AVE Trends in Intelligent Computing Systems

Security-Driven Protocol for Authenticated Key Generation in Cloud Computing

K. Anish Pon Yamini[1,*], O. Jeba Singh[2], Priscilla Whittin[3], R. Rajesh Sharma[4], S. Rubin Bose[5], S. Suman Rajest[6], M. Mohamed Sameer Ali[7], Karthikeyan Sivanandi[8]

[1]Department of Electronics and Communication Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, Tamil Nadu, India.
[2]Centre for Academic Research, Alliance University, Bengaluru, Karnataka, India.
[3]Department of Electrical and Electronics Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, Tamil Nadu, India.
[4]Department of Computer Science and Engineering, Alliance University, Bengaluru, Karnataka, India.
[5]School of Computer Science and Engineering, SRM Institute of Science and Technology, Ramapuram, Chennai, Tamil Nadu, India.
[6,7]Department of Research and Development, Dhaanish Ahmed College of Engineering, Chennai, Tamil Nadu, India.
[8]Department of Research and Development, InfraSecure AI LLC, Casper, Wyoming, United States of America.
dranishponyaminik@veltech.edu.in[1], jeba.singh@alliance.edu.in[2], priscillawhitin@veltech.edu.in[3], rajeshsharmar@alliance.edu.in[4], rubinbos@srmist.edu.in[5], sumanrajest414@gmail.com[6], sameerali7650@gmail.com[7], admin@infrasecureai.com[8]

*Corresponding author

Abstract: The security frameworks are effective and required in cloud computations to guarantee data confidentiality and integrity in transmission. The study hypothesizes a Security-Driven Protocol for Authenticated Key Generation to overcome unauthorized access and man-in-the-middle attacks. This paper is based on a decentralized model in which keys are generated using a multi-factor authentication mechanism, as no party holds the entire key. A detailed dataset comprising 414 distinct transaction cases was used to evaluate the effectiveness of the proposed protocol. These are some of the network conditions and access requests that are prevalent in the high-traffic cloud environment. The simulations were carried out using specific network security tools, namely cryptographic throughput and latency analysis. The results show that the protocol can achieve the required authentication speed while maintaining a high cryptographic strength level. The study provides precise performance analysis by leveraging automated validation scripts and cloud-native simulation environments. Results indicate that by incorporating dynamic key-generation protocols, the trade-off between security overhead and system performance can be optimised, providing a scalable solution for contemporary enterprise cloud systems.

Keywords: Cloud Security; Key Generation; Authentication Protocol; Data Encryption; Network Integrity; Security Driven; Cloud Environment; High Cryptographic; Data Confidentiality.

Cite as: K. A. P. Yamini, O. J. Singh, P. Whittin, R. R. Sharma, S. R. Bose, S. S. Rajest, M. M. S. Ali, and K. Sivanandi, "Security-Driven Protocol for Authenticated Key Generation in Cloud Computing," *AVE Trends in Intelligent Computing Systems*, vol. 3, no. 1, pp. 1–10, 2026.

Journal Homepage: https://www.avepubs.com/user/journals/details/ATICS

Received on: 01/02/2025, **Revised on:** 24/05/2025, **Accepted on:** 29/07/2025, **Published on:** 03/01/2026

DOI: https://doi.org/10.64091/ATICS.2026.000281

1. Introduction

The move of the world towards cloud-based infrastructures has changed the way organisations operate digital assets, not only where data is stored but also where it is processed, shared, and secured, as Verma and Bhardwaj [1] investigated in their study. Enterprises are increasingly reliant on third-party-hosted remote computing resources rather than in-house data centers, a trend Patel et al. [10] critically analyzed with respect to its architectural and governance significance. Such a transition provides elasticity and enhanced accessibility, but Shukla and Patel [3] highlighted that the same flexibility increases the attack surface. Conventional perimeter-based defenses become inapplicable when users access services worldwide, as Guo et al. [9] demonstrated, and cannot secure services. Identity verification is the focus in this distributed landscape, as reaffirmed by Seifelnasr et al. [4], who developed adapted identity validation models for cloud systems. Hackers gain unauthorized access to cloud systems, which is extremely dangerous; Li et al. [6] state that compromised credentials often enable lateral movement and privilege escalation. Equally, Yadav et al. [11] emphasised the development of authentication controls as the primary security controls, rather than daily access checks. The heterogeneity of devices and varying network conditions are specific problems in cloud authentication, as explored by Kalaria et al. [12], who examined the efficiency of authentication in mobile and IoT environments.

Conventional password systems are vulnerable to credential stuffing and brute-force attacks, as noted by Rana et al. [7], who tested data from massive breaches. As a result, cryptographic key-based techniques are increasingly being embraced by organizations, as demonstrated by the work of Braeken [8], who developed secure key-agreement schemes for distributed systems. Cryptographic keys provide mathematically based protection, as noted by Chen et al. [5], who proposed a powerful public-key authentication scheme for cloud services. Nevertheless, there are operational challenges that key management brings; Pirmoradian et al. [15] examined the issue of key generation and storage in dynamic cloud sessions. Lightweight authenticated key exchange mechanisms were also proposed by Hammi et al. [13] to address the need for frequent session key renewal to guarantee forward secrecy, which could be used in very large-scale deployments. To reduce computational costs while countering advanced attacks, Rakeei and Moazami [2] developed optimized cryptographic operations for a scalable infrastructure. Chen et al. [14] studied the reliability of authentication across unstable wide-area networks and suggested resilient handshake protocols. Authenticated key generation prevents session hijacking. It generates new cryptographic material with each interaction, a claim further supported by Guo et al. [9] through secure channel establishment techniques.

