



Secure Authentication Mechanisms for Cloud-Based Computing Environments

Rezaul Wijesekara

Department of Structural Engineering, Borneo School of Business and Technology, Malaysia

rezaul.wijesekara@bsbt-my.org

Peer Review Information

Submission: 18 Jan 2018

Revision: 03 Feb 2018

Acceptance: 18 March 2018

Keywords

Cloud computing, Multi-factor authentication, Identity and access management, Elliptic-curve cryptography, Access control

Abstract

Cloud computing allows users to obtain software, platforms, storage and processing resources on demand, but the same distributed and multi-tenant model creates serious authentication risks. Password-only login is inadequate when credentials are stolen, replayed, guessed or used from an untrusted device. This methodology paper develops a secure authentication framework for cloud-based computing environments by integrating primary credentials, device-bound cryptographic proof, risk-aware multi-factor authentication, mutual verification, session-key establishment and role-based authorization. The method is derived from peer-reviewed research published up to 2018 and is evaluated through a structured threat model and an analytical security-cost comparison. The proposed design applies nonce-based freshness, elliptic-curve key agreement, step-up one-time-password or biometric verification, short-lived access tokens and centralized revocation. Analytical results show that the framework covers eight major authentication and access-control requirements while keeping the normal login path lighter than an always-on high-cost multi-factor design. The study concludes that layered, adaptive authentication provides a better security-usability balance than a single authentication factor and is suitable as a reference methodology for cloud applications requiring scalable access control and stronger resistance to credential compromise.

Introduction

Cloud computing changes the location and ownership of computational resources: data and applications are hosted on infrastructure that is accessed remotely and often shared among many tenants. This model improves flexibility and scalability but reduces the effectiveness of security assumptions based on a trusted local perimeter. Takabi, Joshi and Ahn (2010) identified identity management, access control and privacy as central obstacles to trustworthy cloud adoption, while Subashini and Kavitha (2011) showed that security risks differ across SaaS, PaaS and IaaS service models. Zissis and Lekkas (2012) further argued that cloud security requires explicit mechanisms for authentication,

confidentiality, integrity and trusted service relationships rather than simple migration of traditional controls.

Authentication is the first control that determines whether a claimed user, device or service is legitimate. In a cloud environment, the authentication decision must remain dependable even when requests arrive from changing locations, browsers and devices. It must also cooperate with authorization so that a successfully authenticated user receives only the privileges required for the requested service. Broad surveys have shown that multi-tenancy, outsourcing, malicious insiders, account hijacking and exposed interfaces expand the

attack surface of cloud systems (Khalil, Khreishah, & Azeem, 2014). Consequently, a practical mechanism must address not only password verification but also replay resistance, mutual authentication, session protection, user privacy, revocation and efficient scaling.

Identity and access management (IAM) provides the organizational layer for this process. Indu, Anand and Bhaskar (2018) reviewed cloud IAM mechanisms and emphasized that authentication, authorization and identity lifecycle management must operate together. Access-control research similarly demonstrates that authenticated identity alone is not enough: policies must decide what that identity can do after login (Karatas & Akbulut, 2018). These findings motivate an integrated methodology in which authentication assurance; cryptographic session establishment and authorization are treated as one continuous access decision.

The objective of this paper is therefore to design and evaluate a layered authentication framework that is secure against common credential and network attacks while remaining usable for normal cloud access. The proposed mechanism uses a low-friction normal path and invokes additional factors only when the context is risky. It combines password verification with device-bound elliptic-curve proof, fresh nonces, adaptive one-time-password or biometric verification, a derived session key and a short-lived authorization token. The paper is methodological: the results are an analytical evaluation of security properties and normalized computational overhead rather than production benchmark measurements.

Literature Review

Indu et al. (2018) presented a comprehensive review of identity and access management in cloud environments. Their study explains that cloud users interact with resources across organizational and service boundaries, making authentication, federation, authorization and identity lifecycle control essential. The work is especially relevant to the present methodology because it treats IAM as an architecture rather than as a single login operation. It supports the inclusion of registration, verification, authorization and revocation phases in a unified cloud authentication design.

