An Integration of Service Compliance System for Cloud Providers

A Final Year Research Project submitted in partial fulfilment of the requirements for the degree of

Master of Science in Computer Science

ZCAS UNIVERSTIY

2025

Declaration

Name: Franklin Doroba

Student Number: ZU18070

This final year's research project is entirely my own work, except for summaries and quotations that have been properly cited.

Plagiarism check: %

Signature:  Date: 30th June 2025

Supervisor Name: Dr Njovu

Supervisor Signature: Dr Njovu

Date: 30th June 2025

Abstract

A major barrier to the mainstream adoption of cloud computing in the workplace is security, specifically security compliance. Cloud providers are required to adhere to certain security compliance standards for several reasons, including trust, legislative restrictions, and commercial needs. To date, security professionals have created this compliance or auditing data by hand. This approach necessitates manual data collection and processing, which is costly and time-consuming. To verify and evaluate the level of compliance of various cloud providers, an automated compliance tool is necessary. Such technology can eventually save time and money by reducing the requirement for human participation through automatic compliance confirmation. Cloud providers will be able to exchange security compliance data in a standard manner with this method. Because of the shared architecture, customers can compare various cloud service providers based on their security needs. These goals guided the design of our architecture, which aims to provide an automated security compliance solution for cloud computing platforms. Four distinct approaches could be used to achieve this automation. For data retrieval from cloud systems, there are four different design patterns: vulnerability scanning, log analysis, API, and human entry. Finally, we developed a proof-of-concept prototype of this automated security compliance system using the Grafana monitoring tool. The results of this prototype implementation are shared with cloud users and linked to the OpenStack cloud platform, based on the Cloud Audit API architecture developed by the Cloud Security Alliance.

Keywords: Prometheus, Grafana, OpenStack, OpenVAS, Cloud Control Matrix (CCM), Cloud Audit, and OSAPI

Acknowledgement

I would like to use this opportunity to express my gratitude to Dr. Njovu, my supervisor, for all his assistance, tolerance, and wise counsel during this research. I also want to express my gratitude to Professor Zimba and Professor Chembe for their outstanding assistance with my research.

Thank you

Dedication

I would want to express my gratitude to my wife, Nancy Ngulube Doroba, my children, Ryan and Lubuto Doroba, and my entire family for their support of my academic endeavours.

Abbreviations and Acronyms

AMQP	Advanced Message Queuing Protocol
API	Application Programming Interface
CCM	Cloud Control Matrix
CLI	Command Line Interface
COBIT	Control Objectives for Information and related Technology
CSA	Cloud Security Alliance
CSP	Cloud Service Provider
EC2API	Elastic Compute Cloud Application Programming Interface
DSS	Data Security Standard
GRC	Governance, Risk Management and Compliance
GUI	Graphical User Interface
HIPAA	Health Insurance Portability and Accountability Act
HTML	Hypertext Markup Language
HTTP	Hyper Text Transfer Protocol
IaaS	Infrastructure as a Service
ISO	International Organization for Standardization
JSON	JavaScript Object Notation
NASL	Nessus Attack Script Language
NIST	National Institute of Standards and Technology
NTP	Network Time Protocol
NVT	Network Vulnerability Tests
OAP	OpenVAS Administration Protocol
OMP	OpenVAS Management Protocol

OTP	OpenVAS Transfer Protocol
OpenVAS	Open Vulnerability Assessment System
OSAPI	OpenStack Application Programming Interface
PaaS	Platform as a Service
PCI	Payment Card Industry
REST	Representational State Transfer
SaaS	Software as a Service
SLA	Service Level Agreement
UTC	Co-ordinated Universal Time
VM	Virtual Machine
WSGI	Web Service Gateway Interface
XML	Extensible Markup Language

Table of Contents

List of Tables

List of Figures

Chapter 1 Introduction

1.1 Background of Study

The number of cloud service providers (CSP) is expanding quickly because of the pay-per-use business model's widespread adoption, which has attracted millions of customers globally. Potential clients will find it simpler to acquire a greater variety of solutions to meet the needs of their product due to the increasing number of cloud providers. The consumer must evaluate and analyse multiple cloud providers at the same time to select the best one for their products. Since most cloud service providers now use proprietary technologies to deliver cloud-based services, it is difficult to evaluate many providers using the same evaluation standards. (Justus, 2022)

The vast volume of important data kept in the cloud computing environment is the primary cause of the problem. One of the primary issues is that cloud companies are creating cloud services without adhering to open standards for cloud computing. The ongoing efforts to standardise cloud computing and several proprietary solutions that were built prior to the creation of standards are the primary causes of this. The cost of implementing cloud services in compliance with standards is another major issue. Cloud providers will not be willing to pay for it if a significant portion of their clientele disagrees with the criteria being applied. Since some vendors can be open and follow standards, as was previously indicated, there is no reason why they should keep employing closed, proprietary solutions. (Baldini, 2017)

If all providers were obliged to use the open standards, the only problem would be comparing the performance of cloud companies. The use of non-standard, proprietary technologies has made auditing very challenging and intricate. Similar difficulties arise when comparing the security practices of cloud vendors to industry norms. Despite ongoing study on the security of cloud computing, the security features provided by different cloud vendors are still not comparable to industry requirements (Werff, 2019)

Moreover, there is currently no mechanism in place to rapidly verify the security protocols that cloud companies employ. This is also one of the most often requested features by customers, according to the Martin Kuppinger Top Trends 2012-2013 Report. Furthermore, because the Cloud is not directly under its control, this report claims that there won't be enough standards or instruments for audits and authorisation. The inability of cloud computing to be audited in real-time or almost real-time is another major obstacle to its wider adoption. The viability of

creating an automated cloud computing security compliance tool that would allow users to regularly compare security methods to industry standards is examined in this thesis.

However, there are many different cloud platforms to pick from, each with its own distinct features, and cloud computing security is a broad topic in and of itself. Our thesis study aims to develop a proof-of-concept automated security compliance system, with a focus on a single cloud platform (Sakthi, 2020)

1.2 Problem Statement

The primary issue with cloud computing is its many security vulnerabilities, which include data loss, data protection, and governance loss. clients cannot realistically use a single platform for security assessments to compare or verify the security measures provided by different cloud service providers, even though all the main cloud vendors currently in use offer their clients a wide range of security protections. To address this issue, the Cloud Security Alliance (CSA) has developed guidelines. They enable cloud providers to ensure client security by offering a consistent, secure, and adaptable interface. A framework named Cloud Audit was developed by CSA (Chen, 2019)

While Cloud Control Matrix (CCM) was developed to assist cloud vendors and their clients in assessing the risks associated with a particular cloud service provider, Cloud Audit aims to provide a standardised interface for cloud service providers' auditing methodologies. To provide a comprehensive set of regulations, the CCM incorporates industry-accepted standards such as ISO 27001/27002, HIPAA, ISACA COBIT, PCI DSS, and NIST security standards. If all the current cloud vendors follow the rules outlined in the CCM and offer a standard interface for a client to validate the security measures using the Cloud Audit framework, then a client may confidently verify, analyse, and compare the risks from various cloud vendors. (Bruma, 2021)

The architecture and principles of Cloud Audit and CCM enable cloud providers to exchange security and compliance-related data, even though efforts are ongoing to determine how to give this data automatically. Users cannot independently verify this information, even if a cloud provider advertises that it has conformed with industry security procedure criteria. Our goal is to overcome this issue by automating a cloud vendor's risk assessment procedure. We intend to use a variety of techniques to automate the risk assessment procedure to generate the required data whenever it is required and eliminate the need for human involvement. (Lijun, 2019)

The primary goal of the thesis is to create a service compliance system that can consistently retrieve the necessary data from a target cloud system. The system confirms that the information gathered conforms with CCM, Cloud Audit, and other requirements. Finally, the results will be sent to the user. The preferred platform for this project is OpenStack, an open-source cloud computing platform that supports the Infrastructure as a Service (IaaS) cloud service paradigm. Before the actual job begins, a few questions need to be answered. Finding out if any aspects of the auditing process can be automated is the first step. It is impossible to automate every part of a project; therefore we need to focus on the one or two that we will automate and deal with this. The second challenging topic is how the automated auditing procedure may be integrated into OpenStack, the ideal cloud infrastructure. What techniques are available for automated security compliance analysis that adheres to CSA criteria, in summary?

1.3 Aim and Objectives of the Study

Designing and implementing a service compliance system for cloud providers is the goal of the study.

Objectives

- To analyse the current state of cloud service compliance, highlighting important industry standards and regulatory frameworks.
- Evaluate the shortcomings and difficulties in the ways that compliance is currently ensured in cloud systems.
- Create a system for service compliance that fixes issues found and complies with applicable laws.
- To assess how well the suggested system works by proving compliance

1.4 Scope and Limitation

A service compliance system is a set of protocols, policies, and tools that cloud service providers employ to ensure that their offerings meet legal requirements, industry standards, and corporate policies. The scope of such a system typically entails close observation of operating procedures, security protocols, privacy guidelines, and data handling methods associated to cloud infrastructure. It comprises implementing safeguards to reduce the likelihood of data breaches, illegal access, service interruptions, and noncompliance with regulations.

Additionally, audits, assessments, and certifications demonstrating compliance with relevant compliance standards, including ISO 27002, GDPR, HIPAA, and SOC 2, may be conducted using the system (Panetta, 2019)

Despite its many features, a cloud provider's service compliance system has a few limitations. One of the primary disadvantages of cloud computing systems is their dynamic nature, which leaves infrastructures, services, and settings open to rapid change. In a situation like this, it could be challenging to maintain compliance measures; doing so requires ongoing monitoring, adjustment, and integration with new technology. Moreover, the effectiveness of the compliance system may be limited if it depends on upstream providers, subcontractors, or other services whose compliance posture may not be totally clear or consistent with the cloud provider's standards. (Lucas 2024)

Regulatory frameworks and compliance standards are inherently complicated and differ by sector and region, which is another disadvantage. Cloud providers often operate globally in order to meet their clients varied contractual responsibilities and legal constraints. This complexity requires a thorough understanding of many legal frameworks, industry standards, and customer expectations. Attempts to standardise compliance efforts through frameworks like the GDPR or the Cloud Security Alliance's STAR program have not been successful in achieving full compliance across all major governments and companies. Because of conflicting or ambiguous regulatory standards, cloud providers may also find it challenging to maintain a uniform and cohesive compliance posture while serving a broad customer.

1.5 Significant of the Research

Cloud computing has new security vulnerabilities. The development of technologies that verify cloud providers follow industry norms and regulations like SOC 2, GDPR, and HIPAA may come from research in this area. Customers and businesses alike value data security and privacy, which is why this helps protect them. By implementing a service compliance system, cloud providers may better manage the risks of noncompliance, penalties, and data breaches. Best practices for risk assessment and mitigation strategies can be found through study in the cloud computing world. Adherence to industry norms and regulations boosts customer trust in cloud services. Research can be conducted to find the most effective methods for educating clients about compliance procedures to improve accountability and transparency.

Cloud providers must adapt to the constantly shifting laws and regulations pertaining to data protection and privacy. Research can assist cloud providers in putting in place mechanisms that

ensure ongoing adherence to legal requirements and regulatory updates. Integrating a service compliance system has two advantages: it streamlines compliance processes and reduces the time and expense of audits and assessments. Cloud service providers can reduce costs by employing research to identify potential for automation and efficiency gains. Demonstrating that their compliance requirements are robust could provide cloud providers a competitive edge in the market. Research in this area may yield creative compliance tactics that attract customers and differentiate suppliers from competitors (Mumbai, 2021)

Cloud computing has made it possible for businesses to operate internationally, but doing so requires navigating a complicated web of foreign regulations. Cloud services can expand into new fields with the help of studies that address the difficulties of maintaining compliance in various legal and cultural environments.

1.6 Security Compliance

To understand security compliance, we need to separate security from itself. While security refers to a method that must be used to safeguard a system from possible dangers, security compliance is the state of following a certain set of security guidelines. Although security is meant to prevent attacks on a system, security compliance is not the same as security. Nonetheless, security compliance ensures that the security measures put in place to protect the system meet the requirements. The processes a company employs to achieve these objectives are referred to as audit and compliance (Nasser, 2020)

The following activities must be taken to guarantee that the requirements are met and that the systems are monitored to ensure that the procedures are being followed correctly to determine which regulations the organisation must adhere to. To highlight the security component of the compliance process, IBM Research's Klaus Julisch defined security compliance as follows: The state of conformance with externally imposed functional security standards and the supply of proof (assurance) of such compliance are known as security compliance in information technology systems (Taleb, 2020).

The external security requirements for a system can now be used to define security compliance. Examples of external security requirements include government mandates, industry best practices, and internal corporation policies. However, it is now generally acknowledged that industry-recognized security standards such as PCI, HIPAA, ISO 270001/27002, NIST, and others meet the requirements for security compliance. This thesis project aims to achieve this degree of conformance. Regardless of whether an employee is prepared to follow the policy, we only concentrate on the technology aspect of security compliance in this project, even though there is a behavioural component as well. (Nasser, 2020)

1.7 Why Security Compliance

One way to think of cloud computing is as an old concept with a new name. This perspective is based on the finding that cloud computing is typically utilized to supply popular products like internet services or email through an alternative channel. It is crucial to acknowledge that we have long since established precise rules and requirements for these services. This raises the question of why maintaining cloud infrastructure security compliance is necessary while offering the same range of services. Nathaniel Borenstein and James Blake (2020) of Mimecast responded that gaining the trust of overly worried users requires compliance. This makes sense because businesses that choose to outsource their product delivery to cloud service providers forfeit control over the underlying system and are unaware of how cloud platforms operate inside. Compliance is not the same as true security, even while it promotes security. A system can still survive most security threats if it adheres to a recognized security standard (Sharma, 2018)

Data Breach Investigation Report, one of the primary causes of data breaches in the payment card sector is non-compliance. This survey found that 96% of the compromised organizations were still not in compliance with PCI DSS. Just 4% of organizations continued to be the target of hacks after achieving PCI DSS compliance. This effectively demonstrates the possible repercussions of security compliance. In general, security compliance is crucial for several additional reasons. The auditing process is the first significant and vital application of security compliance. This is justified by the fact that compliance, not security, is the focus of audits and enforcement (Suicimeczov, 2014)

The second important aspect of compliance is that assessing a system's overall security can be difficult, despite much research. However, compliance may be measured, and published matrices have been made available for this purpose. The maintenance of governance and

service level agreements (SLA) between the customer and cloud vendor is contingent upon compliance, especially security compliance, which is the third reason compliance matters. Ultimately, for a security solution to be successful in the workplace in the modern world, security audits and compliance are essential. A novel security technology that comes out of a research lab may have little use or value if security auditors do not use or recognise it (Kundu et al, 2018)

1.8 Challenges in Automating Security Compliance Checks

During a security compliance evaluation, a system's adherence to the requirements is assessed by contrasting it with a set of security standards. Up to now, manual auditing techniques have been used for this purpose. Manual audits are very time-consuming and expensive; they require information collection and informed security decisions. However, by eliminating the requirement for human interaction, automated security compliance check techniques can save a significant amount of time and money. However, before creating an automated security compliance solution for a system, we have found several obstacles that must be removed. These difficulties can be summed up as follows:

Formalizing the external standards that the system must satisfy is the first step in automating security compliance. Because there are numerous standards and not all of them are appropriate for all kinds of systems, determining requirements can be challenging. More specifically, not all the security features included in a standard will be advantageous to all systems. Regretfully, the standards now in use are quite ambiguous and provide minimal assistance with implementation. This component of standardization has made the automation process challenging. Heuristic values must be selected when implementing certain controls to ensure compliance (Nasser, 2020)

Determining what data needs to be taken out of the system to confirm the security measures is the third problem. The fourth challenge is figuring out a practical method to recover these data. Other information needed for verification can be found elsewhere, but certain information can only be found within the system. It could be necessary to modify the system to retrieve these data, which can be difficult for a deployed, working system. To prevent data from falling into the wrong hands, information must be securely moved to the authorized compliance tool (Makhlouf, 2020).

