between IMS and Cloud

4 Secure Services, Users and Providers

4.1 Cloud Computing Security-OpenStack

The Cloud Security Alliance (CSA), in its Security Guidance for Critical Areas of Focus in Cloud Computing, highlights that OpenStack's primary security controls target data protection, compliance, and operational efficiency (CSA 2025):

- Access Management: Enforce Role-Based Access Control (RBAC) and Multi-Factor Authentication (MFA).
- Data Encryption: Secure data at rest and during transmission using OpenStack tools like Cinder.
- Network Security: Implement network segmentation and configure firewalls to minimize vulnerabilities.
- Monitoring & Logging: Centralize logs and enable real-time threat detection.
- Compliance & Governance: Ensure adherence to regulations such as GDPR (General Data Protection Regulation), HIPAA (Health Insurance Portability and Accountability Act), and PCI DSS (Payment Card Industry Data Security Standard).
- Incident Response: Prepare response plans and regularly test disaster recovery processes.
- Configuration Hardening: Strengthen security of OpenStack components, including Nova, Keystone, and Neutron.
- Third-Party Integration Security: Assess external tools and protect APIs.

RBAC simplifies access management by assigning permissions based on roles rather than individual users, which improves security in OpenStack environments. Using *policy.json* files, RBAC combined with MFA (Multi Factor Authentication) provides a robust foundation for controlling access securely (Wei 2013).
Encryption complements these access controls by safeguarding data both at rest and in transit. OpenStack supports encryption across its services, ensuring sensitive information is protected throughout storage and transmission processes. Table 2, illustrates the encryptions service offer by OpenStack.

Table 2: OpenStack Services

Encryptions Services	*OpenStack Service*	*Security Level*
Data at Rest	Cinder	AES-256
Data at Rest	Nova	Server-Enc
Data in Transit	All Services	TLS

Network security functions as the first line of defense against unauthorized access and potential threats in private cloud environments. In OpenStack, it is a key concern and often ranked as a top priority by cloud administrators. Network segmentation helps isolate workloads and contain security incidents, protecting critical resources (Balan 2018). OpenStack's Neutron service provides detailed control over network configurations, while firewalls and security groups allow traffic management at the instance level. These mechanisms work together with network

segmentation, applying security rules directly to individual instances and scaling effectively as the cloud environment grows.

OpenStack environments are designed to comply with a variety of regulatory standards and industry requirements, including GDPR, HIPAA, and SOX (Sarbanes-Oxley Act). The platform includes tools for automated policy enforcement, simplifying adherence to these regulations. Additionally, these tools support audit logging, integrating with monitoring systems to provide continuous oversight and maintain accountability. The following figure, figure 6) lists the many security issues on cloud computing.

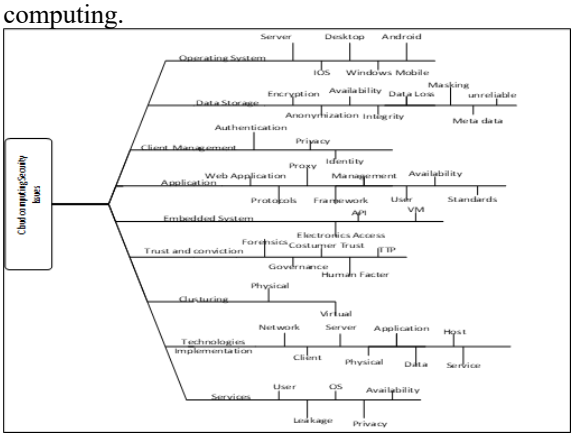

Figure 6: Cloud Computing Security issues

4.2 IMS Security

Within the IMS core, securing the services provided by IMS complements but does not replace network security. The focus is on protecting the data transmitted across the network and the services offered by the framework. This data contains both service-related information and user-specific information. To ensure the security of service data, several measures must be implemented:

- The data privacy service provides protection from unauthorized disclosure of information.
- Service data integrity: provides the means to prove data integrity and detects data modifications, deletion and re-direction.
- Service authentication: there are three different kinds of service authentication:
    - Entity service authentication: confirm partner identities during connection establishment. This service detects, simulates or replaces the identity of the equipment.
    - Human user service authentication: same idea as above for human users.
    - Data origin authentication: provides confidence in the identity of the information source.
- No-repudiation with proof of origin delivery: with this service, the recipient of a message can prove that data is sent by a specific author, and vice versa. The sender of the message can prove that the recipient received a specific message from him.
- Access to the control service: this service protects resources of this unauthorized user network (eg, reading or writing of unauthorized information, unauthorized use of services, unauthorized use of processing capacity of storage, etc.)
- Service anonymity: this service protects users against any tracking operation by any operator or user staff. As an example, it should be impossible to say where a user is currently located or what service is, he using on the network; In this context, it is inevitable, that some form of location or identity information must pass through the network to accuse purposes or to secure and establish a connection.
- Check services: these services provide functions to detect and investigate security attacks.