Wang et al. (2018) examined user authentication based on elliptic-curve cryptography (ECC) and showed why cloud protocols must protect user anonymity, forward secrecy and resistance to offline password guessing. Their analysis also stresses that a public-key primitive must be

deployed correctly: adding cryptography without an appropriate adversary model does not automatically make a protocol secure. This work motivates the use of ephemeral ECC key agreement and explicit attack-oriented validation in the proposed framework.

Karatas and Akbulut (2018) surveyed cloud access-control mechanisms and compared multiple approaches for controlling resource use after identity verification. Their study demonstrates that authentication and authorization should be linked because a legitimate identity may still become a security risk if excessive privileges are granted. The proposed methodology therefore ends successful authentication with policy-based authorization and limits the lifetime and scope of the issued access token.

Namasudra and Roy (2017) proposed a secure cloud authentication scheme using two-factor security and cryptographic techniques intended to provide mutual authentication, anonymity, untraceability and forward security. The study illustrates the security value of combining more than one credential and of refreshing session secrets. These properties are incorporated into the proposed design through a device-bound factor, optional step-up factor and per-session key derivation.

Wazid et al. (2016) introduced a biometric-based user authentication and key-agreement scheme for cloud computing. Their work targeted impersonation, replay and privileged-insider attacks while supporting user anonymity and credential update. It demonstrates that a biometric can strengthen authentication when it is used as an additional factor rather than transmitted or stored as a simple password substitute. In the present methodology, biometrics are treated as an optional local step-up factor; the cloud receives only a verified assertion or cryptographic result, not raw biometric data.

Amin et al. (2016) studied anonymous authentication for distributed mobile cloud environments and emphasized privacy-aware access from resource-constrained devices. Their work is important because cloud authentication often occurs from mobile or heterogeneous clients that cannot sustain heavy cryptographic work on every request. This observation supports an adaptive design in which expensive operations are reserved for registration, device binding or elevated-risk sessions instead of being repeated unnecessarily.

Moghaddam et al. (2014) proposed a scalable authentication scheme that separates authentication and cryptographic functions from primary cloud application servers. This separation reduces coupling between business services and security logic and can improve manageability. The present framework similarly places the identity provider, risk engine and authorization layer in front of cloud services so that applications consume a verified, short-lived identity token rather than implementing independent password databases.

Khalil et al. (2014) surveyed cloud threats and security controls, highlighting insider attacks, account compromise, insecure interfaces and virtualization-related risks. Their broad threat taxonomy supports a defense-in-depth approach because no single authentication factor can address every cloud threat. Zissis and Lekkas (2012) likewise proposed a trusted security layer involving public-key infrastructure, single sign-on and directory services, while Takabi et al. (2010) and Subashini and Kavitha (2011) established the earlier security rationale for strong identity and access controls across cloud service models.

The literature demonstrates strong individual mechanisms - two-factor login, ECC-based mutual authentication, biometrics, anonymity and access-control models - but a practical cloud deployment needs these elements to cooperate without forcing the same authentication burden on every request. Fixed multi-factor schemes increase friction even for low-risk sessions, whereas password-only login is vulnerable to credential theft. A methodological gap therefore exists for a compact architecture that (i) binds authentication to a known device, (ii) evaluates contextual risk, (iii) escalates to a stronger factor only, when necessary, (iv) derives a fresh session key, and (v) links the final identity to least-

privilege authorization and rapid revocation. The proposed framework addresses this gap.

Methodology

1. Research Design

The research follows a design-and-evaluate methodology. First, security requirements are extracted from the reviewed literature. Second, a cloud authentication architecture is specified around an identity provider or registration authority (RA), user device, risk engine and cloud service gateway. Third, an authentication protocol is defined using fresh nonces, password verification, a device public/private key pair and optional step-up verification. Fourth, the protocol is evaluated against a predefined threat model. Finally, alternative mechanisms are compared using a normalized operation-cost model and an eight-property security checklist. This approach is reproducible because the security properties, decision rules and cost assumptions are stated explicitly.