Finally, a major concern is guaranteeing the compliance status as established by the automated program. The customer must be reassured about the decision or provided with additional

information regarding the compliance check process because the compliance status may be determined using some heuristic values. The fact that many of the standards' procedures include human participation and are thus not automatable presents another obstacle to automating security compliance. For instance, physical security controls such as hardware security or staff physical entry or departure to the site cannot be verified by the automated security compliance so these issues which are related to automating security compliance checks, are present in all systems. Assessing security compliance on cloud computing systems presents new difficulties. As per the 2010 State of Enterprise Security Report by Symantec Corporation, the most troublesome security categories include Software as a Service, Endpoint Virtualization, Server Virtualization, Platform as a Service, and Infrastructure as a Service. It is far more difficult to automate security compliance tests in cloud computing because all the above-mentioned industries are intrinsically included (Makhlouf, 2020)

1.9 Cloud Computing Overview

The idea of cloud computing, which provides computing resources as a service to a range of end users, is still relatively new. End customers usually use web-based interfaces to access cloud computing services. Using a desktop or mobile interface is an additional method of accessing cloud services. Even if the software, data, and business logic are kept on the cloud system, users can still access the service via these computer interfaces. A common pool of reconfigurable computing resources, such as servers, networks, storage, apps, and services, can be accessed easily, on-demand, and through a public network, according to Elmacioglu (2021). These resources can be quickly created and made available with little help from service providers or management personnel. Cloud computing is becoming more and more popular due to its characteristics, which include resource pooling, on-demand self-service, measurable service, quick elasticity, and extensive network connectivity. For a business to avoid the headache of managing its own computer infrastructure, each of these qualities is necessary. As a result, the business may focus solely on its goods and charge the CSP to handle the computer infrastructure's security, maintenance, and other aspects (Sharma, 2018)

1.10 Cloud Service Model

The service model is the most crucial factor to consider while using a cloud system since, from the user's point of view, it establishes the limits of their control over the service. The following are the three types of service models that are available in cloud computing. Cloud users' access to programs hosted on cloud infrastructure is restricted by Software as a Service (SaaS) models.

Other than configuring a few program-related parameters, the user will have no control over the cloud infrastructure or the application itself. The client typically has no knowledge of the system components that the software is operating on when using this strategy. Customers will find this more convenient because they won't have to manage the underlying operating system, network, storage, etc. on their own. (Mahak, 2020)

The cloud service provider is responsible for maintaining all these resources. Many businesses currently adopt the Software as a Service (SaaS) concept; among the most well-known examples are Salesforce CRM and Google Docs.

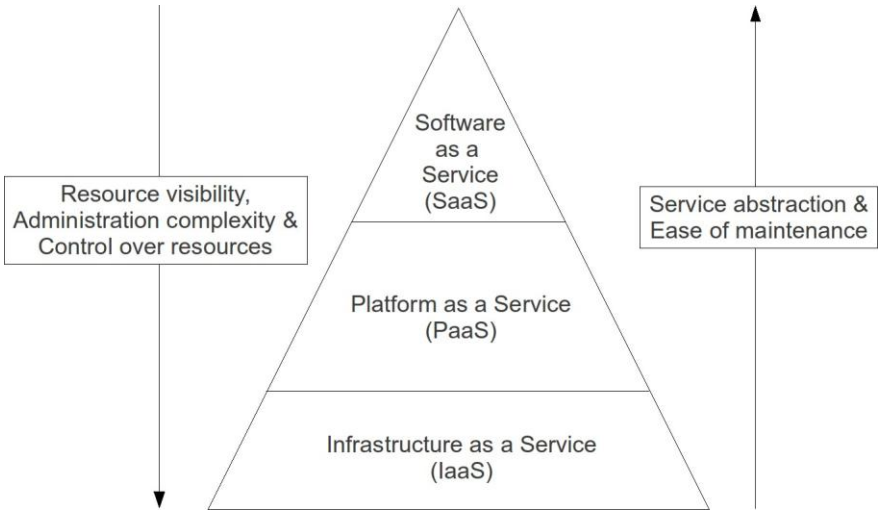


Figure 1 Show the Cloud Service Model

Platform as a Service (PaaS): This architecture for cloud computing enables users to install their own applications on the cloud infrastructure. Nonetheless, the application developed by the user needs to work with the programming languages, libraries, services, and tools offered by the cloud service provider. With this method, the user controls the application he owns and uses, but he has no influence over the underlying components that the application is based on. PaaS service providers include Google App Engine, Windows Azure from Microsoft, and others.

With Infrastructure as a Service (IaaS) models, users can get the basic computer resources they require to run any application they choose. These options for software include operating systems and any other apps the user needs. The customer has no control over the underlying cloud infrastructure under this scenario. Besides managing operating systems, apps, and storage needs, the user might also be able to select network configurations. AWS, Rackspace, and other well-known companies are among the leading IaaS service providers (Mahak, 2020)

1.11 Deployment Models

Every firm selects a cloud computing solution deployment plan based on its unique technological, business, and operational requirements. The following are the four deployment models that are currently available:

1.11.1 Private Cloud

Under this deployment strategy, the cloud infrastructure is either owned or used by a single business. However, the organization or any other outside business may oversee management and upkeep.

1.11.2 Community Cloud

Under this deployment strategy, the cloud infrastructure is either owned or used by a single business. However, the organization or any other outside business may oversee management and upkeep.

1.11.3 Public Cloud

This deployment scenario occurs when the cloud provider makes the cloud infrastructure publicly accessible. Depending on the cloud service provider's policies, using the service could be free or cost money.

1.11.4 Hybrid Cloud

This kind of cloud deployment happens when the infrastructure integrates two or more of the previously mentioned concepts.

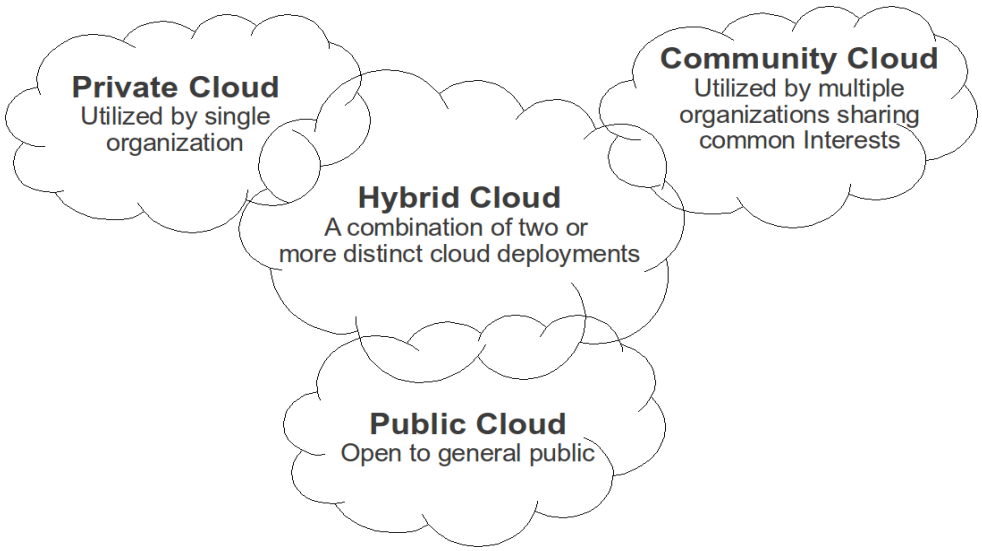


Figure 2 Shows the deployment model

This situation is useful when a company needs more processing capacity for a brief spike in traffic but maintains a private cloud to sustain ongoing operations. The company can employ a public or community cloud to reduce part of the burden during periods of high traffic. By doing this, the company can guarantee that there is never a backlog and that all services are provided on schedule. However, using a hybrid cloud deployment strategy necessitates exact synchronization of data and control logic between two or more different clouds. The degree of control an organization may have over its cloud infrastructure depends on the deployment type. The idea of a private cloud offers the company complete control and ownership of the cloud's infrastructure. Customers of public clouds, on the other hand, have comparatively little control over the cloud's architecture.

Chapter 2 Literature Review

2.1 Broad Literature Review of the Topic

A lot of basic ideas and technological advancements form the foundation of cloud computing. Famous computer scientist J.C.R. Licklider developed the concept of an Intergalactic Computer Network in the 1960s, laying the foundation for distributed computing models and enabling users in far locations to access data and applications. Grid computing first appeared in the 1990s to employ geographically distributed resources for computationally intensive activities. The Globus Toolkit by Ian Foster and Carl Kesselman offered a software framework for managing computer grids. During this same period, utility computing—which provided computer resources as a pay-per-use service—also first appeared. Sun Microsystems and Amazon created utility computing models to illustrate the advantages of renting resources on demand (Werff, 2019)

The distribution and use of computer resources underwent a dramatic shift when the phrase "cloud computing" was coined in the middle of the 2000s. AWS created Infrastructure as a Service (IaaS) in 2006 to provide online working capabilities, storage, and virtualized servers. Through the introduction of cloud-based architecture, which is both scalable and flexible, this revolutionized conventional hosting paradigms. The National Institute of Standards and Technology (NIST) played a significant role in standardizing cloud computing in 2011 by defining the word as a shared pool of programmable computer resources that may be accessed over a network on a demand basis. NIST highlighted the model's applicability and ease of use. Numerous perspectives on cloud computing have been thoroughly examined, illuminating both its promise and drawbacks. (Sakthi, 2020)

Cloud security frameworks aim to guarantee the availability, confidentiality, and integrity of an organization's data in a cloud environment. Kundu, Sura, and Sharma (2018) conducted a thorough analysis to ascertain whether new security frameworks were effective in enhancing cloud security. To determine the advantages, disadvantages, and overall impact of various cloud security frameworks and standards on enhancing cloud security posture, the study will examine and assess them. The authors have examined several well-known cloud security frameworks, including the ISO/IEC 27002:2022 Cloud Security Controls, the NIST Cloud Computing Security Reference Architecture, and the Cloud Security Alliance (CSA) Security Guidance. The effectiveness of these frameworks in addressing critical cloud security issues as access management, encryption, incident response, and data protection was assessed (Bruma, 2021)

2.2 Critical Review of Related Work

Although automated audits are crucial for security compliance in the cloud computing industry, we found that there hasn't been much research done in this field. But there has been a lot of research done in the related topic of cloud computing auditing. In cloud systems, for instance, conducting Payment Card Industry Data Security Standard (PCI DSS) standard audits presents certain difficulties, as noted by Duncan (2016). Furthermore, a theoretical foundation for PCI compliance has been developed. By evaluating the network security setup in IaaS cloud platforms using attack and reachability graphs, Bandari (2023) has demonstrated a method for identifying network-related flaws in the system. Their automated method eliminates the requirement for human interaction in both configuration data retrieval and decision-making. Their only responsibility is to assess network security.

Similar studies in the fields of risk assessment and cloud privacy have already been published. The Privacy Impact Assessment is a technique developed that can be used to evaluate whether privacy laws are being observed. Furthermore, because of the significant distinctions between cloud computing and a standard computer system, Cloud monitoring is one issue of interest that is related. From the standpoint of monitoring, this has determined the domain that distinguishes a cloud computing environment from a traditional computer environment. Since the research findings also identify information sources, we think they are important for achieving automatic security compliance in cloud systems. Some of the challenges noted for cloud monitoring are consistent with our results (Chang, 2016)

Cloud monitoring is listed here because it is now collecting data on the cloud system, and using this data could be one approach to automate security compliance solutions. Consequently, cloud monitoring can be integrated into the automated security compliance system or data can be taken from an existing cloud monitoring system. While offering cloud services, many of the ongoing projects prioritize privacy and compliance, particularly about GDPR. For instance, a two-phase method was suggested to enable different levels of privacy for the personal data of nodes. The nodes were able to offer heterogeneous privacy protection across multiple data servers by employing one-shot noise perturbation. Additionally, an effective incentive system was implemented to maximize calculation accuracy, if data servers had fixed budgets (Ducato, 2016)

The architecture's adherence to the most recent data privacy regulations was not investigated. This framework, which was created with the Lemonade platform, uses measure and monitor

criteria to calculate a trust score. However, it's unclear how this research evaluates the reliability of various cloud providers. Moreover, there is no discussion of the GDPR rules that would encourage transparency and equity. The architecture's adherence to the most recent data privacy regulations was not investigated; suggested the atmosphere, an open and equitable framework for a reliable cloud ecosystem. This framework, which was created with the Lemonade platform, uses measure and monitor criteria to calculate a trust score. However, it's unclear how this research evaluates the reliability of various cloud providers. Moreover, there is no discussion of the GDPR rules that would encourage transparency and equity (Nuno, 2018)

According to the present plans, blockchain technology can also be a helpful tool for confirming whether a certain security measure has been implemented. This makes it possible for cloud providers to create an audit trail using a secure, completely distributed, consensus-based method. offered a platform to assist cloud users in creating and implementing multi-cloud solutions. Despite using GDPR regulations to protect cloud users' privacy when managing their data, the framework lacked a Blockchain-based system to automatically confirm these regulations for provider processing activities. However, the cloud architecture should provide an automatic method for identifying any privacy or GDPR infractions. For example, it was suggested that blockchain technology and smart contracts be used to automatically monitor and enforce data sharing agreements between cloud providers and clients (Desai, 2018)

This method allowed several voters named in a voting contract to discover that the providers had breached their joint obligations. In a similar vein, the combination of GDPR and Blockchain enabled the development of a privacy-conscious cloud ecosystem design that encourages access limitation. To manage the sharing of medical data in an untrusted cloud environment, for example, the MeDShare system has been proposed. By employing a blockchain to document every action taken over the data during its movement from one party to another, this approach makes data provenance, auditing, and control techniques possible. Here, data travel is tracked, parties who break data permissions are identified, and data access is revoked for those violators using smart contracts and access control approaches (Xia, 2017).