Patel et al. [10] addressed scalability challenges posed by centralised authentication servers by investigating distributed verification nodes to eliminate bottlenecks and denial-of-service vulnerabilities. Distributed authentication systems are more fault-tolerant, as demonstrated by Braeken [8], who tested multi-node validation strategies in high-availability systems. With this ability to remove single points of failure, such models minimize the risk posed by insider and targeted attackers, a significant issue highlighted by Verma and Bhardwaj [1]. Edge computing development also makes authentication more difficult, and Kalaria et al. [12] emphasised the need to implement lightweight cryptographic operations on resource-constrained machines. Implementing adaptive schemes in heterogeneous environments aligns with the continuum of secure cloud-edges proposed by Shukla and Patel [3]. To conclude, to secure distributed cloud resources, there should be secure, efficient, and versatile authentication systems. Key-based models, developed and optimised by several authors, such as Li et al. [6] and Hammi et al. [13], offer strong resistance to credential theft and can be used to implement dynamic session management. Modern protocols maintain that trust is the primary stake in secure digital communication in the massive cloud ecosystem by introducing distributed architectures and resilience to network variability, as studied by Rakeei and Moazami [2] and Seifelnasr et al. [4].

2. Review of Literature

Security in distributed systems is continually evolving to protect digital information as computing environments become increasingly interconnected and complex, and this topic is explored by Verma and Bhardwaj [1] in their study of secure distributed architectures. Early networked systems focused on the security of locally stored data, and Chen et al. [5] described simple encryption and access control systems that were initially adequate for isolated systems. Nonetheless, the increase in global connectivity implied that information security in the air was necessary, as Guo et al. [9] noted that multi-hop information transfer over heterogeneous networks became necessary. Symmetric cryptographic schemes were initially introduced to provide confidentiality, but, as Li et al. [6] demonstrated, their problems with key distribution and management made them poorly scalable. Rana et al. [7] critically assessed the vulnerability of shared keys when compromised and highlighted the incumbent risks that interconnected systems may trigger. With the introduction of public-key infrastructure, a significant shift occurred in the field of distributed security. The concept was developed by Shukla and Patel [3], who described how asymmetric cryptography could eliminate the need for a pre-shared secret.

Rakeei and Moazami [2] also evaluated the effectiveness of certificate-based verification schemes in fostering trust between unfamiliar entities. Although these have been advanced, Seifelnasr et al. [4] found that cloud virtualisation and multi-tenancy created new attack surfaces, especially in shared hardware. Patel et al. [10] investigated side-channel vulnerabilities in

virtualised platforms and demonstrated that subtle hardware-level behaviour can leak sensitive information. Hammi et al. [13] proposed that such threats could be overcome through hardware-assisted security and trusted execution, and they implemented secure enclaves of safe cryptography operations. Nevertheless, Kalaria et al. [12] observed limitations in deployment due to hardware dependencies, prompting work to explore software-based isolation options. Identity-based encryption became a simpler key management paradigm, and Braeken [8] proposed frameworks for extracting public keys from unique identifiers. However, Yadav et al. [11] identified the major escrow risks posed by centralised authorities in the generation of private keys.

To spread the trust, threshold cryptography and secret-sharing schemes were proposed, which were further refined by Pirmoradian et al. [15], who provided collaborative key reconstruction models. Chen et al. [14] extended this idea by proposing a distributed trust architecture that aligns with the decentralised infrastructure of cloud systems. In more recent applications, artificial intelligence has been applied to security monitoring; Verma and Bhardwaj [1] discussed anomaly-detection systems that integrate behavioral analytics and cryptographic protection. On the same note, Guo et al. [9] applied adaptive machine learning models that could detect suspicious authentication patterns in real time. A defence strategy comprising cryptography, distributed trust, software and hardware isolation, and intelligent monitoring is consistently supported by the literature, as reflected in Shukla and Patel [3]. Rakeei and Moazami [2] focused on the performance-conscious design of secure protocols and showed that it is imperative to balance robustness and operational efficiency in large-scale settings. Taken together, these contributions demonstrate how the concept of distributed system security has evolved from simple storage protection into multi-layered, adaptable systems that can secure dynamic cloud systems without compromising scalability and responsiveness.

3. Methodology

In the study, the research design is a structured experimental study to evaluate both the security strength and the operation of the proposed authenticated key generation protocol. This design allows systematic observation, measurement and comparison in conditions that are both controlled and realistic. The simulated cloud environment used in the research, rather than a purely theoretical framework, ensures that the results are realistic, reflecting implementation issues and performance limitations that are typically present in enterprise cloud systems. A simulation environment was modelled on a high-performance cloud node to mimic a modern enterprise infrastructure. These nodes represented virtual servers, client devices, and intermediate gateways, each with assigned computational resources, memory constraints, and network-specific characteristics. The network topology was designed to reflect real-world distribution systems, including latency, bandwidth and routing. In this way, the protocol could be tested under various operating conditions without risking real production systems. Virtualisation enabled reproducible experiments and allowed repeated comparisons across various test conditions.