2. Proposed Authentication Architecture

The proposed architecture contains five logical entities. The user device stores a protected device private key and initiates login. The identity provider/RA manages registration, password verifiers, device public keys and multi-factor settings. The risk engine examines contextual evidence such as new-device status, repeated failures, unusual network origin and abnormal request rate. The cloud gateway validates the issued token before forwarding the request, and the cloud service applies role or attribute-based permissions. This separation follows the scalability principle described by Moghaddam et al. (2014) and the IAM integration emphasized by Indu et al. (2018).

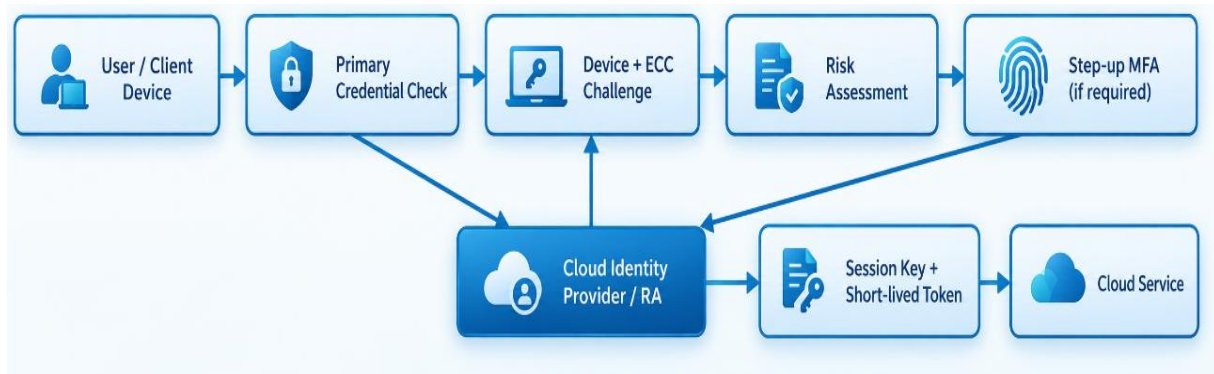


Figure 1. Proposed secure adaptive authentication architecture for a cloud environment.

3. Threat Model and Security Requirements

The model assumes that an attacker can observe, replay, delay or modify messages on an untrusted network and can attempt password guessing, credential stuffing, phishing-derived password reuse, user impersonation and session replay. A stolen password or a copied browser session is also considered. The RA and protected key store are assumed to be hardened trust anchors, but the protocol limits damage from a compromised session by using short token lifetimes and per-session keys. The required security properties are: replay resistance, resistance to offline password guessing, protection after password theft, mutual authentication, man-in-the-middle resistance, forward secrecy, identity privacy/pseudonymity and least-privilege authorization. These properties reflect the weaknesses repeatedly discussed in cloud-authentication studies (Namasudra & Roy, 2017; Wazid et al., 2016; Wang et al., 2018).

4. Enrollment Phase

Step 1 - User registration: the user selects an identifier and password. The server stores only a salted, iterated password verifier rather than the plaintext password. Step 2 - Device binding: the client generates an ECC key pair; the private key remains in protected device storage and the public key is registered with the RA. Step 3 - Multi-factor setup: an OTP secret or locally protected biometric factor is enrolled for step-up authentication. Step 4 - Authorization mapping: the account is assigned a role or attribute set, such as user, analyst or administrator. Step 5 - Recovery and revocation: the RA records recovery methods and supports immediate invalidation of a device key, OTP secret or active token after suspected compromise.

5. Login, Mutual Authentication and Session Establishment

Step 1 - Primary request: the client submits its identifier and password-derived proof through a protected channel. The RA verifies the stored password verifier and creates a fresh server nonce N_s . Step 2 - Device proof: the client signs or otherwise proves possession of the device private key over N_s and a fresh client nonce N_c . The RA verifies the response using the registered device public key. This makes a stolen password insufficient by itself.

Step 3 - Risk scoring: the risk engine calculates a contextual score $R = w1F + w2D + w3L + w4G$, where F represents recent failed attempts, D indicates device novelty, L represents unusual

location/network context and G represents request-rate or behavioral deviation. The weights are deployment parameters. If R remains below the step-up threshold, the protocol proceeds without an additional user action; if R exceeds the threshold, an OTP or local biometric confirmation is required.