According to some of these theories, container-based monitoring can offer a real-time method of tracking and documenting data events in a cloud environment while respecting privacy laws. Additionally, container-based monitoring can be used to enforce organization-specified standards and rules (like GDPR). To examine the impact of GDPR, an independent scenario for privacy-conscious software design was created. Virtualization technologies, encryption,

and authorization mechanisms form the foundation of this design. However, it does not take blockchain technology into account for the event audit trail, nor does it provide a means of automatically verifying compliance (Auila, 2020)

To automatically confirm legal compliance for actions taken by providers on cloud customers' data, some GDPR criteria were converted into smart contracts (Barati & Rana, 2020)

Despite using Blockchain and GDPR to enhance data privacy, none of the techniques offer enforcement or preventive measures for cloud providers' requests for data access and transfer. Additionally, they are incompatible with currently used container technologies like Docker and Kubernetes (Zhon, 2019)

Table 1Break Down of the System model implemented before to the proposed new system

Year	**Compliance System/Framework**	**Description**
2017	ISO/IEC 270017 Regulation for General Data Protection (GDPR)	Implementing ISO/IEC 27002 for cloud security in a cloud-specific manner Compliance for entities that handle the personal data of EU nationals
2019	Service Organization Control 2, or SOC 2, Federal Risk and Authorization Management Program (Federal RAMP)	A structure for service providers to guarantee the safe handling of client information The US government's criteria for safe cloud computing
2022	California Consumer Privacy Act, (CCPA) Cloud HIPAA	Centred on Californians' right to privacy and protection of consumer data Making sure cloud-based healthcare services adhere to HIPAA regulations
2023	Cybersecurity Framework (NIST CSF) Cloud Security Alliance (CSA STAR)	NIST framework designed for cloud-based cybersecurity deployments. Certification guaranteeing confidence and transparency in cloud security
2024	ISO/IEC 27001 Cloud PCI DSS (v4)	Cloud services using the Privacy Information Management System (PIMS) foundation

		updated specifications for protecting credit card information in cloud settings.
2025	ISO27005:2022	Integration of Service Compliance System for cloud providers

2.3 Proposed System/ Model

The OpenStack open-source cloud computing platform is the recommended cloud paradigm. In a short time, OpenStack has attracted a lot of interest from the cloud community. To develop an open-source cloud computing platform, Rackspace and the US National Aeronautics and Space Administration (NASA) developed OpenStack in 2010. OpenStack is a relatively recent player in the cloud business. Even though it began with just two firms, NASA and Rackspace, the OpenStack community now has over 40 million participating members globally, including, to name a few, Dell, AMD, Intel, Cisco, HP, and Ericsson.

OpenStack's first iteration, called Austin, was made available in October 2010. OpenStack later published four more versions: Bexar, Cactus, Diablo, Essex, and Antelope. In March 2024, the latest version, Caracal, was made available. To accomplish its goals, the collaborative software project OpenStack manages other projects. Currently in use are OpenStack Compute, OpenStack Object Storage, and OpenStack Image Service, the three main OpenStack initiatives. Two other projects, OpenStack Identity and OpenStack Dashboard, are supporting these three main projects with Caracal's release. In what follows, we provide a quick overview of each of these five OpenStack initiatives (Desai, 2018)

2.4 OpenStack Compute

This project was first developed by NASA under the codename Nova. Nova, the cloud computing fabric controller, is the central component of the Infrastructure as a Service (IaaS) cloud system. It provides and oversees massive networks of virtual machines to offer a scalable and redundant cloud computing platform. All the control panels, software, and APIs needed to set up, operate, and manage a cloud system are offered by Nova. Component-based architecture, high availability, fault tolerance, recoverability, open standards, and API compatibility are some of the design principles that influenced Nova's creation (Chen, 2019)

2.5 OpenStack Object Storage

The project was also known by its codename, Swift, at Rackspace, where it was first developed. Using collections of standardized servers, Swift enables the creation of object storage that can access and store petabytes of data. Email, virtual machine images, and photo storage are examples of more permanent data types that Swift is designed to keep, retrieve, use, and alter. It is neither a file system nor a real-time data storage system. Since Swift offers software logic for data distribution and replication among the servers, this huge storage service might be built utilizing inexpensive commodity hardware rather than specialized, costly technologies. Swift is widely used by service providers for IaaS-based storage services, document storage, archiving, and Microsoft SharePoint backend programs (Justus, 2022)

2.6 OpenStack Image Service

Through this project, OpenStack's virtual machine image delivery, registration, and discovery features are made possible. Glance is the codename for this project, same as the previous ones. With Glance, a user can search for the saved images stored in different back-end storage systems. RESTful API services are offered by Glance to make it easier to query the stored photos. With the codename Keystone, OpenStack Identity is one of the two new projects that are part of the Antelope release. OpenStack identity APIs are implemented by Keystone, which also provides OpenStack projects with identity, token, catalogue, and policy services. To safeguard the secret credentials, all RESTful APIs use SSL over HTTP ,HTTPS (Lijun, 2019)

2.7 OpenStack Dashboard

This project provides the basic user interface for OpenStack service management. The codename Horizon is typically associated with this project. Using the Django framework, Horizon offers an online user interface for OpenStack. At first, Horizon was built solely to assist with the Nova project. Later, it began to support Swift and Glance, two other OpenStack projects. Although Horizon only offers a rudimentary user interface for OpenStack services, it is adaptable and manageable for an administrator to add additional capabilities to the project (Sakthi, 2020)

Among the five OpenStack projects listed above, Nova and Horizon are merely two that are significant from the perspective of the thesis. After verifying that the Nova system complied with security regulations, we needed to expand the project's capabilities. To incorporate our security compliance-related data into the OpenStack dashboard, we also used a customized

Horizon project. Since this thesis claims that Nova is the most significant OpenStack project, we shall examine it in more detail in the following section.

2.8 OpenStack Nova Architecture

The foundational software that powers OpenStack's Infrastructure as a Service (IaaS) cloud computing platform is called Nova, formerly known as OpenStack Compute. Nova's reach is comparable to that of Amazon EC2 and Rackspace cloud servers. The OpenStack Nova architecture is shown in Figure 2.3 and is made up of the following components:

The core of Nova, the Cloud Controller, communicates with other modules via several protocols, including local methods, Advanced Message Queuing Protocol (AMQP), and Hyper Text Transfer Protocol (HTTP). Additionally, it alters the system's overall condition. Storage-related services are offered by the Object Store. One object storage option that can be used with Nova is OpenStack Swift. Auth Manager offers features pertaining to access and authentication. Initially, the authentication-related tasks were managed by the nova.auth. management class (Werff , 2019)

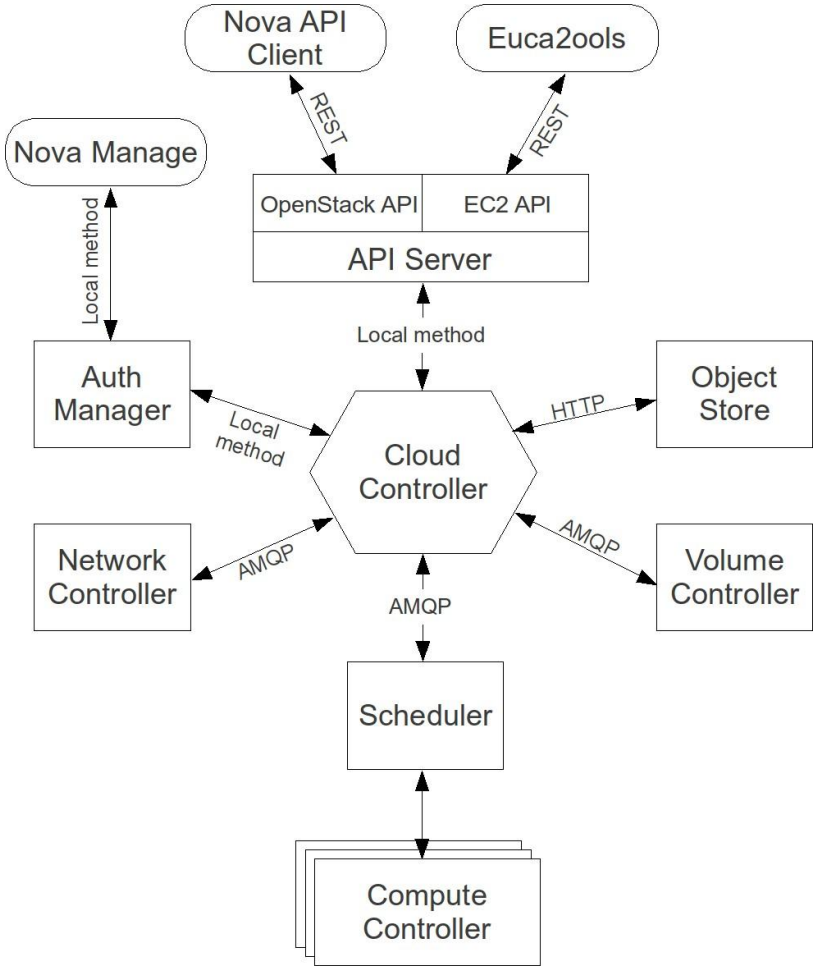


Figure 3 Shows OpenStack Nova Architecture

Features that are no longer active. The capability of authentication is currently available in the OpenStack Keystone project.

The volume controller controls the storage-related operations of a compute node, including creation, deletion, attachment, and detachment. This allows a compute node to have a persistent storage medium since non-persistent storage connected to a compute node loses all its data when the compute storage is disconnected or the instance is terminated. In Open-Stack, the Network Controller sets up the network on the host PC. For the computer nodes, it oversees the establishment of security groups, VPN setups, and IP addressing. The Scheduler connects the API calls to the relevant OpenStack component. Additionally, it chooses the resource for a task from a pool of available resources using an algorithm (Chen, 2019)

The computing resources where the instances are installed are managed by the computing controller. The OpenStack cloud architecture can be managed and interacted with by users via an external interface via the API Server. It connects to other Nova modules through the cloud controller and offers RESTful APIs to the external environment.

2.9 Information Security Standards and Frameworks

Users of cloud services are accountable for consistently ensuring that sensitive data kept on the cloud is secure. This can be a difficult, complex undertaking that calls for a layering of tools, a solid foundation of rules, guiding concepts, and techniques because the threat landscape is always changing. Industry-specific security frameworks give organizations a holistic approach to safeguarding their important data. The infrastructure supporting vital operations and processes is safeguarded by the security frameworks. Achieving regulatory compliance, establishing information handling governance, and managing data and monetary losses from a security breach are further organizational advantages. A security framework uses a coordinated set of behaviours and tools to monitor data and transactions to guarantee end-to-end security (Vahradsky, 2012)

Users of cloud services should carefully assess the framework and standard considering their needs and operations. Most of the time. CSC is required to follow several guidelines and standards. The top security frameworks and standards that satisfy legal criteria and have a big influence on cloud computing security are outlined below:

The Information Technology Security Techniques Code of Practice for Information Security Management was developed by the International Organization for Standardization (ISO). ISO 27002:2022 is the most recent iteration of ISO 27002. Tens of thousands, if not hundreds of thousands, of organizations worldwide have embraced this widely accepted standard as a benchmark for information security best practices. ISO 27002:2022 is an advisory document and not a valid information security standard. This article addresses the hazards to information security in an organized way. Information security best practices are presented in this standard in 11 security domains: Organizational information security, asset management, compliance, communications and operations management, access control, physical and environmental security, security policy, business continuity management, and information security incident management (Chen , 2019)

Organizations can use the worldwide information security standard ISO-27001 / ISO-27002 to create, implement, administer, continuously monitor, and enhance their information security management system (ISMS). To protect the availability, confidentiality, and integrity of an organization's information assets, an ISMS is a set of technologies, protocols, guidelines, and controls. To comply with the ISO-27001 standard's requirements, controls and implementation guidelines must be set up in compliance with ISO-27002.

Chen (2019) When assessing a cloud service provider's risk, cloud customers can utilize the Cloud Control Matrix (CCM), a control architecture that includes a set of fundamental security guidelines for cloud computing, as a guidance. To support security assurance, the Cloud Security Alliance (CSA) has developed a set of baseline security measures to promote the use of best practices in the cloud computing industry. The CSA CCM seeks to create a standard for the cloud security area by bridging the gap between cloud security and industry-accepted security standards.

A comprehensive understanding of security concepts and principles that align with the 13 security domains addressed by the CSA's security advice is provided by the CSA CCM. CSA claims that the following 13 domains are the most important ones to focus on when it comes to cloud computing:

Framework for Cloud Computing Architecture, Governance, and Enterprise Risk Management Legal and electronic discovery, auditing and compliance, Interoperability, portability, and information lifecycle management Disaster recovery, business continuity, and traditional security Data centre operations, notification, remediation, and incident response encryption, key management, and application security virtualization, and identity and access management (Vahradsky, 2012)

Businesses that handle cardholder data are the main targets of the Payment Card Industry Data Security Standard (PCI DSS). This standard offers technical and operational standards to safeguard cardholder information. PCI DSS compliance facilitates the development of applications with a secure credit card payment system for developers and cloud service providers. They are exempt from using a third-party merchant account provider in this situation. The PCI DSS is a global standard for credit card payment security. Any company that keeps, handles, or transfers cardholder data including credit card payment data is subject to this standard. Cardholder data includes personal information such as the cardholder's name, card verification value (CVV), card verification value (CVV2), primary account number, magnetic stripe, and expiration date. Credit card theft is avoided due to the PCI standard's enhanced data security and more vulnerability to breach (Chen & Nhien-An, 2019)

The Health Insurance Portability and Accountability Act, or HIPAA, is applicable to businesses that deal with patient data. Strict information security regulations are in place to protect patient privacy for all healthcare practitioners and organizations that handle and deal with protected health information (PHI). Electronic protected health information (ePHI) can be processed and

stored in cloud computing by HIPAA-compliant organizations if they have a Business Associate Agreement (BAA) in place between their organization and the cloud service provider. Data from the US Department of Health and Human Services (HSS) is used. After BAA is implemented, the cloud service provider must comply with HIPAA security rules to protect ePHI.

Second Service Organization (SOC 2) Control Performing an evaluation and generating a report confirming that customer data is handled in accordance with the five Trust Services Criteria confidentiality, processing integrity, availability, security, and privacy is the primary objective of the SOC 2 framework. It was the American Institute of Certified Public Accountants (AICPA) that created. As a set foundation for privacy and security, the SOC 2 report gives clients an overview of the security and privacy safeguards that are in place. Businesses can use it to build trust in their rules, procedures, and products.