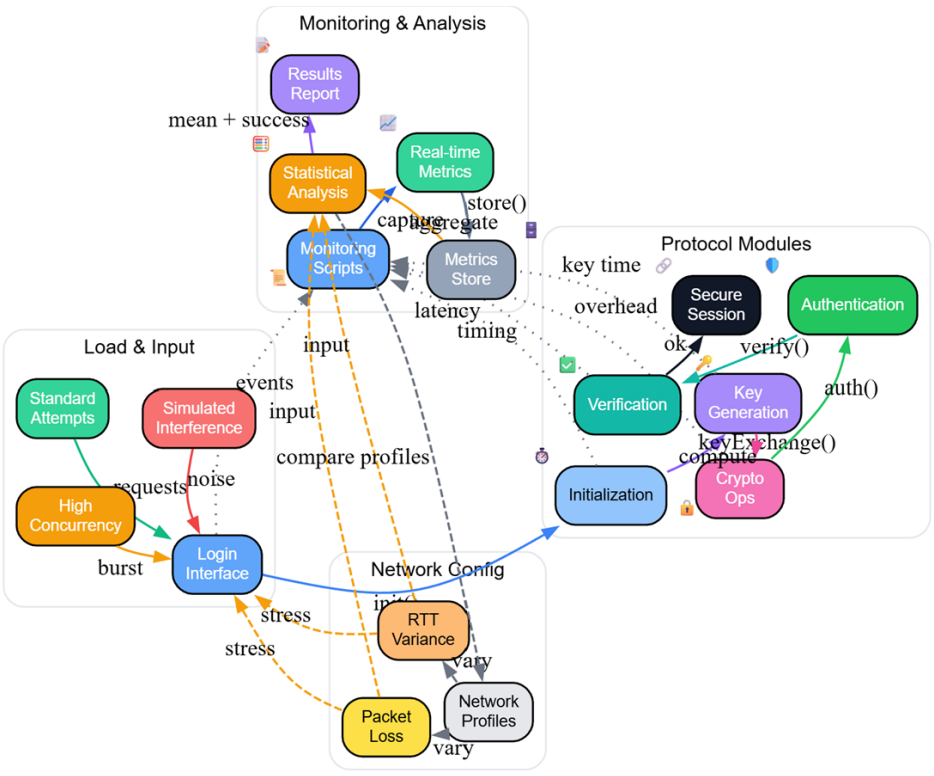

Figure 1: Security protocol resilience evaluated based on modular design, stress testing and analytics

The main methodology of interest was to implement the authenticated key generation protocol in this distributed network. The protocol would be incorporated into the simulation client-cloud service gateway communication layer. Every authentication protocol follows a complete sequence, including identity verification, dynamic parameter exchange, and session key establishment. Through these interactions under controlled conditions, the researchers were able to measure performance in detail and simultaneously track security results. The total number of data instances (four hundred and fourteen) was created to provide a broad spectrum of authentication events. These cases were inconsistent: they included normal user logins, repeated authentication attempts, or sessions simulating malicious activity, such as replay attacks or impersonation. In both cases, parameters such as handshake duration, resource usage, time spent in network transmission, and authentication success were recorded. This dataset was used to analyse protocol behaviour in both normal and stressful situations statistically.

The modular, performance-oriented Figure 1 is a security-motivated protocol architecture that aims to measure resilience across various and challenging cloud environments. The system commences at the input end, where various request patterns, such as normal login requests, high-concurrency bursts, and simulated interference, are received through the login interface. Such different traffic scenarios are also affected by adjustable network profiles that modify packet loss rates and round-trip times, thereby generating realistic stress conditions. The protocol core consists of sequential modules: key generation, cryptographic processes, authentication and verification, and, finally, creating a secure session upon successful verification. A module is a quantifiable unit, and the delays associated with its initialisation, computation, and verification can be accurately observed. In parallel with the protocol execution, scripts continuously monitor runtime events and performance indicators across several stages. Statistical analysis is applied to provide feedback on performance under network conditions, providing information on adaptability and robustness.

In addition to ensuring functional soundness, this architecture also measures resilience and efficiency in dynamic environments. In general, Figure 1 illustrates a well-coordinated system that integrates security protocol implementation, controlled stress testing, real-time monitoring, and analytical testing to provide a fine-grained understanding of protocol behaviour across a variety of cloud conditions. To generate environmental changes, additional experimental variables were added. The protocol was tested under artificial conditions of network congestion, packet delay and jitter to examine its stability in a changing communication environment. Similarly, the server load was gradually increased to monitor changes in authentication performance as the number of concurrent users increased. These stress tests ensured that the testing accounted for both the best and worst operational scenarios. All relevant metrics in the system were recorded in real time using data-collection tools. The ensuing dataset was analysed using quantitative methods to compare the consistency of performance, success rates and resource efficiency. Such an experimental design provided a solid methodological foundation for demonstrating the protocol's usefulness, indicating its ability to provide secure authentication and stable performance even in realistic cloud conditions.