Step 4 - Mutual verification and key derivation: the client and RA/cloud gateway complete an ephemeral ECC exchange. A session key is derived as $SK = KDF(ECDH, N_c, N_s, \text{session-context})$. Because a fresh ephemeral secret is used for each session, compromise of a later long-term credential does not reveal earlier session keys. This design choice follows the forward-secrecy and public-key deployment concerns discussed by Wang et al. (2018).

Step 5 - Token issuance and authorization: after all required factors are validated, the RA issues a signed, short-lived token containing a pseudonymous subject, authentication assurance level, role/attributes, issued time, expiry and a unique session identifier. The gateway validates the signature and expiry before each protected request. Authorization then enforces least privilege, consistent with the access-control findings summarized by Karatas and Akbulut (2018). Step 6 - Revocation and audit: logout, device loss, repeated anomalies or administrator action revoke the session identifier and optionally the device key. Security events are logged for investigation without recording raw passwords or biometrics.

6. Analytical Evaluation Procedure

The evaluation uses two complementary measures. First, each candidate mechanism receives one point for directly satisfying each of the eight required properties: replay resistance, offline-guessing resistance, stolen-password protection, mutual authentication/MITM resistance, session forward secrecy, identity privacy, device binding and least-privilege authorization. The maximum security-coverage score is therefore 8. Second, computational effort is represented by relative computation units (RCU). Password hashing/verification is the baseline 1.0 RCU; OTP verification contributes a small additional cost; ECC possession proof and ephemeral key agreement contribute a larger cost. RCU values are normalized design indicators, not milliseconds and not production benchmark measurements. The adaptive framework is evaluated on two paths: normal risk and elevated risk.

Results

Table 1. Analytical comparison of cloud authentication alternatives.

Mechanism	Security coverage (out of 8)	Relative cost (RCU)	Extra user action	Main limitation
Password only	2/8	1.0	None	Fails after credential theft; weak replay/context protection
Password + OTP	5/8	1.6	OTP each login	Improves stolen-password resistance but lacks device-bound forward secrecy
Fixed ECC two-factor	7/8	2.4	Device factor	Strong security but same cryptographic burden for every session
Proposed adaptive MFA	8/8	1.9 normal / 3.0 high risk	None normally; OTP/biometric when risky	Depends on sound risk thresholds and secure recovery

Table 1 shows that the proposed framework satisfies all eight design requirements because it combines credential checking, device proof, fresh nonces, ephemeral key establishment, privacy-aware identity representation and explicit authorization. Password-only authentication reaches only two properties in the checklist because a correct password does not prove that the current device or network session is trustworthy. Password plus OTP offers substantial improvement against stolen passwords and replayed credentials, but a simple OTP design does not by itself provide device-bound cryptographic identity, session forward secrecy or fine-grained authorization.

The fixed ECC two-factor design obtains strong coverage but executes the stronger cryptographic path for every login. In the normalized model its cost is 2.4 RCU, whereas the proposed architecture uses a 1.9 RCU normal path and activates the additional factor only when contextual risk is elevated. Thus, the normal path is approximately 20.8% lower in relative cost than the always-on fixed ECC alternative, while the 3.0 RCU high-risk path deliberately spends more effort to obtain stronger assurance. The result is a security-usability trade-off: low-risk users experience fewer prompts, but high-risk activity receives stronger verification.

Security reasoning also indicates that the framework is resistant to replay because both client and server contribute fresh nonces to the authenticated exchange; captured responses therefore cannot be reused in a later session. A stolen password is insufficient because the attacker must also demonstrate possession of the registered device key and, under elevated risk, the OTP or biometric factor. Ephemeral ECC key establishment limits the impact of later key

compromise on earlier sessions. Short-lived tokens and centralized revocation reduce the exposure period of copied sessions. These controls are consistent with the security properties emphasized in earlier two-factor, biometric and ECC-based cloud schemes (Namasudra & Roy, 2017; Wazid et al., 2016; Wang et al., 2018).

Conclusion and Discussion

This paper presented a methodology for secure authentication in cloud-based computing environments using a layered, adaptive architecture. The method was constructed from cloud security and authentication research published no later than 2018. Earlier studies established that cloud security depends on coordinated identity management, authentication, access control and session protection rather than a single password check (Takabi et al., 2010; Subashini & Kavitha, 2011; Indu et al., 2018). Authentication-specific research further showed the benefits of two-factor security, biometrics, anonymity and ECC-based key agreement (Namasudra & Roy, 2017; Wazid et al., 2016; Wang et al., 2018). The proposed framework integrates these principles in one practical flow.