4.3 Securing IMS Services with OpenStack Cloud

In the network architecture that integrates IMS with cloud computing, IMS brings substantial advantages to cloud operations, while cloud platforms provide various services to support IMS. For our integration, we employ the open-source OpenStack platform (www.openstack.org) together with ImsOpenCore, developed by Fraunhofer FOKUS (Sanou 2021), as illustrated in Figure 7.

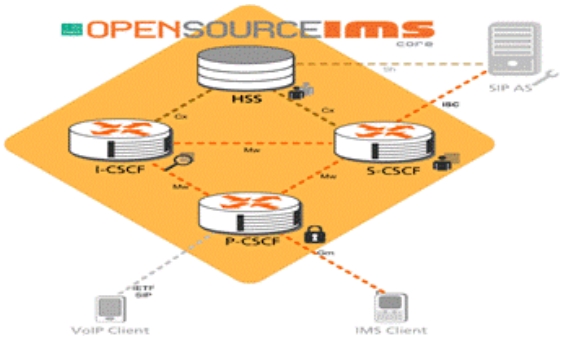

Figure 7: Open IMS Core

OpenStack features Keystone, a centralized authentication and authorization service that verifies both users and OpenStack services. Keystone manages identities, issues tokens, maintains service catalogs, and enforces policies using API versions 2 and 3. When a functional request is received, Keystone checks the credentials (username, password, URL) to confirm authorization. Upon successful verification, a token is issued, detailing the user's projects and roles. This token allows users to make subsequent requests without needing to re-authenticate, and its expiration period can be configured as needed. Figure 8, illustrates the OpenStack's main components and their interaction, (Maleki 2019) and figure 9, Keystone architecture and process with cloud components.

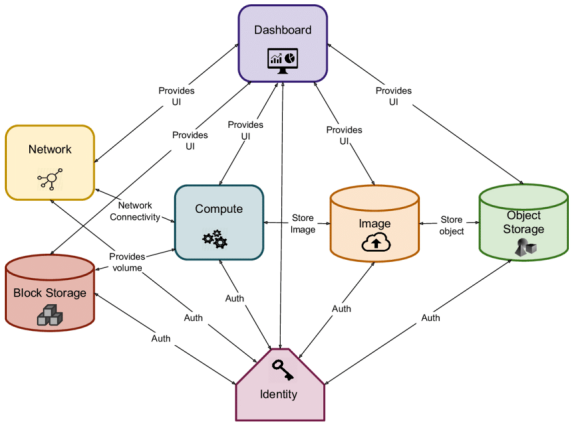

Figure 8: OpenStack main components and their interaction.

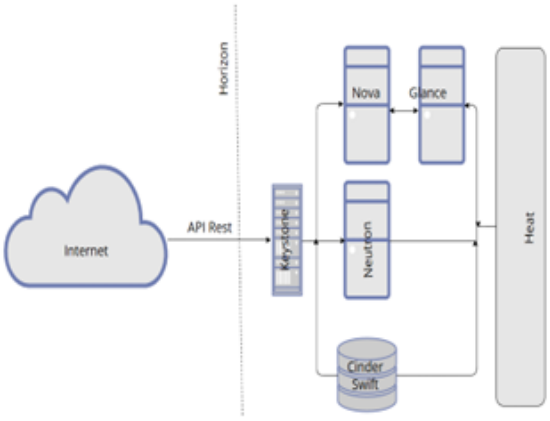

Figure 9: Keystone architecture and process with Cloud components.

According to the integration illustrated in figure 5, we create 6 Virtual machines (VM1 to VM6), and 3 Containers. Each machine presents a software service or database (Container), dedicated to a consumer. The different machine can be connected from external user with a public IP address after authentication and authorization realized by Keystone services figure 10-11.