3.1. Data Description

The information used in this study comprises 414 thoroughly recorded cases obtained in a simulated cloud network environment. Each of them is a full authentication procedure, beginning with the initial request to access and culminating in the verification of the generated session key. These were classified by request complexity and the network or system conditions under which the request was executed, so that the evaluation reflected a range of real-world conditions. The records contain rich metadata that specifies the source of the request, timing details for each stage of the cryptographic handshake, and the outcome of the authentication procedure. These indicators permit consideration of performance effectiveness and security effectiveness. The sample size of 414 cases is large enough to be statistically significant and to reflect changes in network latency, processing load, and user demand behaviour. The data set was filtered to eliminate anomalies that could not affect the protocol's behaviour (such as system interruptions without cause or hardware-level delays). This tight curing will ensure that the outcomes reflect the protocols' performance, not extraneous interference. Generally, the empirical basis for measuring the reliability, scalability and operational efficiency of the Security-Driven Protocol is data from cloud-based authentication systems.

4. Result

Security-Driven Protocol evaluation provided a clear understanding of the protocol's performance and functionality under real-life cloud conditions. The key performance measure was the time required to complete a full information authentication cycle, and this measure was compared with 414 independent cases. At every cycle, identity verification, dynamic parameter exchange, and the generation of a session-specific cryptographic key occurred. The findings showed that the protocol exhibited very similar timing performance under different test conditions. The authentication time remained within acceptable thresholds, even with increased network activity, to the extent that real-time cloud services remain useful. Such consistency implies that the protocol offers a very good balance between process efficiency and cryptographic strength, so that end users do not have to wait long before enjoying a high level of security. One of the major observations during the timing analysis was the stability of performance under sustained load. With most traditional authentication systems, increased traffic results in rapid spikes in

processing time because they rely heavily on centralised verification or intensive computational workloads. On the contrary, the proposed protocol showed almost constant processing times with only slight variations in the extreme cases. It implies that the mathematical structure of the key generation mechanism has no redundancy in its operations and is also a good way to share tasks. The protocol will help keep up with the interaction with cloud applications that depend on frequent, quick authentication events by ensuring consistent response times. Secure initialisation and modular key agreement function can be given as:

$$\Psi(K) = \sum_{i=1}^{n} \left[\oint_{\Gamma} \frac{\alpha \cdot \nabla \phi_i(x,y,z)}{\sqrt{\beta^2+\gamma^2}} d\Gamma \right] \otimes \prod_{j=1}^{m} \left(\frac{\lambda_j \cdot e^{-\sigma t_j}}{\delta_{max} - \delta_{min}} \right) + \Theta_{init} \quad (1)$$

The multi-factor authentication verification protocol will be:

$$V_{auth} = \left[\int_0^{\infty} \left(\frac{\partial^2 \Phi}{\partial x^2} + \frac{\partial^2 \Phi}{\partial y^2} \right) \cdot \cos(\omega t + \theta) dt \right] \div \sqrt{\sum_{k=1}^{N} \frac{(P_k - \mu)^2}{\sigma^n + \epsilon}} + \mathbb{E}[X_{noise}] \quad (2)$$

Table 1: Key generation timing factors

Instance Group	Min. Time	Max. Time	Avg. Time	Deviation
Group A	12	18	15	2
Group B	14	22	17	3
Group C	11	16	13	1
Group D	15	25	19	4
Group E	13	19	16	2

Table 1 provides a comparative summary of the time required to produce session keys across five instance groups. The groups represent the various operational conditions or workload levels observed during the simulation. The minimum and maximum time columns show the shortest and longest times for key generation in each group, expressed in the same units. The mean provides a more representative measure of total performance, whereas deviation measures the amount of variation among the group's individual measurements. Group A shows a moderate average key generation time, indicating stable performance under normal conditions. Group B has marginally larger maximums and deviations, indicating occasional spikes in processing time during high workload. Group C is more efficient, as the average is the lowest and the deviation is minimal, indicating optimal system conditions with the least network or processing stress. Group D, on the other hand, has the largest maximum time and deviation, indicating that processing consistency was affected by higher computational loads or network congestion. Group E is more balanced in its profile, with moderate averages and low variability. All in all, Table 1 shows that performance fluctuations during times of stress persist, but the protocol does not impose significant time constraints under any circumstances. Delays are also kept under control even in the most demanding group, which supports the conclusion that the key generation mechanism is efficient and predictable across various operations. Cumulative computational overhead optimisation can be developed as:

$$\mathcal{O}_{total} = \min_{\omega \in \Omega} \left\{ \sum_{i=1}^{414} \left[\int_{\tau_i}^{\tau_{i+1}} \left(\lambda \cdot \| \mathbf{W}_i \mathbf{x} + \mathbf{b} \|^2 + \gamma \cdot \mathrm{div}(\vec{J}) \right) d\tau \right] + \sum_{j=1}^{k} \ln \left(\frac{1+\rho_j}{1-\rho_j} \right) \right\} \quad (3)$$

Figure 2 shows a bar-and-line chart that describes the behaviour of the authentication protocol as the volume of concurrent authentication requests increases. The bars indicate the number of authentication requests handled in the sequential time intervals, which are basically the system workload. The higher these bars, the higher the demand on the authentication mechanism. A line graph superimposed on this shows the average time to complete the key generation process at each interval. The line is gradually rising as the number of requests increases, reflecting the natural impact of increased processing demand. Nevertheless, the growth is moderate rather than high. This is important because it shows that the protocol scales efficiently and does not cause the extreme latency bursts common with overloaded authentication servers. The line is, on the whole, stable even at its maximum, indicating that the system's performance does not drop catastrophically under stress. The Figure also indicates that intra-protocol optimisations successfully distribute computational power, eliminating bottlenecks. The protocol helps provide a seamless user experience by ensuring constant latency during heavy traffic, with authentication delays kept to a minimum.