One framework in the 800 series is NIST-specific Publication 800-53. It outlines federal organizations, policies, and standards for creating and preserving efficient risk management and information security systems. In addition to consistent information security regulations for all information systems, it offers rational, repeatable recommendations for choosing and implementing baseline security processes and controls. Private businesses may also benefit from these rules and regulations. The Information Systems Audit and Control Association (ISACA) created the Control Objectives for Information and Related Technology (COBIT). It serves as a foundation for information technology management and governance. In the areas of risk management, regulatory compliance, and coordinating IT strategy with organizational objectives, COBIT assists companies in overcoming operational and business issues. (Chen and Nhien-An, 2019)

A cyber security control architecture for cloud services, the Cloud Security Alliance Cloud rules Matrix (CSA CCM) is a set of guidelines based on multiple international Information Security Management System (ISMS) standards. Its 197 control objectives cover every important aspect of cloud computing. CSA The Cloud Audit framework is a standardized way to give an authorized customer information about the security and performance of a cloud service provider. This standardized approach to gathering information about these aspects enables customers to analyse various performance and security statuses and compare the services provided by various cloud service providers. This also makes it easier for cloud

providers to deliver and update the data to many customers. Using the Cloud Audit architecture, cloud vendors can provide the data and update it frequently or anytime a system change occurs.

The Automated Audit, Assertion, Assessment, and Assurance API (A6) was used in the development of Cloud Audit. The Cloud Audit 1.0 draft document from the IETF states that the specification provides "a common interface, naming convention, set of processes and technologies utilising the HTTP protocol to enable cloud service providers to automate the collection and assertion of operational, security, audit, assessment, and assurance information." The project, initiated by Christopher Hoff, director of cloud and virtualization systems at Cisco Systems Inc., was formally adopted by the Cloud Security Alliance (CSA) in October 2010.

There are three primary parts to the Cloud Audit framework. The HTTP backbone of Cloud Audit is the first crucial element. Only authorized users can access the service via HTTP. This suggests that the service can be accessed from anywhere via the Internet. The second crucial element of Cloud Audit is defining two distinct namespace types for service provision. These two namespaces are the service namespace and the glossary. The glossary namespace provides definitions and sometimes additional information about a service, whereas the service namespace provides assertions about the local or remote services. The service namespace response must be a valid HTML page that is readable by people. The final crucial element of a cloud audit are the compliance packets. These compliance packs define the namespaces for the controls listed in CCM and the matching control from a particular standard. Five compliance packs are now offered. These consist of the compliance kits for Cloud Audit-COBIT, Cloud Audit-HIPAA, Cloud Audit-ISO 27002, Cloud Audit-NIST800-53, and Cloud Audit PCI.

Chapter 3 Research Methodology

3.1 Research Design

The software development process we employed to build our prototype was called evolutionary prototyping. The progressive software development life cycle approach known as evolutionary prototyping makes it possible to incorporate new features at any point and modify the program in response to input from clients or end users. The creation of the prototype is the most important step in the evolutionary prototyping process since it influences the final system design. The components that are most visible are initially made during the prototyping stage. Prototyping advances with a more thorough design as development continues and input from that development is integrated. Evolutionary prototyping's on-the-go concept-development feature was very helpful to our research because it fit our workflow well, especially while creating a prototype automated security compliance system. Adding incremental enhancements to our prototype system has also been made possible by the incremental software development lifecycle feature. (Chen, 2019)

3.2 Reused system components

- The OpenStack cloud computing platform is the first item on the list. To apply our idea, we have made changes to the OpenStack Nova project.
- Another open-source program that we utilized to evaluate the target cloud system's vulnerability was the Open Vulnerability Assessment System (OpenVAS).
- Our solution was built using the third framework, the Cloud Audit framework from Piston Cloud Computing.
- The OpenStack Horizon project was altered to serve as the user interface in the last stage.

3.3 Implemented Security Controls

We have examined several standards for our project, including Corbit, PCI DSS, NIST, ISO 27001, and ISO 27002. Following a comparison, we have decided to base our automated security compliance system on ISO 27002:2022. Compared to other standards, this one provides a more comprehensive description of the controls, which is why it was chosen. Additionally, each security control in ISO 27002:2022 has implementation guidelines that are not found in the other standards we examined. We have chosen to include two controls from ISO 27002:2022, which are also a component of the Cloud Control Matrix (CCM), in our

prototype system after taking these benefits into account. The following subsections cite the ISO 27002 standard in relation to these two security controls. It should be mentioned that CCM offers the ISO 27001 control number instead of the ISO 27002 control number directly. Nonetheless, many of the controls are the same because ISO 27002 is based on ISO 27001. The ISO 27002 control number is provided by the Cloud Audit compliance pack, even in situations when CCM does not give a cloud compliance control ID. Thus, it is possible to translate CCM control IDs to ISO 27002.

3.4 Clock Synchronization

This control is defined in ISO 27002, Section 10.10.6. The details of this control are as follows:

- Control: A precise time source that has been chosen should be used to synchronize the clocks of all pertinent information processing systems within a business or security region.
- Implementation Advice: A real-time clock on a computer or other communications device should be configured to a previously defined standard, such as local standard time or Coordinated Universal Time (UTC). Some clocks are known to drift, so there should be a process that finds and fixes any noticeable time difference. It is necessary to fully comprehend the date and time format to ensure that the timestamp matches the actual day and time. Consideration should be given to local characteristics, such as daylight savings time.
- More information: Accurate audit records are necessary for investigations and are sometimes used as proof in court or disciplinary proceedings. For this reason, computer clock settings need to be accurate. Inaccurate audit logs could complicate these inquiries and jeopardize the validity of the supporting documentation. One possible master clock for logging systems is one that is connected to a radio time broadcast from a national atomic clock. All servers can be kept in sync with the master clock by implementing a network time protocol.

As was already stated under additional information in the paragraph before, this security control is crucial. Furthermore, it is a crucial issue from the standpoint of a cloud user. This is because a cloud system that is out of sync with UTC may have some algorithms that aren't working properly. A precise timestamp on every transaction must also be maintained by banks and other cloud-based financial institutions. Failure to keep an accurate timestamp could result in serious repercussions for these organizations (Bruma, 2021)

This control corresponds to cloud security compliance with control ID SA-12 in the Cloud Audit ISO 27002 compliance pack. Control ID SA-12 identifies these security measures, which are identical to those mentioned by CCM SA-12. We used these steps to confirm this security precaution (Chen, 2019)

- Acquire the target machine's system time to verify this control.
- At the same time, obtain the UTC time from a trustworthy NTP server.
- The target system's synchronization with the UTC time should be determined.

In this case, getting the timings from the Network Time Protocol (NTP) server and the target system simultaneously is crucial. The decision might be incorrect if this isn't the case. For instance, even if the target system is in sync with the time, the system will still be deemed to be out of sync with the UTC time since the measurements from the two systems are made at different times. In a different case, the measurements from the target computer and the NTP server are off by a few milliseconds, and the target system is off by a few milliseconds from the UTC time. The decision in this instance might imply that, although not being synchronized, the target system is in sync with UTC time (Panetta, 2019).

3.5 Remote Administrative and Diagnostic Port Protection

According to ISO 27002, this security measure is defined in Section 11.4.4. What this control is specifically about is as follows:

- Control: Limiting both logical and physical access to configuration and diagnostic ports is essential.
- Implementation guidance: Using a key lock and coordinating protocols to control physical port access are two possible strategies to restrict access to diagnostic and configuration ports. Making sure that diagnostic and configuration ports are only open via a contract between the computer service manager and the hardware/software support specialists who need access is an example of such a supporting approach. A computer or network facility should have any installed ports, services, and the like that are not particularly needed for business activities disabled or deleted.
- Additional Information: Many computer, networking, and communication systems come with a remote diagnostic or configuration tool that maintenance engineers can use. These diagnostic ports offer an avenue for unauthorized access if they are not guarded.

Through the usage of open ports, this control ensures that an unauthorized person cannot gain access to a system. Not every manual or physical procedure covered by this control can have its compliance automatically verified. On the other hand, it is vital to disable or eliminate any ports that are not needed for business activities. This statement might potentially be automatically confirmed. We have carefully examined the system for unneeded open ports in our implementation. Any system's open, unoccupied ports are always considered a security concern. The Cloud Audit ISO 27002 compliance pack's cloud security compliance is represented by this control (control ID IS-30). Control ID IS-30 indicates that these controls are the same as those found in the CCM IS-30 Cloud Security Alliance (July, 2017)

3.6 Adopted Method and Justification

We begin by using APIs to collect data for compliance evaluations. The benefit of this method is that the system from which the data must be extracted controls the API. Therefore, compared to an external system that seeks to collect the same data, the system can generate such information more readily and consistently. This method was chosen to confirm the ISO 27002 Clock synchronization control. To verify this control, the system time needs to be acquired. By providing a new OpenStack API that retrieves and transmits the system time, this data might be acquired. Since this API technique is better suited for accessing this kind of system data, we have chosen to use it to retrieve system time. Using the alternate tactics outlined in chapter 3 would have made it much more difficult to regularly obtain system time (Patrick, 2020)

We changed the OpenStack Nova compute cloud fabric controller to add a new API to its OSAPI pool to provide it. Using Python Paste Deployment; OpenStack Nova is implemented in a modular approach. Furthermore, most of the APIs are Flask APIs. As a result, we created a Flask API and made it available through Nova OSAPI, all the while preserving the Nova API architecture. To offer a new API, we did the following (Chen, 2019).

- The new API's URL needs to be made first. The name of the server hosting the Nova API server is indicated by the "server name" field in this URL. The Nova APIs are located on port 8774 in OpenStack. The API version is V1.1. The user ID is x, and the name that makes the query to the Nova API to get the current system time is current time.

- OpenStack uses the Routes module from the code implementation to convert this URL into a controller and an action. Once received, control is passed to the URL's controller class, where it is mapped to an action function.

- The action method retrieves the system time by doing the real work. Furthermore, parameters that were taken from the URL are passed to this action method. The result is finally returned as a dictionary data structure by the action method.

- The client that made the API request receives the result dictionary back after it has been serialized to XML or JSON by the Web Service Gateway Interface (WSGI) Controller.

The result dictionary is sent back to the client that made the API call after being serialized to XML or JSON by the Web Service Gateway Interface (WSGI) Controller.

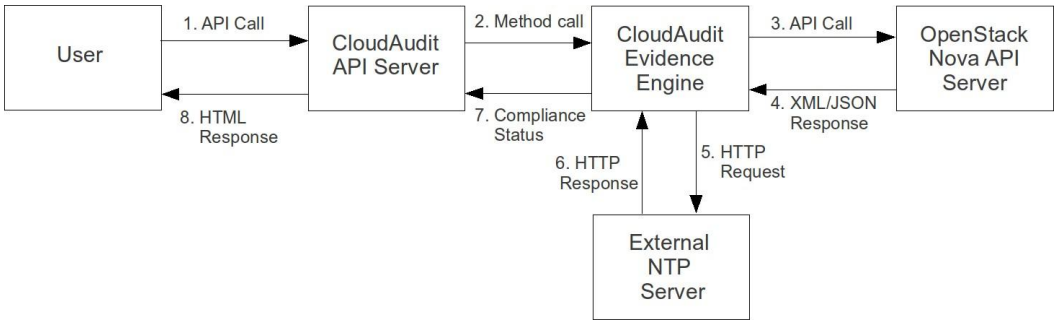


Figure 4 Control Flow for Automated Security Compliance Tool using API Mechanism

3.7 Association of Research Method to Project

The second method is to collect data using a vulnerability scanner. We use the open-source OpenVAS tool as a vulnerability scanner in our automated security compliance tool prototype. This method allows us to obtain external information about a target system.

This method has been used to validate the Remote Administrative and Diagnostic Port Protection control, which is ISO 27002. Port scanning will only be used to partially verify this control. There are various ways to accomplish this. For this, the first three strategies can be applied. Since the log analysis method and the API strategy both collect data from the system internally, we have determined that it is preferable to perform this port scanning externally using the Vulnerability Scanner approach. Most of the risks we might anticipate from external systems were taken into consideration while making this design choice because the control is focused on remote access. We had to modify the OpenVAS tool's scanning engine to include a

new NVT to use it for our objectives. To get our prototype solution to function, we've done the following:

- We have created a new NVT for port scanning using the NASL programming language. We have included the logic for determining which ports should be closed for OpenStack in the updated NVT. The NVT creates a report listing all needless open ports on the target system if it discovers any that are open and not required for operation or by OpenStack.
- To integrate the recently added NVT, rebuild the OpenVAS system.
- Adding this NVT to an OpenVAS scan setup is the next step. When the OpenVAS scanner is configured for scanning, it knows which NVT to execute first.
- We had to start an OpenVAS task from scratch using the previously mentioned scan setup. At this point, the target system that the NVTs will function against is also established.
- To launch, track, and report on the job we established in OpenVAS, we had to write a shell script.

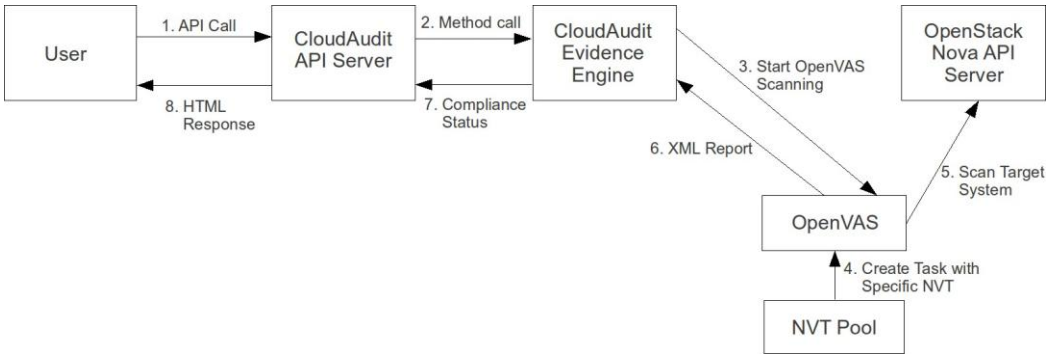


Figure 5 Control Flow for Automated Security Compliance tool

3.8 Vulnerability Scanning Mechanism

The Remote Administrative & Diagnostic Port Protection API is called by the Cloud Audit evidence engine to launch the shell script. Once the job is complete, the Cloud Audit evidence engine analyses the generated XML report to validate the control and go on to further processing. Figure 3.2 shows the entire architecture using this OpenVAS vulnerability scanner technique.