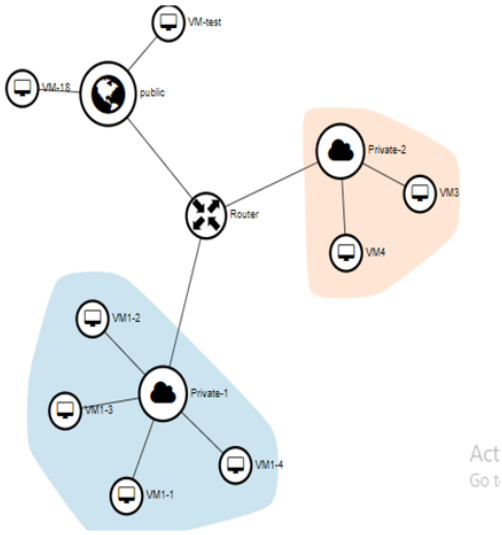

Figure 10: OpenStack Virtual machine

JMCER

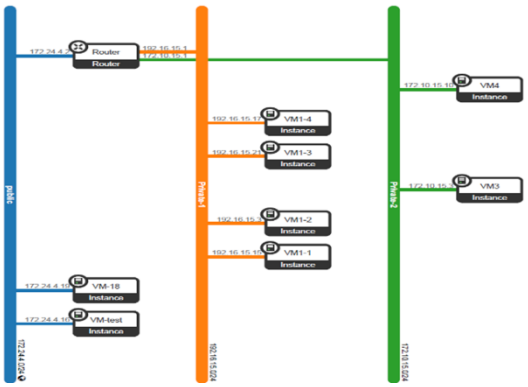

Figure 11 OpenStack Virtual Machines:

To beginning the process, we will create and send a REST API request to interact with an OpenStack service (for example, the Compute service, Nova):

Algorithm 1 Create a instance (VM-Server) via Nova

1**: POST**: http://172.24.4.10/v2.1/server1
2: **X-Auth-Token: X-Auth-Token**
3: Content-Type: application/json
4: "**Server**": VM1
5: "**imageRef**": "a6f90712-4de7-4c3c-8df4-520d7f9aab4f"
6: "**flavorRef**": 1
7: "**networks**": "private": "1b8b7d18-7b5f-…
OpenStack uses Keystone for identity and authentication, we must first send a POST request to the Keystone API to obtain an authentication token

Algorithm 2 Authenticate and get a token for IMS service with Keystone

POST: http://172.24.4.11:5000/v3/auth/tokens
2: **Content-Type**: application/json
"**Auth**":
4: "**identity**":
Methods": password

For example, to list all servers (instances) in Nova

Algorithm 3 Use the token to call OpenStack Service (list of all servers (instances) in Nova)

GET: http://172.24.4.15 8774/v2.1/servers/detail
X-Auth-Token: TOKEN
3: **Content-Type**: application/json

Algorithm 4 Registration from an IMS user service

REGISTER: sips:test.example.com SIP/2.0
Via: SIP/2.0/TLS
client.test.example.com:5061
From: Bob
¡sips:bob@test.example.com¿;tag=
4: **To**: eve ¡sips:eve@test.example.com¿
Call-ID: 12345@test.example.com
CSeq: 1 REGISTER
Authorization: Digest username="test",
realm="test.example.com",
8: **uri**= sips:ss2.test.example.com
Server : 173.24.4.1

Another security mechanism offered by OpenStack, called Security Group (figure 12). A security group is a container for security group rules. Security groups and their rules allow administrators and projects the ability to specify the type of traffic and direction (ingress/egress) that is allowed to pass through a virtual interface port.
When a virtual interface port is created in OpenStack Networking it is associated with a security group. Thus, nova.conf should always disable built-in security groups and proxy all security group calls to the OpenStack Networking API when using OpenStack Networking.

Figure 12: OpenStack security Group

Integrated into the IMS service and data is related to the Openstack API configured in the security group proposed by Openstack. Figure 13.

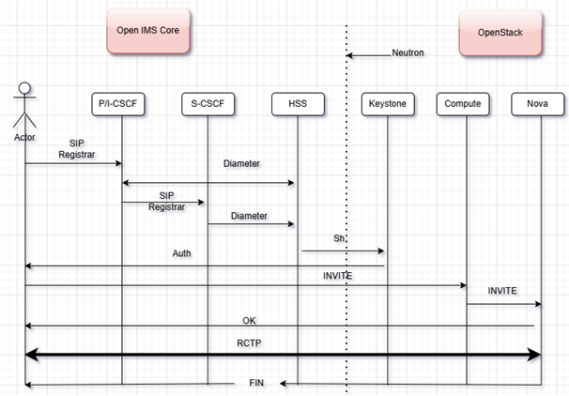

Figure 13: IMS OpenStack Connection with Authentication system.

5 Conclusions

In this paper, an integration platform based on open standards for combining IMS and cloud computing is proposed. Various vendors and developers can offer different cloud services, which are regarded as standard IMS applications. Only minor adjustments are needed for IMS terminals to access these diverse cloud services. The SIP/Diameter protocol is employed to establish and manage the communication channel between clients and cloud platforms. Once an IMS function is activated, both user and service security within the IMS framework must be ensured.