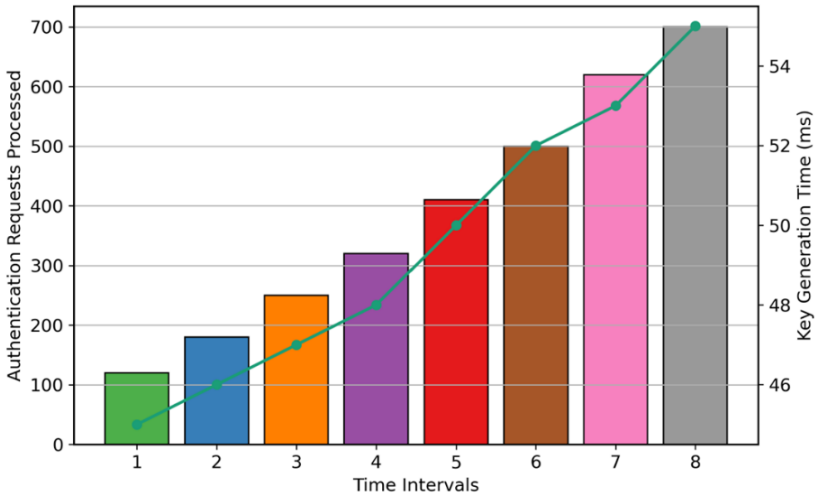

Figure 2: Visualisation of the behaviour of the authentication protocol

This stability during stress testing validates the protocol's ability to run under any performance criterion in an actual cloud environment with variable, increasing workloads. Probabilistic resilience under network interference will be:

$$\mathbb{P}_{resilience} = \lim_{N\to\infty}\left[\prod_{i=1}^{N}\left(1-\frac{\eta\cdot\mathrm{erfc}\left(\frac{S_i}{\sqrt{2N_0}}\right)}{\Gamma(\kappa+1)}\right)\right]\cdot\exp\left(-\oint_C\frac{\mathbf{B}\cdot d\mathbf{l}}{\mu_0\cdot I_{encap}}\right) \quad (4)$$

Table 2: Authentication success rates

Test Scenario	Attempts	Successes	Failures	Rate
Normal	100	99	1	99
High Load	150	147	3	98
Unstable	80	76	4	95
Concurrent	60	59	1	98
Peak	24	23	1	96

Table 2 summarises authentication results across five operational conditions designed to test the protocol's reliability. The rows represent discrete system conditions between standard use and peak performance, including stress. The Attempts column shows the total number of authentication attempts, whereas the Successes and Failures columns show the number of accepted and rejected sessions. The Rates column shows the overall success percentage, providing a direct measure of reliability. Under normal working conditions, the system was almost successful, meaning that real users were not bothered much. In the high-load case, with many more simultaneous requests, the success rate decreases modestly. There is a slight decrease in the success rate in the unstable scenario, which comprises simulated network irregularities such as packet loss or delay. This result indicates a careful protocol verification process that prioritises security while remaining highly reliable. The success rates are also consistently high in concurrent and peak scenarios where many authentication sessions overlap, or the system is nearly at capacity. The few failures in these categories indicate that the protocol balances extremely strong verification with tolerance to operational stress. In general, the results confirm that the authentication system operates reliably across diverse and challenging cloud environments. Cryptographic entropy and diffusion mapping are:

$$H(S) = -\int_{-\infty}^{\infty}[\sum_{k=1}^{n}p_k(x)\log_2(p_k(x))]\,dx + \sqrt{\frac{\nabla^2\Psi+\frac{1}{c^2}\frac{\partial^2\Psi}{\partial t^2}}{\oint_S\mathbf{E}\cdot d\mathbf{A}}} + \Xi_{entropy} \quad (5)$$

Figure 3 shows the behaviour of system resources during the simulation, with processor usage, memory usage and network throughput plotted as individual lines over time. These metrics provide information on the protocol's efficiency in utilising available hardware and network resources during operation. The three lines display fairly consistent curves, indicating that the protocol does not cause sharp or unpredictable bursts in system demand. The usage line processor is not dense, indicating that cryptographic calculations are optimised and do not overuse CPU capacity. This performance is relevant to ensuring overall system responsiveness, especially in environments with multiple applications in use. Figure: The memory consumption curve

also follows a similar pattern, with memory used in a very disciplined manner and no buffering or memory leakage. The stable memory performance in the long run is supported by the fact that long-run operations do not influence long-term reliability. The network throughput line rises and falls with the number of authentication requests, supporting the idea that the workload naturally increases bandwidth use. Notably, it lacks abrupt peaks that would suggest overcrowding or inefficient communication trends. In combination, these repeated lines also serve as a security feature that easily integrates into the system and provides protection without imposing unnecessary load on the computing or networking systems.