3.9 Building Cloud Audit API/Evidence Engine

Piston Cloud Computing developed the Cloud Audit platform, which serves as the foundation for our automated security compliance software. Only four APIs that use log-based analysis methods to verify compliance are supported by Piston Cloud Computing's Cloud Audit platform. This framework uses the NIST standard to verify conformance. To improve the framework, we added two new techniques: the API mechanism and the vulnerability scanner methodology. To comply with ISO 27002:2022, we have also modified the code base control structure. Python Paste Deployment and all its APIs are used in the implementation of the Cloud Audit system. We have completed the following steps to develop our solution utilizing the Cloud Audit framework: (Bruma, 2021)

- Registering our APIs with the framework is the initial step. This also describes how control is transferred when a request is received.
- Finding the controller class that accepts and manages the request is the next stage. Each request must be handled by a single controller class for each control in the standard. In the end, it calls a function in the Cloud Audit evidence engine to process the request.
- We must set up suitable procedures in accordance with the Cloud Audit evidence engine to gather genuine proof of compliance, compare it to the standard control, and then report the findings. Our evidence engine design has a single process for every distinct security control.
- The result from the evidence engine is transformed into a visual state by the controller class and sent to the Cloud Audit API server, which then sends it to the client.

We must base our decisions on certain implementation issues as part of the Cloud Audit evidence engine development process. To reassure the user of that choice, we also had to think about what information should be displayed in the user interface. These are covered in the subsections that follow.

3.10 Implementation Considerations

It can be challenging to automate security compliance checks because there is little to no guidance on how to apply the standards' criteria. We were therefore forced to make certain independent choices to put these limitations into effect. We call such choices "implementation considerations." The Clock Synchronization regulation of ISO 27002, for instance, does not outline the maximum amount of drift from the external NTP server that can occur before the

system is considered synchronized. We must decide based on the responses we get from the NTP server and the cloud interface, thus this is crucial for us (Panetta, 2019)

Since the responses from both systems cannot be collected simultaneously, we must consider some temporal drift between them. The two systems in our implementation are therefore still time synced even after a 1000 Ms difference.

3.11 Providing Assurance

Another significant concern is giving confidence regarding the choice made by the automated security compliance system. A user of the tool should be well-informed about the decision-making process and degree of accuracy because we are building the controls by taking certain implementation aspects into account on our own. We have crafted our output to provide the customer with sufficient information on implementation aspects in addition to the facts that aided in decision-making, to help them feel more confident. The Cloud Audit proof engine's visual output includes the clock synchronization and ISO 27002:2022 compliance status.

3.12 Grafana OpenStack cloud Monitoring Tool

Grafana is an open-source data analytics tool that helps us better comprehend the massive amount of data and complex architecture that our services manage by using metrics. It is possible to personalize the dashboards. Any data source, such as MySQL, PostgreSQL, Influx DB, Graphite, Prometheus, and Elasticsearch, can be connected to by Grafana. Because the system is open source, we could also develop our own plugins to connect to whatever data source we like. Technology that helps in the study, analysis, and tracking of data over a long period of time is called time series analytics. By providing us with pertinent information, it helps us monitor user and application behaviour, the kinds of issues that arise, the contextual settings, the frequency of errors that occur in production, pre-production, or any other environment. The project's ability to be deployed on-premises for companies who do not want their data moved to a vendor cloud due to security concerns is one of its key benefits. Over the years, this paradigm has been quite popular in the commercial world, being used by big organizations like PayPal, eBay, Intel, and many others (Seyi-Lande et al., 2024)

The development of a user interface for auditing purposes marked the conclusion of the prototype's implementation. One example of an established graphical user interface (GUI) for OpenStack is the Grafana dashboard project. This web-based user interface allows a user to create, start, and stop new software instances. They can monitor the state of the instances in

addition to carrying out other activities. The best location to display the data is on the dashboard. Customers may get all the information they need in one location rather of having to go to several different places (Chen, 2019)

The Cloud compliance monitoring tool is constructed using a web framework known as Grafana. The high-level flask API framework facilitates the fast and organized building of web applications. When a user accesses the dashboard, they are asked for their login credentials. Upon logging in, the user is directed to a screen that lists all the managerial relationships according to their job. We have updated this administrative page to include links to our created compliance check APIs. Thus, the Cloud Audit evidence engine effectively invokes the Cloud Audit APIs and returns the results when a user clicks on one of these links, generating the compliance check result (Reis et al., 2024a)

3.13 Grafana integration with Prometheus

Monitoring is essential to guaranteeing system availability, performance, and dependability in modern cloud-native applications. Prometheus and Grafana are two popular observability stack products that assist teams in gathering, analysing, and visualising metrics in real time. Time-series data is gathered and stored by the open-source monitoring system Prometheus. It is frequently used to monitor infrastructure, microservices, and applications.

Prometheus has emerged as a crucial tool for cloud monitoring because to its robust time-series data collection, alerting, and querying capabilities. The foundation of an all-encompassing monitoring system that improves cloud security and operational efficacy is Prometheus. It is made to manage streams of high-frequency data. Prometheus collects and alerts metrics automatically, unlike traditional monitoring systems that depend on human modifications. This is crucial for cloud systems that function at an unprecedented scale and are constantly changing (Ononiwu et al., 2024a)

Prometheus' pull-based architecture enables cloud managers to gather data from several dispersed sources, guaranteeing a thorough understanding of system health and performance. Prometheus's time-series database (TSDB), designed for high-resolution data storage, is one of its unique features. It can be used by businesses to precisely identify abnormalities and analyse patterns. Teams can detect any problems before they have an impact on operations by using this TSDB system to track key performance indicators (KPIs) throughout time (Seyi-Lande et al., 2024).

Additionally, this capability is especially useful in the financial services industry, where quick transaction processing and operational dependability are crucial (Reis et al., 2024b) Prometheus enables thorough examination of performance metrics and early issue resolution by keeping a high-quality record of measurements. Prometheus's exceptional adaptability in integrating with numerous cloud services and systems is crucial for businesses overseeing intricate multi-cloud infrastructures (Layode et al., 2024a)

The value of this flexibility is particularly important in a variety of operational contexts, including financial or healthcare settings, where data monitoring requirements differ greatly. By enabling administrators to keep an eye on specialised applications like databases and containerised environments, Prometheus's exporter ecosystem expands its usefulness across a variety of industries and operational needs. The strong alerting systems of Prometheus are another essential component of its operation. Teams can create proactive security protocols that reduce hazards before they worsen by customising these warnings to sound in response to thresholds or conditions (Ononiwu et al., 2024b)

Users can specify alert routing rules, silencing, and aggregation to prioritise important alerts and reduce noncritical ones using the Alert Manager integrated Prometheus component. In high-stakes sectors like banking, where a delayed reaction to security incidents or performance problems could have catastrophic consequences, this trait is especially helpful. Prometheus also improves data-driven decision-making by allowing businesses to predict future system behaviours using historical performance data. Teams can effectively plan resources and get ready for performance by utilising previous data to find seasonal trends or recurring problems (Ojo & Kiobel, 2024a)

Predictive skills are advantageous in sectors like healthcare and finance, where they can influence risk management strategies and resource allocation (Reis et al., 2024a). Teams can access sizable databases that provide trend analysis and sophisticated analytics since past data may be preserved (Ochigbo et al., 2024a)

Grafana's monitoring capabilities are further enhanced by its interaction with Prometheus, which makes it possible to visualise complex data and create user-friendly dashboards that expedite decision-making. Because timely data interpretation is crucial to preserving the stability of cloud systems, the combination is highly advantageous for real-time monitoring. Administrators can quickly identify and address issues as they arise with Grafana's ability to visualise metrics and provide graphically accessible real-time information. Highly

customisable dashboards that give teams focused perspectives of system performance are made possible by this integration, which is especially helpful in circumstances where prompt action is essential (Reis et al., 2024b)

Furthermore, Prometheus's adaptability and ongoing development are supported by its open-source nature and vibrant developer community. Organisations can customise Prometheus to meet their unique monitoring needs thanks to the open-source approach, and frequent updates and improvements are guaranteed by the community-driven process (Umana et al., 2024a)

For industries like finance, where rules and security requirements are often changing, this flexibility is essential. Systems for monitoring must change with the times. Additionally, the vibrant developer community fosters innovation by permitting the production of exporters and plugins that expand Prometheus's capability to meet new monitoring requirements (Joseph et al., 2024)

Multi-layered access control is one security feature of Prometheus's architecture that gives businesses the power to manage user rights and guarantee that monitoring data is shielded from unwanted access (Seyi Lande et al., 2024)

Because it helps keep important metrics from being revealed to unauthorized parties, this access control is crucial in settings with high data sensitivity. Limiting access to monitoring data is essential for operational security and regulatory compliance in regulated sectors including healthcare and finance (Ononiwu et al., 2024c)

To sum up, Prometheus is a unique monitoring tool that tackles cloud computing's difficulties. It is an essential tool for proactive monitoring because of its time-series data capabilities, adaptable warning systems, and interoperability with a wide variety of cloud and container technologies. Prometheus supports data-driven, real-time decision-making when paired with visualization tools like Grafana, which contributes to operational security and stability. Tools like Prometheus will likely become more crucial for maintaining system health and efficiency as cloud infrastructures develop, especially in sectors with strict security and compliance requirements (Layode et al., 2024b)

3.14 Enhancing Visualization and Alerting with Grafana

Grafana has become a top visualization and alerting tool in cloud monitoring thanks to its potent, real-time insights that facilitate responsive system management and data-driven decision-making (Joseph & Uzondu, 2024a)

Organizations may conduct in-depth analyses of operational KPIs because to its highly configurable and adaptable dashboards, which improve user experience and enable quick changes to cloud infrastructure as needed. This feature is especially helpful in complicated cloud environments where it can be difficult to keep an eye on far-off and dynamic systems (Layode et al., 2024a)

Additionally, Grafana's alerting functions are essential to proactive cloud management since they let users specify thresholds that, when achieved or exceeded, trigger alerts (Joseph & Uzondu, 2024b)

By using these warning strategies, system administrators can take care of possible problems before they cause major disruptions. To ensure that the appropriate staff members are notified as soon as possible, Grafana's alert management system offers choices for routing notifications across a variety of communication channels, including email, Slack, and PagerDuty. In high-stakes settings like financial services, where a delayed reaction to security or performance issues can have serious financial and reputational repercussions, this real-time warning system is essential (Layode et al., 2024c)

Businesses may customize visualizations to meet their needs with Grafana's dashboard customization tool, which is crucial in industries like healthcare and energy that have strict monitoring requirements. By allowing users to concentrate on signs that are most pertinent to their operations, this customization enhances situational awareness and facilitates quicker, better-informed decision-making. The importance of this component is highlighted by the fact that energy sector managers can monitor real-time data on load balancing and power consumption by concentrating on personalized visualizations, which contributes to operational stability and efficiency (Naiho et al., 2024b)

Grafana is a popular option for companies wishing to develop specialized plugins or combine it with custom applications because it is open source and offers more functionality than just visualization and alerting. Apart from enabling users to tailor Grafana to their own requirements, the open-source methodology cultivates a flourishing community that facilitates ongoing enhancements and extensions of Grafana's capabilities (Joseph and Uzondu, 2024c)

Businesses in industries where operational constraints demand data visualization requirements, such as waste management and environmental research, would particularly benefit from this flexibility. Support for machine learning integration adds even more flexibility to Grafana by automating reaction tactics and improving predictive monitoring. Because machine learning

algorithms may identify possible risks or system anomalies by analyzing patterns in visual data, this skill is very pertinent to cybersecurity applications. Through proactive resolution of issues like network traffic fluctuations or anomalous login patterns, such as by using Grafana to train algorithms on historical performance data, organizations can lower risks and downtime (Layode and others, 2024a)

Role-based access control and user authentication are two essential data protection measures that Grafana offers to ensure that only authorized users may access private dashboards and data. These protections are vital in sectors like healthcare where data security is critical. By compliance laws like the US's Health Insurance Portability and Accountability Act (HIPAA), Grafana's access control helps prevent unauthorized access to patient data presented on dashboards in these types of environments, claim (Olorunsogo et al. (2024)

Grafana's extensive capabilities also include facilitating cross-departmental communication since its visualizations provide a common language that enables technical and non-technical stakeholders to collaborate on performance indicators and system health. This is especially beneficial in domains like STEM education, where data-driven decision-making is becoming increasingly important. Through intuitive dashboards, Grafana enables educators and administrators to monitor and evaluate learning outcome data collaboratively, effectively informing strategy and resource allocation (Joseph et al., 2024)

Finally, Grafana's scalability is a crucial component, particularly for companies that expect rapid growth or shifting data needs. Grafana's architecture allows it to scale with increasing data quantities without compromising performance. For companies expanding their cloud infrastructure or handling a sharp rise in data volume, this is essential. Grafana's scalability allows it to grow with the demands of the digital age and remain a reliable monitoring tool for both small and large enterprises. In conclusion, Grafana's scalability, alerting, and data visualization capabilities make it a crucial tool for enhancing cloud monitoring systems. Businesses can adapt the platform to their own operational needs because of its open-source flexibility, integration options, and robust security measures, which promote proactive decision-making across a range of industries. As cloud settings continue to get more complex, Grafana's ability to provide succinct, practical insights will remain crucial to effective cloud management and cybersecurity protocols (Naiho et al., 2024a)

Chapter 4 Data, Experiments, and Implementation

4.1 Appropriate Modelling in relation to Project

This chapter provides a detailed illustration of the design created solution. Several parts are assembled to form the architecture. These consist of ISO 27002, OpenStack, Cloud Audit, OpenVAS, and Cloud Control Matrix. First, we provide a high-level overview of the steps we took to develop this service compliance solution for cloud providers. The system architecture is then covered in detail, and lastly, several methods for obtaining the data needed for system design are discussed.

The automated security compliance solution is made up of three main parts, as shown in Figure 4.1. These are:

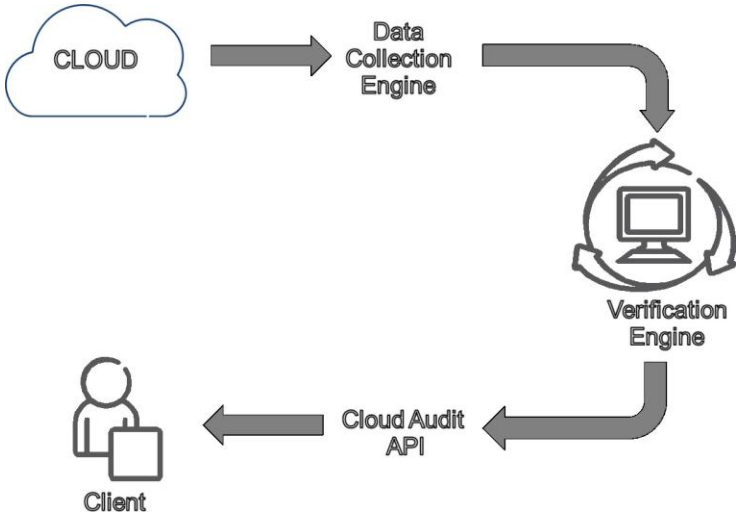


Figure 6 Automated Security Compliance Tool's High Architecture

4.2 Data Collection Engine

Effective and efficient data collection is a crucial first step in developing an automated security compliance tool. Effectiveness is the ability to gather all required data, whereas efficiency is the economical use of resources during the data retrieval process. There are a number of methods for gathering data from a cloud. The verification engine is the key component of an automated security compliance tool. This verification engine has all the information required to make decisions, including details on the controls that are based on the standards. The decision regarding the compliance of the cloud system is made using this intelligence as well as the data collected by the data collection engine.