The main contribution is the use of adaptive assurance. A known device with valid credentials and normal context follows a low-friction path, while an unfamiliar or suspicious request triggers an additional OTP or biometric factor. Mutual cryptographic verification establishes a fresh session key, and successful authentication is immediately connected to a short-lived authorization token and least-privilege access policy. The analytical results indicate complete coverage of the selected security requirements

while reducing normal-path overhead compared with an always-on stronger cryptographic path.

The study also has limitations. The RCU values are normalized analytical costs, not measurements from a commercial cloud platform, and the risk engine requires careful threshold tuning to avoid false challenges or missed attacks. Device fingerprinting and context collection must also be implemented with privacy controls. Future experimental work should deploy the framework on a real cloud testbed, measure authentication latency under concurrent load, evaluate recovery workflows after device loss and test the protocol using formal verification tools. Even with these limitations, the methodology provides a clear and reproducible basis for designing a secure cloud authentication layer that balances security, scalability and usability.

References

- Indu, I., Anand, P. M. R., & Bhaskar, V. (2018). Identity and access management in cloud environment: Mechanisms and challenges. *Engineering Science and Technology, an International Journal*, 21(4), 574-588. <https://doi.org/10.1016/j.jestch.2018.05.010>
- Wang, C., Ding, K., Li, B., Zhao, Y., Xu, G., Guo, Y., & Wang, P. (2018). An enhanced user authentication protocol based on elliptic curve cryptosystem in cloud computing environment. *Wireless Communications and Mobile Computing*, 2018, Article 3048697. <https://doi.org/10.1155/2018/3048697>
- Karatas, G., & Akbulut, A. (2018). Survey on access control mechanisms in cloud computing. *Journal of Cyber Security and Mobility*, 7(3), 1-36. <https://doi.org/10.13052/2245-1439.731>
- Namasudra, S., & Roy, P. (2017). A new secure authentication scheme for cloud computing environment. *Concurrency and Computation: Practice and Experience*, 29(20), e3864. <https://doi.org/10.1002/cpe.3864>
- Wazid, M., Das, A. K., Kumari, S., Li, X., & Wu, F. (2016). Provably secure biometric-based user authentication and key agreement scheme in cloud computing. *Security and Communication Networks*, 9(17), 4103-4119. <https://doi.org/10.1002/sec.1591>
- Amin, R., Islam, S. H., Biswas, G. P., Giri, D., Khan, M. K., & Kumar, N. (2016). A more secure and privacy-aware anonymous user authentication scheme for distributed mobile cloud computing environments. *Security and Communication Networks*, 9, 4650-4666. <https://doi.org/10.1002/sec.1655>
- Moghaddam, F. F., Moghaddam, S. G., Rouzbeh, S., Araghi, S. K., Alibeigi, N. M., & Varnosfaderani, S. D. (2014). A scalable and efficient user authentication scheme for cloud computing environments. *2014 IEEE Region 10 Symposium*, 508-513. <https://doi.org/10.1109/TENCONSpring.2014.6863086>
- Khalil, I. M., Khreishah, A., & Azeem, M. (2014). Cloud computing security: A survey. *Computers*, 3(1), 1-35. <https://doi.org/10.3390/computers3010001>
- Zissis, D., & Lekkas, D. (2012). Addressing cloud computing security issues. *Future Generation Computer Systems*, 28(3), 583-592. <https://doi.org/10.1016/j.future.2010.12.006>
- Subashini, S., & Kavitha, V. (2011). A survey on security issues in service delivery models of cloud computing. *Journal of Network and Computer Applications*, 34(1), 1-11. <https://doi.org/10.1016/j.jnca.2010.07.006>
- Takabi, H., Joshi, J. B. D., & Ahn, G.-J. (2010). Security and privacy challenges in cloud computing environments. *IEEE Security & Privacy*, 8(6), 24-31. <https://doi.org/10.1109/MSP.2010.186>