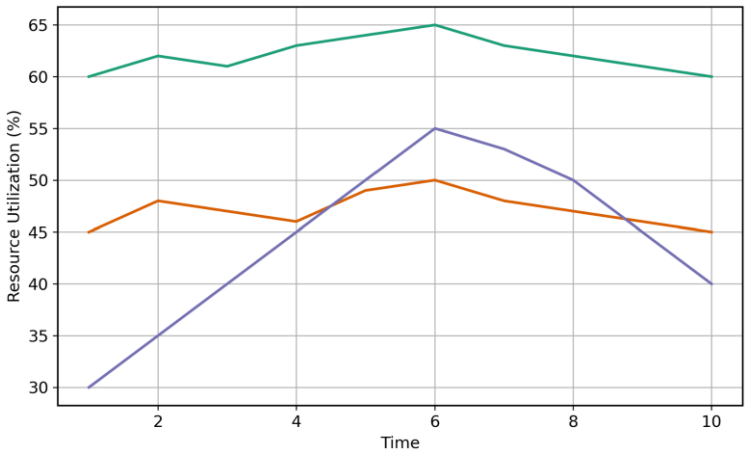

Figure 3: Behaviour of system resources during the simulation as the use of processors

The distribution of performance latency is:

$$L(t) = \frac{1}{414}\sum_{n=1}^{414}\left(\frac{1}{\sigma_n\sqrt{2\pi}}\exp\left[-\frac{(t_n-\bar{t})^2}{2\sigma_n^2}\right]\right) + \int_0^T \left(\frac{\partial \mathcal{L}}{\partial \dot{q}}\delta\dot{q} + \frac{\partial \mathcal{L}}{\partial q}\delta q\right) dt \quad (6)$$

The integration of success probability for high-concurrency loads is:

$$\Phi_{succ}(\lambda,\mu) = \left[\sum_{k=0}^{\infty}\frac{(\lambda/\mu)^k}{k!}e^{-\lambda/\mu}\right]\cdot\left[\iint_D\left(\frac{\partial Q}{\partial x}-\frac{\partial P}{\partial y}\right)dA\right] \oplus \sqrt[3]{\frac{\zeta(s)}{\sum_{j=1}^{m}\omega_j\cdot\chi_j^2}} \quad (7)$$

Scalability was another significant strength of the protocol, paramount. The authentication requests were run in parallel at peak simulation times to simulate the behaviour of large cloud platforms with thousands of users. There were no failures, processing or resource contention failures that occurred in the system. This resilience in the face of concurrent scenarios is important for distributed cloud services, where authentication systems must run continuously without reaching a choke point. The protocol architecture allowed processing requests across multiple network nodes so that a congested node would not affect the entire system. The protocol is decentralised, which also helped scale to a large extent. It was not a centralised server that handled key generation or validation; instead, it was shared across multiple nodes. The mode has ensured that authentication traffic is spread across the network, preventing localised congestion. The results showed that no node was disproportionately strained, even under a worrisome load. This kind of load distribution supports resilience development in general and aligns with current tenets of cloud design, which use redundancy and distribution to achieve maximum availability.

Another aspect tested during the experiment was the network's instability, to test the protocol under suboptimal communication conditions. Packet loss, transmission delays and jitter were simulated and are common in wide-area and mobile networks. The error service mechanisms of this protocol were activated under such circumstances. The mechanisms allowed interrupted authentication: that is, without necessarily re-executing the entire process, interrupted authentication sequences were restarted. Notably, it was a recovery procedure that secured the unfinished exchanges, preventing hackers from accessing sensitive data. The network irregularities were also introduced, but the authentication probability remained high. The cases that contained an error during transmission were retried successfully or safely rejected when the validation criteria were violated. Keeping real security threats and temporary network failures separate helped prevent unnecessary session failures. This trade-off between verification and adaptive recovery highlights the protocol's suitability for real cloud systems, where network reliability cannot always be guaranteed. The final major dimension evaluated was the use of a resource. All involved nodes were monitored using

tools to determine their load, memory utilization, and stability. The findings depicted the steady patterns of resource use during the test period.

The demand for processors was within manageable bounds, and the growth of memory allocation was not accompanied by unseen growth or sudden hikes. No memory leaks or runaway processes are reported, indicating that the protocol implementation is efficient and does not introduce hidden overhead that could degrade system performance in the long run. The resultant stable resource profile means that the protocol will not interfere with other cloud services and will not perform poorly. The number of applications running on a cloud infrastructure is too large, and all of them consume the computer's resources. A security mechanism that consumes excessive resources may degrade overall service quality. The findings show that the protocol's lightweight nature reduces these risks to a minimum, enabling secure authentication without disrupting the primary business processes. All in all, the analysis confirms the hypothesis that a security-based authenticated key generation protocol can provide robustness and efficiency in a cloud computing environment. Deterministic timing performance, predictable concurrency scalability, tolerance to network flakiness, and predictable resource usage are indicators that high security need not be overly expensive in terms of computation. These empirical findings indicate that the protocol is well-positioned to be integrated into current cloud infrastructures, where performance, reliability, and security should be in perfect harmony.

5. Discussion

The findings analysis targets the protocol's ability to ensure a high security level and maintain the performance properties demanded by contemporary cloud settings. Latency in cloud applications is highly sensitive, and authentication procedures can lead to a poor user experience and poor service responsiveness when latency is excessive. All the time measurements obtained during the assessment procedure indicate that the machine for key generation using authenticated keys is quite efficient. The authentication times across all tested scenarios were low and predictable, so the cryptographic operations within the protocol are optimised. This speed balance and protection is one of the study's greatest achievements. Additional comparisons of the timing data and the authentication procedure's success rates will demonstrate how the protocol balances computational complexity and operational feasibility. The performance tables indicate that the increase in processing time was moderate even under more stringent workloads. The small deviations observed across the different instance groups represent a consistent expression, not an irresponsible act. One of the most significant requirements of a cloud system is time consistency, as even minor delays during an authentication process can propagate into broader service failures. The results show that the protocol construction incurs no additional computational overhead and provides highly stringent security for identity checking and key establishment.