The client's user interface is the Cloud Audit API. This compliance-related data can be shown or visualized in a variety of ways. However, as the Cloud Audit framework is the IETF draft and was created specifically by CSA, we choose to present this data via the 4.3 Cloud Audit API.

The system architecture created to provide an integrated service compliance system aimed at the OpenStack cloud platform is shown in Figure 4.2. The following is a list of the system's primary parts:

- Users: Administrator, viewer, and editor
- Dashboard: Data is shown on an OpenStack Grafana-powered web dashboard.
- Cornerstone: OpenStack identity service, which is used for identity verification and validation.
- Cloud Audit Framework: This comprises the evidence engine and the cloud audit API server. After receiving the user's request, the API server processes it using the relevant evidence engine method. The user receives the response from this API service once the evidence engine has sent it. As shown in Figure 4.1, the Cloud Audit evidence engine combines the capabilities of the verification engine and the data collecting engine.
- Cloud Audit and CCM: Describes how industry-accepted standards and the Cloud Audit API format are mapped to cloud security controls.
- Nova API server: The controller for the OpenStack cloud computing fabric that offers the API services.
- The OpenStack system's vulnerabilities are evaluated using OpenVAS, an open-source vulnerability assessment tool.

Now that the system's components have been determined, let's look at the system as it is shown in Figure 4.2. Using their login credentials, the user initially accesses the Grafana dashboard. Keystone is now being used as an identification service. The user can view and confirm a list of compliance checks that are provided in the dashboard. The control flow is sent to the Cloud Audit API server, regardless of which of these compliance checks the user chooses to verify. This Cloud Audit evidence engine ultimately initiates the procedure that oversees that compliance assessment. The evidence engine technology now collects the required data using one of four methods. Control can be given to the Nova API server, a third-party vulnerability assessment service, or perhaps the evidence engine can simply search, depending on the technique used.

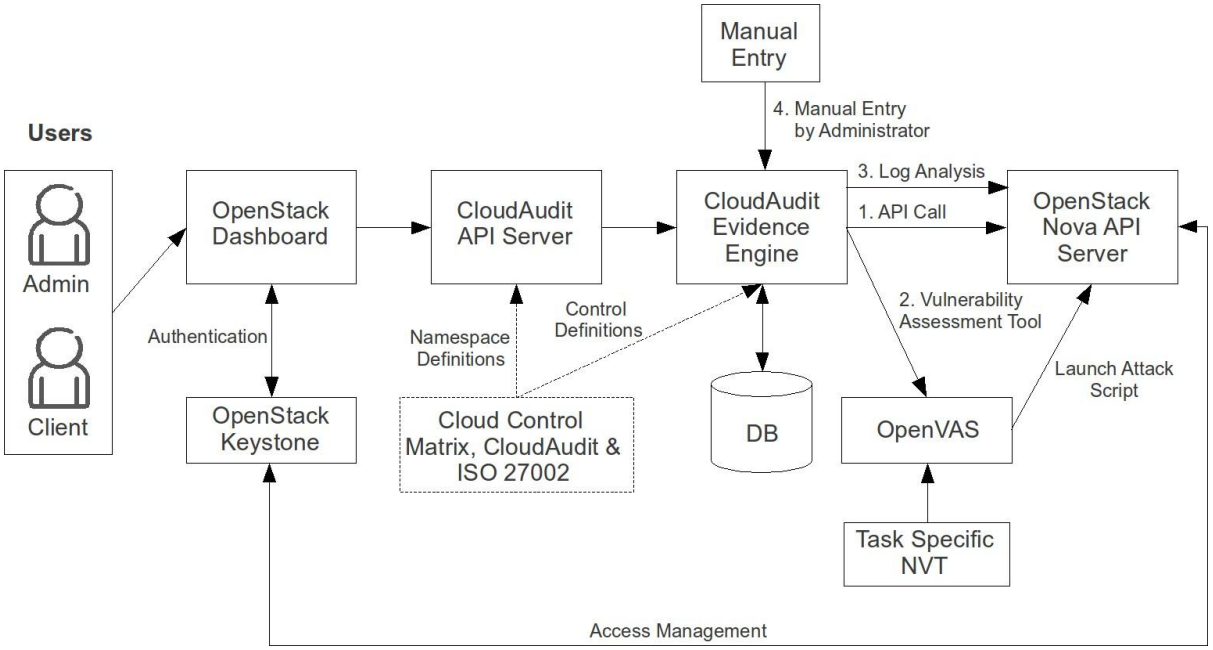


Figure 7 OpenStack Cloud Platform's Automated Security Compliance System Architecture

4.4 Design Pattern

Four potential methods for gathering cloud data for the automated security compliance check were identified during the architectural development phase. Of these four potential approaches, two are new, and we have shown that they work. The following subsections provide descriptions of the four tactics.

4.5 Application Programming Interface (API)

The simplest way to obtain any information from the system is to use the data it offers. This is the first method we have used to retrieve data from a cloud system when the Cloud Audit evidence engine requests it to confirm against security standards. We accomplished this by utilizing the cloud system's API mechanism. We suggest creating a new set of cloud platform APIs to supply the data needed by the Cloud Audit architecture. We have expanded the OSAPI pool to include a new Flask API for OpenStack. The Nova API server receives a call from the Cloud Audit evidence engine, executes the code in response, and provides the caller with the required data. The Cloud Audit framework uses this data to determine if the cloud system complies with the standard control. It is crucial to keep in mind that access to the Nova APIs is restricted. Therefore, authorization information is required before the Cloud Audit evidence engine may call a Nova API. However, the OpenStack dashboard may use the user's login credentials to make these API requests because the user has already authenticated in with them.

This method has the benefit of making it simple and reliable to supply any type of data to the automated security compliance tool since it can be gathered within the system. But there are two problems with this method. The first problem is the need to modify the cloud system to use an API to access the data. The second difficulty is that the automated security compliance program needs to be aware of which API to use to obtain this information. For this strategy to be effective, OpenStack Nova and the Cloud Audit framework must have a close relationship (Panetta, 2019)

4.6 Vulnerability Assessment Tool

The usage of a vulnerability assessment tool is our second method for the automated security compliance check. By executing scripts, this vulnerability assessment tool collects system data and produces findings. The results of this report's analysis are given to the user by the Cloud Audit framework (Patrick, 2020).

We have used the open-source vulnerability assessment tool OpenVAS for this strategy. Numerous vulnerability evaluations, including port scanning, firewall checks, backdoor checks, and local security checks, can be carried out by OpenVAS. You can use OpenVAS for straightforward tasks like locating open ports in a system or for far more complicated ones like confirming organizational regulations. This approach's primary advantage is that the integrated service compliance system can function on its own without needing to communicate with cloud systems. It doesn't need to change the cloud infrastructure to access the data externally. To log onto the cloud system, it might occasionally establish an SSH tunnel, although this doesn't necessitate a system upgrade. This strategy has several advantages, but it has a major disadvantage in that getting information from outside sources is much more difficult. Furthermore, it might take longer to complete scanning and provide the report, which could be important in an on-demand system (Bruma, 2021)

4.7 Log Analysis

Analyzing a system's logs is the third method to obtain the required data. The Cloud Audit evidence engine must use its SSH credentials to access the system to accomplish this. After that, it can examine the relevant log files and extract the necessary compliance data. If this method is employed, the Cloud Audit evidence engine needs to be provided with the target system's SSH login credentials. (Bassett, 2021)

We don't employ this technique in our thesis research because Piston Cloud Computing has already used it to create a few automated control tests for the OpenStack cloud platform. However, the design of this approach is depicted in Figure 4.3.

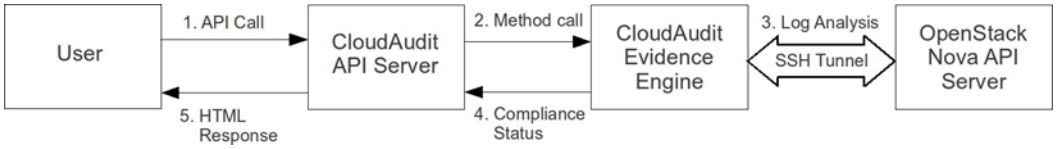


Figure 8 OpenStack Cloud Platform Control Flow for Integrated Service Compliance System

4.8 Using Analysis Mechanism

The advantage of this strategy is that using the cloud infrastructure doesn't require any system modifications. This method's drawback is that different cloud systems have different log files, and log information may be represented in multiple ways. For the automated security compliance solution that use this method to effectively gather the data, it must be closely connected to the cloud system.

4.9 Manual Entry

To get compliance-related data, the last method is hand entry using the data provided by the cloud administrator. To do this, the Cloud Audit framework ought to offer a user interface via which an administrator can enter the information needed for compliance checks. The Cloud Audit framework will store this data in a database and evaluate it based on predetermined standards when a user requests access.

This strategy was developed because some security control verifications request data that the system cannot automatically provide. For instance, the customer may wish to confirm that the cloud provider is backing up the data in accordance with corporate guidelines. The automated security compliance tool must be aware of the organization's data backup policy before it can confirm through the system that the data has been backed up in accordance with the policy. This method allows the policy to be added to the automated security compliance tool (Panetta, 2019)

This manual entry can also be supplied by a cloud vendor as text, HTML, or XML files. The automated security compliance solution must be closely coupled with the cloud system to comprehend the policy files, which in that scenario will be vendor-specific for the cloud. The primary advantage of this strategy is that, unlike in previous circumstances, any kind of data may be fed into the Cloud Audit evidence engine. The fact that the client and cloud vendor still

cannot fully trust each other because the cloud administrator must manually enter the data is another disadvantage of this benefit. (Patrick, 2020)

We have integrated the API and the vulnerability scanning data collection procedures into our prototype.

Table 2 Enumerates the benefits and drawbacks of every strategy discussed

Vulnerability Scanning	Strength	Weaknesses
API	- The mechanism itself may make it easier to extract information. - Controlling access to information could be easier.	- It is necessary to make new APIs available, which calls for a cloud system update. - The automated security compliance solution must be tightly connected with the cloud system since it must know which API to call.
Vulnerability Scanning	- There is no need to modify the cloud system because the data may be extracted elsewhere. - Capable of gathering data from the viewpoint of an outside attacker	- Collecting external data is difficult. - If you utilize a third-party scanning tool, the data collection process may take longer.
Log Analysis	- No system changes are required on the cloud side.	- When it comes to log files, systems vary. Thus, a close coupling between the cloud system and the automated security compliance system is required.
Manual Entry	- Any type of information can be provided using this method.	- The automated security compliance solution needs to maintain an internal database in order for this method to work.

4.10 Evaluation

4.10.1 Security Perspective

When assessing our products, several security vulnerabilities are found. This covers the technology's security benefits as well as possible hazards. These are covered in the subsections that follow.

4.10.2 The CIA Triad

A fundamental concept in computer security is the CIA triad. Three essential elements make up the CIA trial: secrecy, integrity, and availability. To identify information security issues and vulnerabilities, each of these elements stands for a basic security goal. We use this CIA model to assess our method in the following paragraphs.

Confidentiality of data or information refers to the capacity to hide information from people who shouldn't see it. Safely retrieving and sending the data to the automated security compliance program is one of the difficulties. This is to stop unauthorized users or outside intruders from getting their hands on confidential data. To ensure data confidentiality, we have designed architecture to limit access to the data to authorized cloud users. A user must provide their login credentials to view the OpenStack dashboard, as illustrated in Figure 4.2. Keystone, an OpenStack identification service, will confirm this. An authorized user will get access to compliance-related data following a successful authentication process. This ensures that sensitive data can only be accessed by authorized cloud users. More specific information about the confidentiality of this data can be obtained. Since the OpenStack dashboard contains data pertaining to compliance, the information presentation might be limited according to the user's role. While a typical user might only be permitted to examine a portion of the compliance data, an administrator might have access to all information pertaining to the compliance status (Sharma, 2023)

Integrity is the capacity to prevent unauthorized changes to data. In other words, once information has been supplied or requested by an authorized user, it should never be altered without consent. The API strategy used in our product is particularly vulnerable to this security goal. Because the CSP and the automated security compliance tool use the HTTP API protocol to send data, there are several potential attacks, including phishing and man-in-the-middle (MITM), that could compromise the integrity of the data. HTTPS (Secure HTTP) can be used between the automated security compliance system and the CSP to guarantee data integrity. Another way to safeguard data integrity is to install an automated security compliance solution within the CSPs' network.

Therefore, all communication will take place over the internal network of the CSPs, which is often very safe from the outside world. In this situation, it is not feasible to launch an external phishing or MITM attack against the CSPs. However, the CSP's internal network is still vulnerable to these attacks.

The capacity to make information accessible to a designated user at the appropriate time is known as availability. This security objective relates to the system's capacity to function in the event of a power outage, network outage, hardware failure, etc. Denial-of-service (DoS) attacks are one type of assault that could compromise the availability of our product. This is since all data collection techniques depend on the network to obtain information, and the flask API provides the compliance status. As a result, any Distributed DoS (DDoS) or DoS attack could prevent users from accessing the service. (Sharma, 2023)

The automated service compliance system can be implemented within the CSP network and used to gather and transmit data to the CSP via the internal network to protect against DoS or DDoS attacks. The firewall, along with other security measures, protects the CSP's internal network. Therefore, if there are no further failures, like a hardware malfunction, a power outage, or issues with updates, the automatic security compliance tool's service will be accessible.

4.11 Possible Risk

Any errors or inconsistencies could lead to an incorrect conclusion because the automated security compliance tool uses the data it collects to make all its judgments automatically. Two potential sources that might purposefully or unintentionally give false information and produce untrustworthy findings have been found. These two potential sources are mentioned below (Bruma, 2021)

4.11 Cloud Vendor

This could happen if a cloud provider purposefully gives the automated security compliance tool inaccurate information to get the desired outcomes. Another possibility is that inaccurate information was inadvertently included in the data. A malicious cloud vendor could exploit this security vulnerability in two ways. The result of gathering data from the cloud utilizing the API technique is the first case. The cloud provider has the authority to determine what information should be made available through this API and how, as they are in charge of developing and distributing compliance-related API. Consequently, the vendor may use this API technique to give inaccurate information. (Bruma, 2021)

The second scenario is when the cloud administrator is prompted to manually enter data by the automated security compliance tool. By doing this, the cloud provider may successfully feed the program inaccurate data once more, causing the automated security compliance tool to

produce false results. One way to stop inaccurate data from being sent to the automated security compliance program is to hire a third-party auditor. But this is a time-consuming procedure that needs human intervention. Even though cloud providers might purposefully give inaccurate or misleading information, their main motivation to avoid doing so is the concern that they will lose their clients' trust. The cloud vendor's business will suffer from any controversy that damages its reputation.