By adopting the security mechanisms proposed by the cloud environment, IMS can enhance its secure capabilities, thereby promoting investment in IMS-related services within the industry. The transition of users to cloud-based platforms and the expansion of cloud service domains are supported through mechanisms that build user trust in the protection of their data and information.

Further research is required in the future to address several challenging issues, including the scalability of IMS architecture across various technologies and its compatibility with cloud computing providers.

REFERENCES

Abdullah Ehsan Shaikh, and Simon Yusuf Enoch, 2024. “Threat-specific risk assessment for IP multimedia subsystem networks based on hierarchical models.” 20-22 December 2024 International Conference on Ubiquitous Computing and Communications (IUCC) December 20-22 2024 Chengdu-China : 123-130. doi :10.1109/IUCC65928.2024.00034.

Balan, I., Adrian Graur, and Alin Potorac, 2018, “Security assessment of OpenStack cloud using outside and inside software tools.” 14th International Conference on Development and Application Systems, Suceava, Romania, May 24-26 2018: 170–174. doi:10.1109/DAAS.2018 .8396091.

CSA cloud Security Alliance. 2025. “Security guidance for critical areas of focus in cloud computing v5.” https://cloudsecurity alliance .org /artifacts/security-guidance-v5.

Dodd-Noble A., Gundavelli, A and Korhonen, J. 2015. “3GPP IP multimedia subsystems (IMS) option, for the internet key exchange protocol version2 (IKEV2).” RFC7651.

Enisa. 2015. “ Cloud security guide for SMEs.”, ISBN: 978-92-9204-122-9 https://github.Com/ leuk7/open IMSCoreAndServices.

Giuseppe C., Marius C., and Paolo C. 2014. “Cloudified IP multimedia subsystem (IMS) for network function virtualization (NFV)-based architectures.” IEEE Symposium on Computers and Communications (ISCC) 2014. doi:10.1109 /ISCC.2014.6912647.

Handoura, A. 2018. “Secure intelligent services, sip handbook: Services, technologies, and security of session initiation protocol.” Book ISBN 97813 15218939: Chapter 21. https://doi.org/10.1201/9 78131 5218939.

Jaiswal, D. 2025. “Cloud-native transformation of SIP/IMS core networks: A microservices Architecture for next-generation telecomunications.” Sarcouncil Journal of Multidisciplinary, Volume 05, Issue 07-2025: 907–918. https://doi.org/10. 52 81/zenodo.16419992.

Maleki Hoda. 2019. “Towards security analysis of OpenStack.”, Doctor of Philosophy Dissertation 2181. University of Connecticut – Storrs. https://digitalcommons.lib.uconn.edu/dissertation s/2181.

Mehradad Mizani, 2017 “Cloud-based Computing. Key advanced in clinical information.” Elseiver: 239–255. https://doi.org/10.1016/B978-0-12-809 523-2.00017-0.

Mohamed Khalaf Alla H. M, Amin Babikerb, and Magdi. B. M. 2016. “Review in cloud based next generation telecommunication network.” Jurnal Teknologi: 51–57. https://doi.org/10.11113/jt.v7 8.5135.

Mojka.V, and Mitja S. 2010. “IP multimedia

subsystem. a guide to wireless communication engineering technologies." Book Chapter 15: 317–336. https://doi.org/10.1201/9781315219011.

Sanou J L. 2021. "Open-imscore school project." Github https://github.com/leuk7/openIMSCoreAndServices.

Sasko R., Marjan G., and Aleksandar D. 2014. "Security vulnerabil-ity assessment of OpenStack cloud." Sixth International Conference on Computational Intelligence, Communication Systems and Networks may 27-29 2014: 95-100.doi:10.1109/CICSyN.2014.32.

Wei Zhang, Weimin Lei, and Xiao Chen. 2013. "Architecture and key issues of IMS-based cloud computing." IEEE Sixth International Conference on Cloud Computing. June 28- jully 03 2013: 629-35.https://doi.org/10.1109/CLOUD.2013.34

BIOGRAPHY

Abdallah Handoura got his B.Sc., M.Sc., from Tunis university and Ph.D from the University of Southern Brittany – IMT Atlantique - France in 2009. Now, he is a teaching at Faculty of Engineering and Technology at Muscat university-Oman. He researches focuses on Cybersecurity, artificial Intelligence and quantum cryptography. He published several research papers in international conferences and journals.