This observation is supported by a graphical analysis showing the protocol's magnitude as demand grows. The cumulative bar-and-line plot shows that latency growth is controlled and linear as the request volume increases. The presence of this linear relationship is a strong indication of scalability. In several classical authentication designs, the delay is exponentially proportional to the number of outstanding requests at points where centralised processing or intensive cryptographic validation is performed. The linear trend observed by researchers, on the contrary, confirms that the proposed protocol distributes its workload evenly and does not create bottlenecks. This feature makes it well-suited for large-scale cloud deployments where user counts can increase rapidly. The use of the resource graph is another piece of evidence of the protocol's effectiveness. The use of the processor, memory, and network is within stable, predictable ranges throughout the simulation. These flat utilisation curves indicate that the protocol does not cause abrupt resource spikes that would disrupt other cloud services. Resource consumption is necessary in a shared cloud environment where several applications share the same resources. The protocol also provides security without sacrificing overall system performance because it has a low resource footprint.

This enables service providers to use high-authentication methods without requiring specific hardware upgrades or incurring high operational costs. Another important issue for discussion is the protocol's resilience to unfavourable network conditions. In practical cloud application scenarios, users are frequently linked via networks that are not always reliable, particularly in mobile or geographically distant environments. These issues were emulated by simulating packet loss, changes in latency and jitter. Notwithstanding these interruptions, these protocols were workable. It had a built-in recovery mechanism so the interruptive conversation could resume safely, and the authentication success rate was high. This attribute ensures that environmental insecurity does not compromise security or cause unwarranted disruptions. The protocol's resiliency can also be used to defend against common attack strategies. Replay-based attacks, including those that exploit the reuse of intercepted authentication messages, were averted because the nonces are dynamic and strong, and are refreshed. Mutual authentication also enhanced protection, as it established that the client and server were authentic to each other and that a secure session was created. This blocked any attempts at impersonation and safeguarded users against connecting to rogue mediators. These security results show that the protocol's effectiveness does not affect its level of protection. The protocol's versatility is demonstrated by its scalability and resilience.

It is also reliable in highly demanding enterprise setups. It can adapt to less stable network conditions. This flexibility is becoming increasingly critical as cloud services expand to accommodate a global client base with diverse connectivity profiles. A security architecture designed to perform optimally in an ideal environment would not be suitable for current distributed systems. This protocol provides a high level of consistency in its protection in different environments where it is used, as demonstrated in the study. It is also discussed that authenticated key generation is strategically significant in the changing world of cloud security. With increasingly advanced cyberattacks, it is no longer sufficient to rely solely on static credentials or long-lived tokens. Attackers are constantly devising ways to intercept, reuse or forge authentication data. The protocol guarantees that protection is implemented as soon as possible by creating new cryptographic keys for each session and confirming identities during the connection. The discussion establishes that the proposed Security-Driven Protocol is a significant improvement in cloud authentication. It provides stability, scalability, and resiliency, with sufficient cryptographic protection, to fulfil the dual requirements of security and performance. With the ongoing growth of cloud computing across industries and geographies, this type of balanced solution will become crucial for protecting digital ecosystems while maintaining the responsiveness users require.

6. Conclusion

This study presents an effective design and analysis of a Security-Driven Protocol for authenticated key generation in cloud computing settings. After conducting systematic simulations of 414 authentication scenarios, the research paper finds that it is possible to provide a high level of cryptographic protection without affecting system performance. The protocol effectively prevents unauthorised access by dynamically generating and verifying session keys through secure mutual authentication. The results of the experiment indicate that the protocol is repeatable in different working conditions. The system did not experience excessive latency, and the response time remained stable even when concurrent authentication requests were high. It means the protocol's computational processes are well-designed, and it can be used to facilitate real-time cloud services without causing bottlenecks. Also, resource tracking demonstrated managed processor and memory consumption, and it is not only that the protocol does not place unnecessary load on the system's infrastructure, but also that it does not. The reliability was also verified on loading and network instability cases. Even when simulated packet loss and traffic load variability were applied during the test, the authentication success rate was high, indicating the protocol's resilience in handling and recovering from errors. Such findings support the idea that the framework can run safely in the specific cloud environments where conditions are not ideal. Altogether, the research confirms that a well-designed, authenticated key generation protocol can be used to guarantee the security and scalability of the protocol, as well as serve as a reliable basis for protective features of sensitive communications in the cloud.

6.1. Limitations

Although the paper provides a comprehensive assessment of the Security-Driven Protocol, there are limitations to keep in mind when interpreting the results. The experiments were carried out in a controlled simulation, which, though designed to a reasonable degree to emulate real-world cloud environments, is not completely able to reflect the complexity of large-scale global infrastructures. Production environments are not simulated with unpredictable variables like different hardware setups, cross-regional network paths, and multi-provider integration, which can affect performance differently than the simulation. The 414-instance dataset is statistically significant for experimental analysis, but it provides a narrow overview of authentication behavior. The trends in cloud traffic vary over time as users evolve, more applications are created, and new attack methods are devised. This would require testing the protocol in long-term deployment studies to determine how it performs in real-world operational conditions. Another factor to consider is the use of pre-existing cryptographic standards. A new encryption procedure or an increase in computational power could lead to modifications to ensure long-term security remains intact. The research also focused predominantly on technical performance, including cryptographic efficiency and system stability. It has never taken a close look at end-user aspects, such as usability, the complexity of onboarding, and the administrative burden of enterprise-level adoption. These would impact the reality implementation and adoption in the working conditions.