4.13 Third Part Service Provider

Any third-party service that the automated security compliance tool uses could be the second source of inaccurate information. An external NTP server was used to retrieve the current time to confirm the Clock Synchronization control. Since the external, third-party NTP server is dependable, the time information provided in this instance is accurate. However, the automated security compliance program can yield inaccurate findings if this external NTP server is unable to deliver the correct time. Avoiding using any third-party services at all is one potential defense against inaccurate data supplied by third-party service providers. It might be essential to develop all these services internally, which might be expensive, if employing these third-party services is the only option. There is also the option to wait to decide until after verifying that the third-party services are operating as intended (Williams & Anderson, 2019)

4.14 Cloud Assurance

The firm that develops Cloud Assurance, Fortresses, sells security and compliance tools. The governance, security, hazards, and compliance of a cloud vendor may all be evaluated with Cloud Assurance. In April 2012, the product's initial iteration was made available. The product requires a cloud vendor to use a web-based interface to answer a series of self-reporting questions. Cloud Assurance uses risk criteria to determine these provided sets of questions in a comprehensive manner. The Cloud Assurance platform uses the data from the required self-assessments provided by the cloud vendor to produce a score using its own proprietary, complex calculations. The cloud vendor is better in terms of security, governance, risks, and compliance if they have a higher score. This score is valid for the next ninety days and is just temporary. After confirming this score, a business taking part in the HISPI-controlled Cloud Assurance Assessor Program will go from a provisional to a validated status (CAAP). IT security workers can receive security training and certificates from the Holistic Information Security Practitioners Institute (HISPI) (Sharma, 2023)

We can list the following distinctions between our prototype implementation and the Cloud Assurance system based on the introduction above:

- Since the cloud provider must manually enter the self-assessment into the system, data collection is not automated.
- A company taking part in the Cloud Assurance Assessor Program (CAAP), funded by HISPI, must validate the assurance score. Human intervention is also required, and verification is carried out by hand.
- The assurance score provides an overview of the cloud infrastructure's general quality in terms of risks, governance, security, and compliance. It doesn't, however, address whether the cloud system complies with any specific standard control.

The Cloud Assurance platform's focus area or purpose is more expensive than our thesis's goals, even though it does not automate data collection and verification. Unlike this product, which prioritizes security, governance, risks, and compliance, we only concentrate on compliance.

4.15 Piston Cloud Audit Framework

During the thesis, we assessed the publicly available Piston Cloud Audit framework. The security procedures outlined in the NIST standard are used and validated in this implementation. Because log analysis is used to automatically gather data and the framework generates choices without human input, all four of the specified controls are fully automated (Chen, 2020)

With one significant exception, this work and ours are very similar. While our work proposes four distinct approaches to data collection, including the use of log analysis, the Piston Cloud Audit system just employs this mechanism. Furthermore, we contend that not all pertinent data may be obtained by the log analysis mechanism. We therefore present alternative methods as well. The goal of this thesis research is to determine whether developing an automated security compliance system on the cloud is feasible. This section covers the essential ideas for accomplishing this objective. When we began, there was no such tool, thus we had to start the project from the beginning. Our initial goal was to identify the main components and features of the system. We constructed the tool's high-level architecture based on our initial understanding of it. To use the OpenStack cloud platform to build an automated security compliance solution, we later created a system-level design for it. (Patrick, 2020)

The installation process was fraught with difficulties. Selecting a standard to confirm conformity was the first obstacle. Despite being stated similarly, sometimes slightly differently, in other standards, most controls mentioned in one standard do not include implementation advice. We evaluated several standards before deciding to build our proof-of-concept prototype using ISO 27005:2022. The fact that ISO 27005 offers some implementation instructions for the controls it outlines gives it an edge over other standards. Furthermore, according to Symantec's 2010 State of Security Report, businesses most frequently research and utilize ISO standards. Our analysis of the standards revealed that many of the controls described therein may not be automated and many of them require human input. Physical security processes are hard, if not impossible, to automate. Additionally, it's crucial to remember that some controls might only be partially automated, requiring human intervention for the remaining percentage. (Bruma, 2021)

Examining several methods for automating the security compliance check is the aim of our thesis. In this context, "alternative methods" refers to a variety of cloud data collection techniques, four of which we have identified in this thesis. Using a vulnerability scanner is one strategy that we think is especially crucial. A vulnerability scanner's objective is to identify flaws in computers, computer networks, computer systems, or applications. The vulnerability scanners gather data from the machines to perform this evaluation, and we plan to do the same here. These days, there are a wide range of vulnerability scanner types that can do several activities, such as asset profiling, configuration auditing, patch management integration, high-speed discovery, and policy verification (Columbus, 2018)

The PCI DSS compliance check for LAMP (Linux, Apache, MySQL, and PHP) servers is already supported by Nessus, Tenable Network Security's proprietary vulnerability detection tool. These initiatives are all consistent with our efforts to automatically confirm compliance. Therefore, by using these vulnerability scanners, we may improve our understanding of the cloud environment, which presents additional obstacles. The creation of an automated security compliance system prototype is one of the main objectives of this thesis. Based on our deployment observations, close integration between the automated tool and the target system is necessary. The significance of this integration for the various approaches is covered in Section 4.3. The automated security compliance tool's data gathering process may be impacted by system-level modifications, which could have an impact on the tool's functionality. This serves as the primary rationale for close integration. This implies that the target system will be tightly linked to or integrated with future automated security compliance solutions.

Chapter 5 Conclusion

Cloud computing's explosive growth indicates that it will propel the upcoming generation of internet services. Some open-source cloud solutions are growing in popularity even though the majority of cloud platforms on the market today are proprietary. Despite this expansion, the industry's broad use of cloud computing is still hampered by security, especially security compliance. An automated security compliance solution helps the user, and the cloud provider develop confidence. By automating the compliance evaluation process, this technology can also save the cloud vendor money and time. Therefore, this project is pertinent from both a theoretical and practical perspective because it examines various design techniques for creating such a tool and uses a prototype to show how it may be accomplished.

Users may collect, analyze, and visualize metrics in real time with the help of popular observability stack solutions like Prometheus and Grafana. Grafana's scalability, alerting, and data visualization capabilities make it an essential tool for improving cloud monitoring systems. Companies can customize the platform to meet their own operational demands thanks to its open-source flexibility, integration choices, and strong security features, which encourage proactive decision-making in a variety of industries. The capacity of Grafana to offer concise, useful insights will continue to be essential for efficient cloud administration and cybersecurity procedures as cloud systems grow more intricate.

User authentication and role-based access control are two crucial data security features provided by Grafana that guarantee that only authorized users can access private dashboards and data. In industries like healthcare, where data security is essential, these safeguards are essential. In line with regulations such as the Health Insurance Portability and Accountability Act (HIPAA) of the United States, Grafana's access control helps guard against unauthorized access to patient data displayed on dashboards in these kinds of environments.

Chapter 6 References

Abrahams, T.O., Anyanwu, A., Olorunsogo, T., Akindote, O.J., & Reis, O. (2024). Data integrity and confidentiality: an examination of cybersecurity and accounting measures in superannuation institutions. Journal of Computer Science & IT Research, 5(1), 237–253.

Alam, T., and Saxena, U. R. (2023). establishing role-based, trust-oriented access control to preserve cloud data integrity. System Assurance Engineering and Management International Journal, 1–20.

Adeleke, G.S., Layode, O., Udeh, E.O., Labake, T.T., Naiho, H.N.N., & Johnson, E. (2024a). Resolving Cybersecurity Issues in Sustainable Supply Chain Management: An Examination of Present Methods and Prospects. Journal of Management and Entrepreneurship Research International, 6(6), 1954-1981

Adeleke, G.S., Udeh, E.O., Layode, O., Naiho, H.N.N., & Labake, T.T. (2024b). Challenges with data security and privacy in environmental research: Strategies for protecting private data. 6(6), 1193-1214, International Journal of Applied Research in Social Sciences. https://doi.org/10.51594/ijarss.v6i6.1210 is the DOI.

Abidin, S., and Tarannum, W. (2023, March). A review of the integration of cloud computing and blockchain. (pages. 1623-1628) in the 10th International Conference on Computing for Sustainable Global Development (INDIACom) in 2023. IEEE.

Alkhasawneh, A., & Khasawneh, F. A. (2023). Legal issues of consumer privacy protection in the cloud computing environment: an analytical study in GDPR, and USA legislations. International Journal of Cloud Computing, 12(1), 40-62. Adelodun Felicia, O., Wilson, S.,

Amini, A., & Jamil, N. (2018, May). A comprehensive review of existing risk assessment models in cloud computing. In Journal of Physics: Conference Series (Vol. 1018, No. 1, p. 012004). IOP Publishing.

Ahmad, A., Scheepers, R., & Kotsias, J. (2023). implementing and incorporating cyberthreat intelligence into a business. Journal of Information Systems in Europe, 32(1), 35–51.

Asadollah, S. A., Lisper, B., Masud, A. N., Ciccozzi, F., Addazi, L., & Mubeen, S. (2022). a thorough investigation of parallel computing languages. 55(2), 1–39; ACM Computing Surveys (CSUR).

Boudries, A., Amad, M., & Bouzidi, Z. (2021, June). A survey on how deep learning, big data, and parallel computing environments are improving crisis management. Electrical, Communication, and Computer Engineering International Conference (ICECCE) 2021 (pp. 1–7). IEEE.

Chatzidimitriou, E., Kamariotou, M., & Kitsios, F. (2023). How to Get Value Out of Data in the IT Sector: The ISO/IEC 27001 Information Security Management Standard. 5828 in Sustainability, 15(7).

Crisci, L., Beni, M. S., and Cosenza, B. (2023, May). Modern C++'s Enhanced Message Passing Interface is known as EMPI. 23rd International Symposium on Cluster, Cloud, and Internet Computing (CCGrid), IEEE/ACM, 2023 (pp. 141-153). IEEE.

Cloud Security Alliance's Cloud Control Matrix (CCM).

D. Lewke (2023). A Cloud Cybersecurity Benchmarking Framework, the MIT-IBM CloudSec 16 (Doctoral dissertation, Massachusetts Institute of Technology).

D. P. Möller (2023). MITRE Cybersecurity Criteria and the NIST Cybersecurity Framework. Trends, Techniques, Technologies, Applications, and Best Practices in Cybersecurity in Digital Transformation (pp. 231-271). Cham: Switzerland's Springer Nature.

Ehimuan, B., Oguejiofor, B.B., Chimezie, O., Akagha, O.V., and Reis, O. (2024b). International data privacy regulations: An analysis of how technology affects user rights. 21(2), 1058-1070, World Journal of Advanced Research and Reviews.

Goyal, R., and R. Kumar (2021). When security and speed collide: Using DevSecOps to model ongoing security for cloud apps. Proceedings of ICIDCA 2020: Innovative Data Communication Technologies and Application (pp. 415-432). Springer Singapore, Singapore.

Godishala, A. K., Banoth, R., and Narsimha, G. (2022). A Complete Guide to Auditing and Managing Information Security. CRC Publishing.

Huang, W., Fei, M., Wei, L., Yue, S., and Chen, R. (2022, September). The evolution of Secure Access Service Edge in brief. Networking and Computers, International Conference on Signal and Information Processing (pp. 138-145). Singapore: Singapore: Springer Nature.

J. P. Bharadiya (2023). A comparison between artificial intelligence and business intelligence using big data analytics. Artificial Intelligence in America, 7(1), 24.

J. Seaman (2023). Guidelines and Strategies for Zero Trust Security. The Promise, Dangers, and Solutions of Digital Transformation in Policing (pg. 149-168). Cham: International Publishing, Springer.

Layode, O., Adeleke, G.S., Udeh, G.S., Naiho, H.N.N., & Labake, T.T. (2024b). A critical evaluation of cybersecurity issues in the application of cutting-edge waste management systems. Journal of Computer Science and IT Research, 5(6), 1408–1433. https://doi.org/10.51594/csitrj.v5i6.1225 is the doi source.

L. Gasser (2023). Quantum cryptography after the fact. Trends in Encryption and Data Protection Technologies, 47–52.

Li, R., Wu, Q., Peng, M., and Shen, Y. (2023). A machine learning technique for OpenMP variable classification. Computer Systems of the Future, 140, 67–78.

Li, Y., Wang, T., Dong, N., Li, W., Hua, H., & Cao, J. (2023). Artificial intelligence and edge computing: A machine learning viewpoint. 1-35 in ACM Computing Surveys, 55(9).

Li, M., Yahya, R. O., and Qin, M. (2023). dynamic deployment of IoT services using fog-cloud computing's shared parallel architecture. 23, 100856; Internet of Things.

Lu, Y., Zhang, J., Liu, Y., and Li, Z. (2023). Forecast-assisted service function chain dynamic deployment for cloud management systems supported by SDN/NFV. IEEE Journal of Systems.

M. W. (2022). Integrative cybersecurity is the process of combining global standards, layered defense, and zero trust to create a resilient digital future. Journal of Computer Science and Technology International, 6(4), 99–135.

M. McNett (2020). Data protection: privacy and security. Pages. 87–99 in Data for Nurses. Scholarly Press.

R. Walters (2023). Processing, Consent, Controller. Commonwealth Cybersecurity and Data Laws: Global Trade, Investment, and Arbitration (pg. 119-146). Springer Nature Singapore, Singapore.

Schopf, J. M., and Zurawski, J. (2023). (Analysis Report) National Institute of Standards and Technology Requirements. The United States' Lawrence Berkeley National Lab (LBNL) is located in Berkeley, California.

S. O. Olabanji (2023). Using high-level coding languages like Python and SQL to automate top control procedures and reinforce security systems is one way to advance cloud technology security. Scientific Research and Reports Journal, 29(9), 42–54.

Sakpere, W., & In 3rd, W. (2023). Big Data Concept, Analytics and Hadoop Technology: A Systematic Survey. In 3rd International Conference, Faculty of Natural and Applied Sciences (FASCON) 2022. Products or company names used are only for identification purposes.

S. Shreyas (2023). Cloud Computing Security Model: An Organizational Vulnerability Case Study. Information Security Journal, 14(4), 250-263.

Szádeczky, T., and Z. Bederna (2023). Controlling the financial effects of cyberattacks. Quarterly for Security and Defense, 41

Tan, J. (2022). Using Open Policy Agent in a DevSecOps cloud context to facilitate documentation and ensure component dependencies.

V. Bandari (2023). A comparative analysis of the risks and effectiveness of enterprise data security measures across various industries and organizational types. Big Data Analytics and Business Intelligence International Journal, 6(1), 1–11.

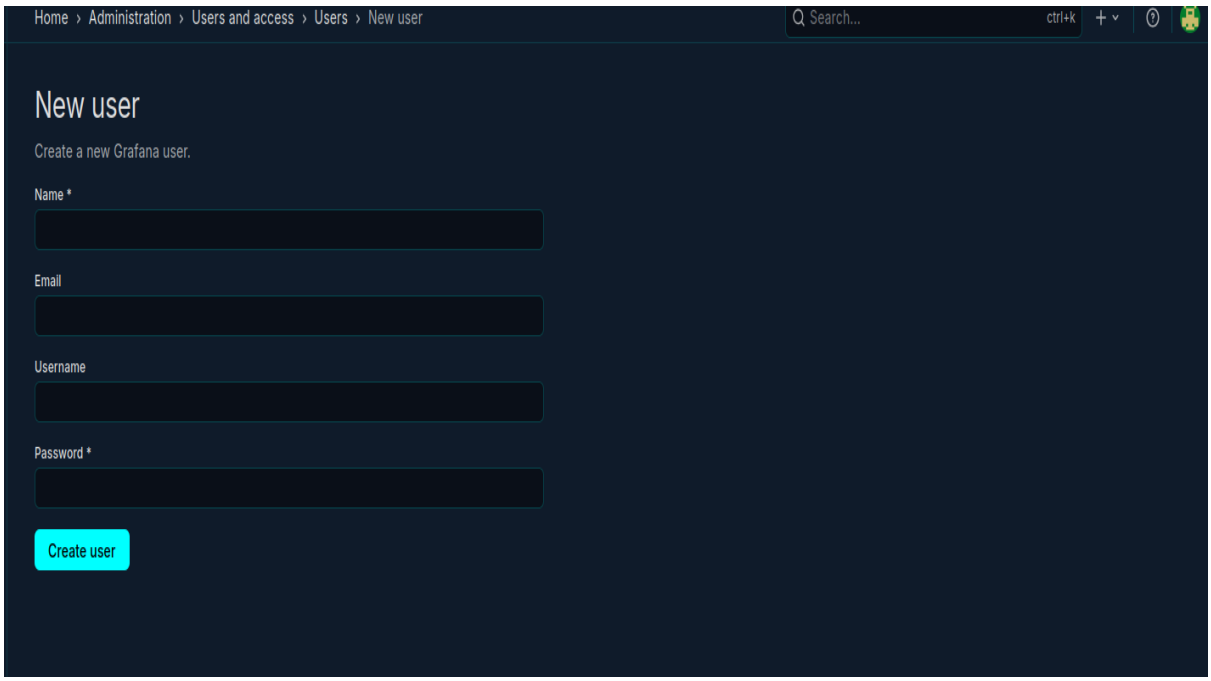
Wang, Y., Liu, W., Guan, S., and Zhang, C. (2023). safe big data storage solution in a cloud computing environment that is based on Hadoop. Networks and Digital Communications.

APPENDIX

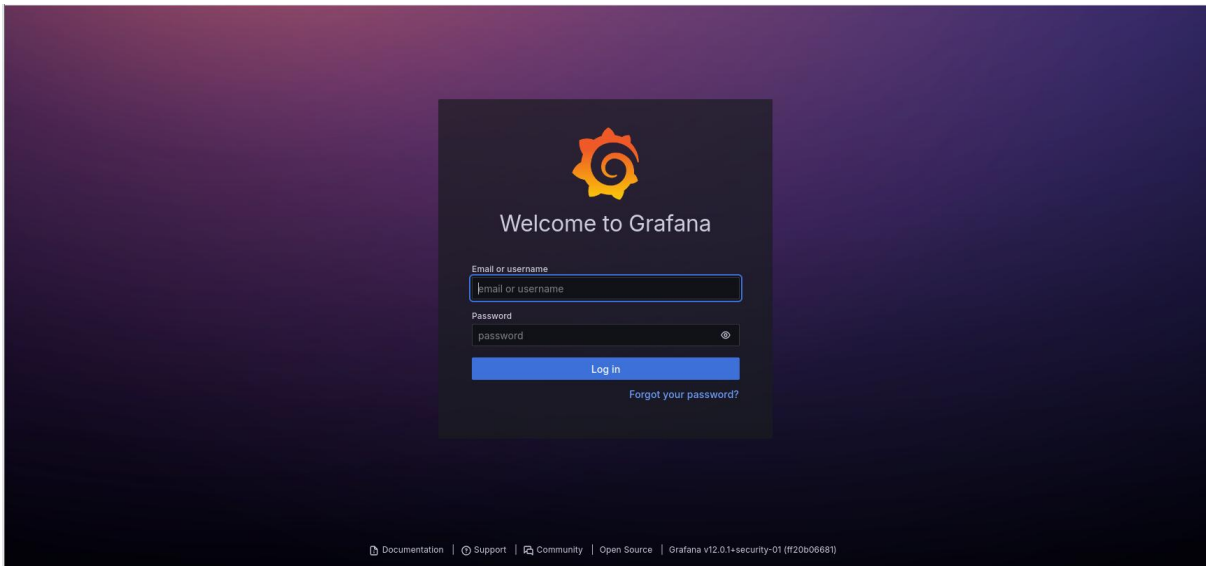
Grafana Visualization output from Cloud Audit

The results of the automated security compliance tool are shown in this section. Data is gathered and compared to the criteria by the appropriate handler from the Cloud Audit evidence engine. To verify compliance checks, a user logs into the Grafana OpenStack monitoring system. It then chooses an option and generates a visual output, such as a graph, bar chart, statistic, etc. In response to the client's request, it displays the data and, using the Cloud Audit API service, generates a visible output with the cloud compliance status. The sample output for the ISO 27005:2022 controls for Clock Synchronization and Remote Administrative & Diagnostic Port Protection, respectively is physically displayed to clients

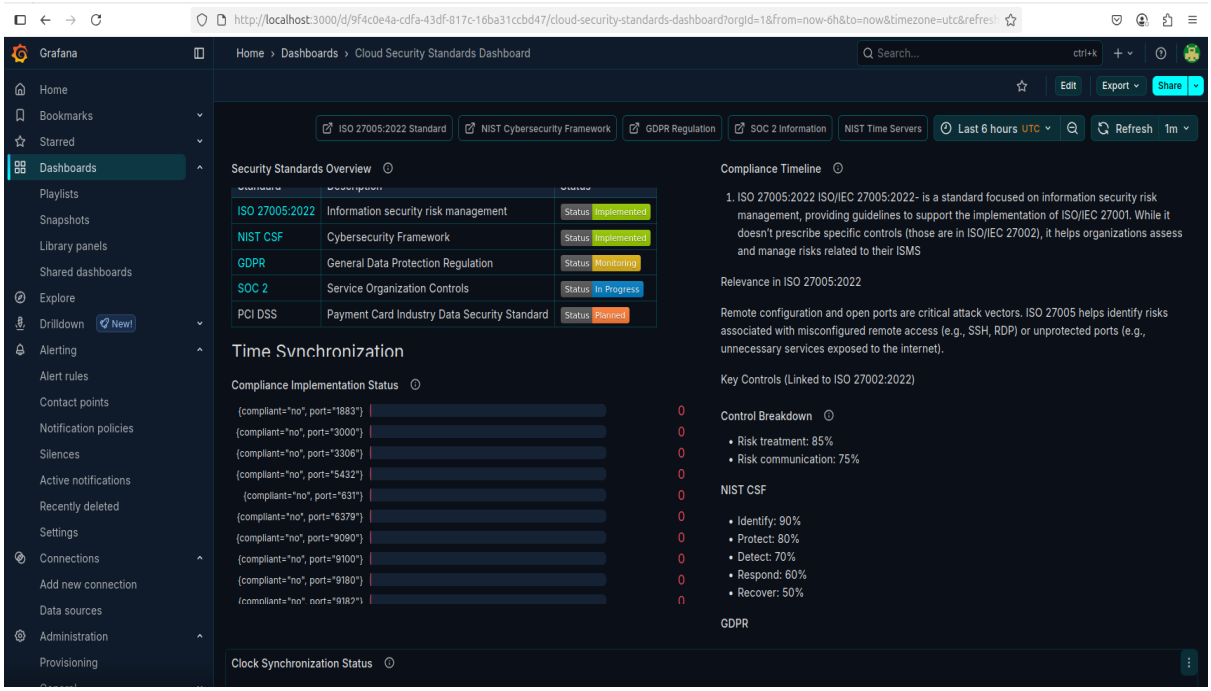
Registration page



Login page



Home Page



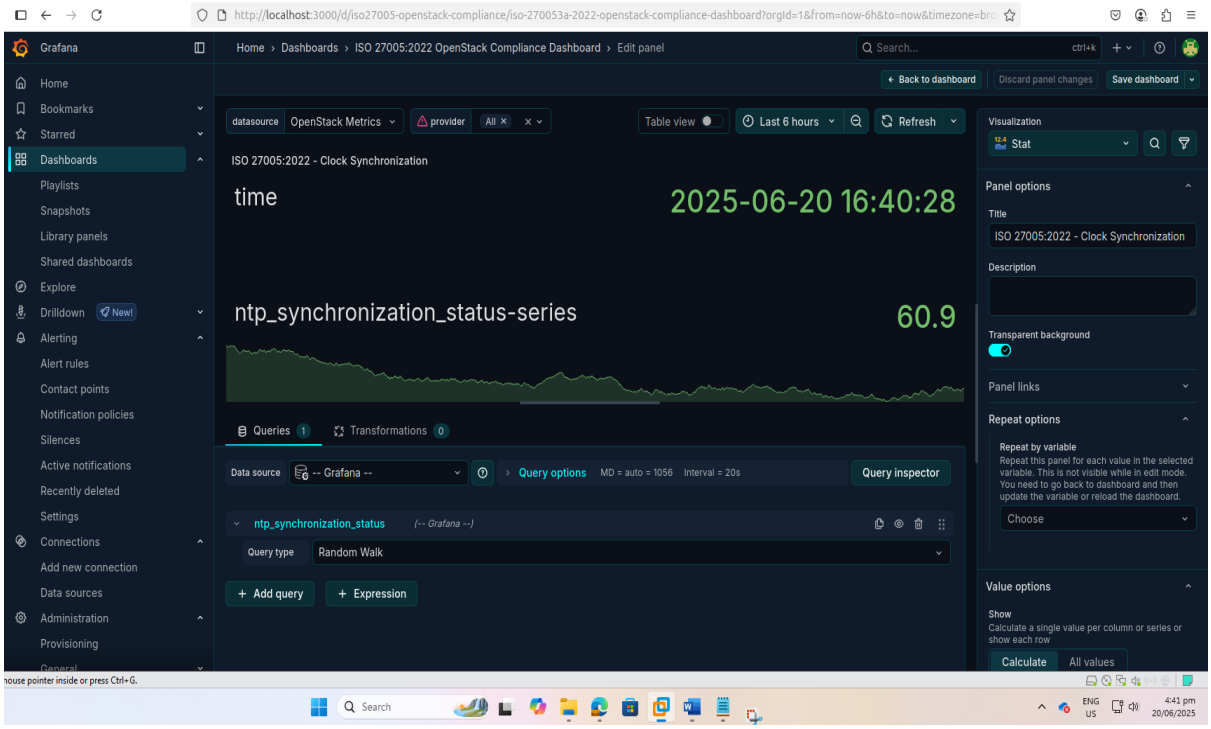
OpenStack Grafana Monitoring Dashboard

ISO 27005:2022- NTP Clock Synchronization

A security area or organization must use a single reference time source to synchronize the clocks of all pertinent information processing systems. When providing evidence of occurrences as part of a judicial or investigative procedure, system clock synchronization is especially important because failure to do so often makes it impossible or very difficult to prove cause and effect.

The visual output indicates that the system conforms with ISO 27005:2022, the Clock Synchronization standard. Green signifies that the system conforms to the standard. Here's an

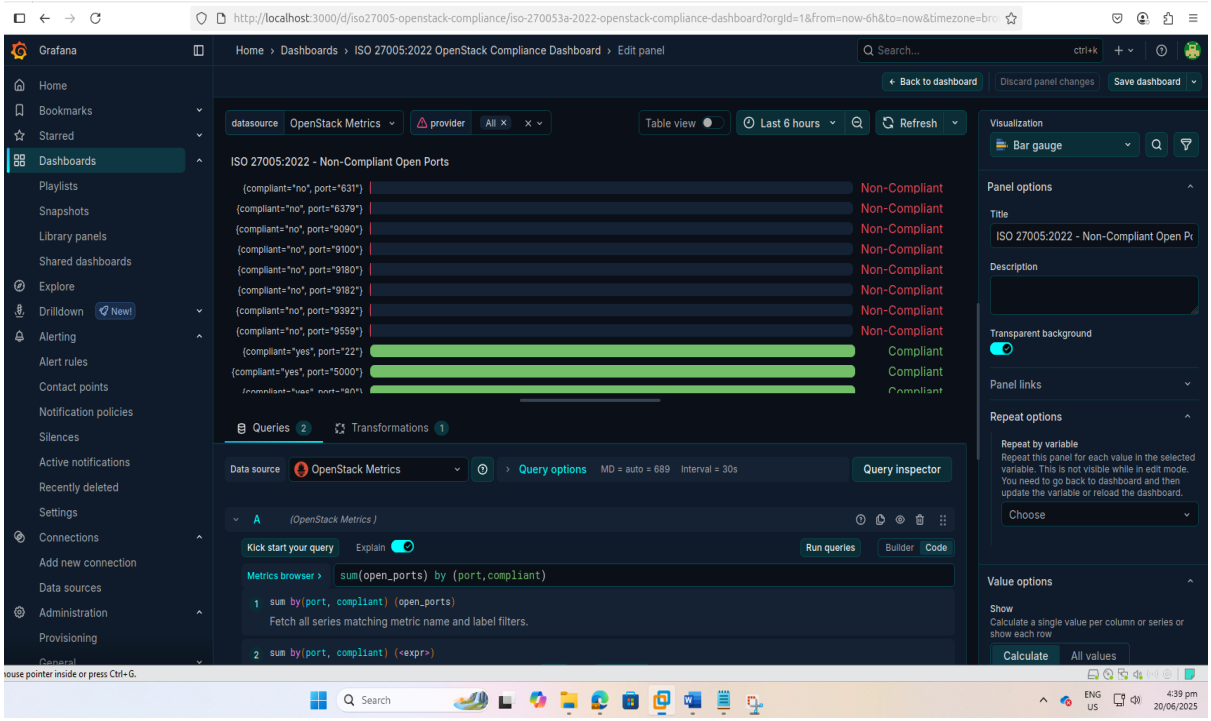
example:



ISO 27002:2022-Remote Administrative & Diagnostic Port Protection

This control makes sure that an unauthorized user cannot access a system by using open ports. Any ports that are not required for company operations must be disabled or removed.

If the system does not adhere to the ISO 27005:2022 standard, which safeguards remote diagnostic and configuration ports, the service provider has violated the control standard.



Registered Users/logs

The three users who have access to the system are the editor, the administrator, and the viewer. To maintain security, the system uses user authentication and role-based access control. Each user in the system belongs to the cloud provider that complies with legal criteria.