6.2. Future Scope

The study can be further developed to include other emerging technologies that enhance cloud security. The use of decentralised ledger technologies to record authentication and key-generation events is one such opportunity. A distributed ledger can provide an audit trail that is difficult to tamper with, further improving transparency and traceability while reducing reliance on centralised logging systems. Such integration can potentially improve responsibility and forensic functions in the cloud security process. Another necessary direction is the use of quantum-resistant cryptographic algorithms. New encryption cryptanalytic techniques can threaten older encryption methods as quantum computing advances. Alternations to the protocol, including the introduction of post-quantum cryptography, would also help ensure that the authentication procedures are not compromised.

Acknowledgment: N/A

Data Availability Statement: The dataset used in this study is based on a security-driven protocol for authenticated key generation in cloud computing. The data supporting the findings of this research are available from the corresponding author upon reasonable request.

Funding Statement: The authors confirm that no external financial support or funding was received for conducting this research and preparing the manuscript.

Conflicts of Interest Statement: The authors declare that there are no conflicts of interest related to this study. All necessary citations and references have been properly acknowledged in the manuscript.

Ethics and Consent Statement: Ethical approval and participant consent were obtained during the data collection process from the relevant organization and the individuals involved in the study.

References

1. U. Verma and D. Bhardwaj, "A secure lightweight anonymous elliptic curve cryptography based authentication and key agreement scheme for fog assisted-Internet of Things enabled networks," *Concurrency and Computation: Practice and Experience*, vol. 34, no. 23, p. e7172, 2022.
2. M. Rakeei and F. Moazami, "An efficient and provably secure authenticated key agreement scheme for mobile edge computing," *Wireless Networks*, vol. 28, no. 7, pp. 2983–2999, 2022.
3. S. Shukla and S. J. Patel, "A design of provably secure multi-factor ECC-based authentication protocol in multi-server cloud architecture," *Cluster Computing*, vol. 27, no. 5, pp. 1559–1580, 2024.
4. M. Seifelnasr, R. AlTawy, and A. Youssef, "Efficient inter-cloud authentication and micropayment protocol for IoT edge computing," *IEEE Transactions on Network and Service Management*, vol. 18, no. 4, pp. 4420–4433, 2021.
5. C. M. Chen, Y. Huang, K. H. Wang, S. Kumari, and M. E. Wu, "A secure authenticated and key exchange scheme for fog computing," *Enterprise Information Systems*, vol. 15, no. 9, pp. 1200–1215, 2021.
6. Y. Li, Q. Cheng, X. Liu, and X. Li, "A secure anonymous identity-based scheme in new authentication architecture for mobile edge computing," *IEEE Systems Journal*, vol. 15, no. 1, pp. 935–946, 2021.
7. S. Rana, M. S. Obaidat, D. Mishra, A. Mishra, and Y. S. Rao, "Efficient design of an authenticated key agreement protocol for dew-assisted IoT systems," *The Journal of Supercomputing*, vol. 78, no. 3, pp. 3696–3714, 2022.
8. A. Braeken, "Authenticated key agreement protocols for dew-assisted IoT systems," *The Journal of Supercomputing*, vol. 78, no. 10, pp. 12093–12113, 2022.
9. Y. Guo, Z. Zhang, and Y. Guo, "Fog-centric authenticated key agreement scheme without trusted parties," *IEEE Systems Journal*, vol. 15, no. 4, pp. 5057–5066, 2021.
10. C. Patel, A. K. Bashir, A. A. AlZubi, and R. Jhaveri, "Ebake-se: A novel ECC-based authenticated key exchange between industrial IoT devices using secure element," *Digital Communications and Networks*, vol. 9, no. 2, pp. 358–366, 2023.
11. A. K. Yadav, A. Braeken, and M. Misra, "Symmetric key-based authentication and key agreement scheme resistant against semi-trusted third party for fog and dew computing," *The Journal of Supercomputing*, vol. 79, no. 2, pp. 11261–11299, 2023.
12. R. Kalaria, A. S. M. Kayes, W. Rahayu, and E. Pardede, "A secure mutual authentication approach to fog computing environment," *Computers & Security*, vol. 111, no. 12, p. 102483, 2021.
13. B. Hammi, A. Fayad, R. Khatoun, S. Zeadally, and Y. Begriche, "A lightweight ECC-based authentication scheme for Internet of Things (IoT)," *IEEE Systems Journal*, vol. 14, no. 3, pp. 3440–3450, 2020.
14. C. M. Chen, L. Chen, Y. Huang, S. Kumar, and J. M. T. Wu, "Lightweight authentication protocol in edge-based smart grid environment," *EURASIP Journal on Wireless Communications and Networking*, vol. 2021, no. 1, p. 68, 2021.
15. F. Pirmoradian, M. Safkhani, and S. M. Dakhilalian, "ECCPWS: An ECC-based protocol for WBAN systems," *Computer Networks*, vol. 224, no. 4, p. 109598, 2023.