CS-062 · Application Security

Authentication

Verifying that a user or system is who they claim to be.

Executive Summary

Authentication is the process of verifying that a user or system is who they claim to be before granting access. It is distinct from authorization, which decides what a verified identity is allowed to do. Because stolen or weak credentials are behind a large share of breaches, strong authentication, especially multi-factor authentication, is one of the highest-value controls an organization can deploy.

What It Is

Authentication answers the question, are you who you say you are. It relies on one or more factors: something you know such as a password, something you have such as a phone or hardware key, and something you are such as a fingerprint or face. Single-factor authentication uses one of these, while multi-factor authentication requires two or more from different categories, so a stolen password alone is not enough. Authentication also covers how a session is established and maintained after login, since a verified identity must be carried securely across subsequent requests without exposing the credentials again.

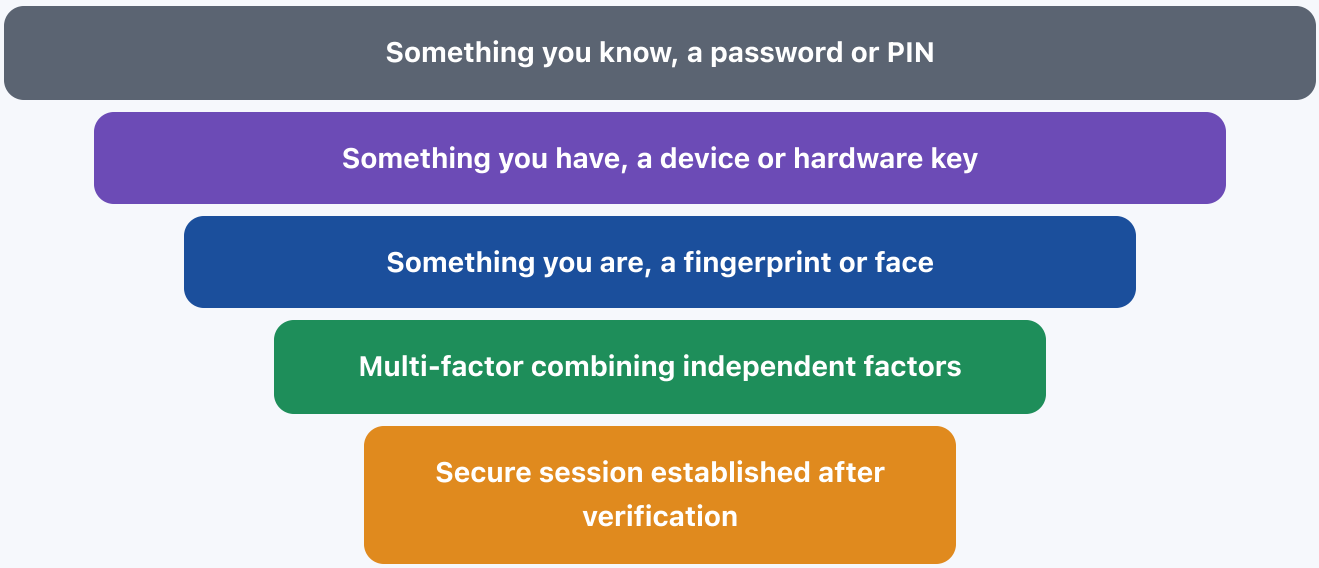
Why It Matters

Credentials are among the most attacked assets in security because they open the door to everything behind them. Phishing, password reuse, and automated guessing all target authentication, and a single compromised account can lead to a full breach. Strong authentication dramatically reduces this risk, which is why multi-factor authentication is one of the most consistently recommended controls. For professionals, understanding authentication is essential across development, operations, and governance, since weaknesses here undermine every other control that depends on knowing who is acting.

How It Works

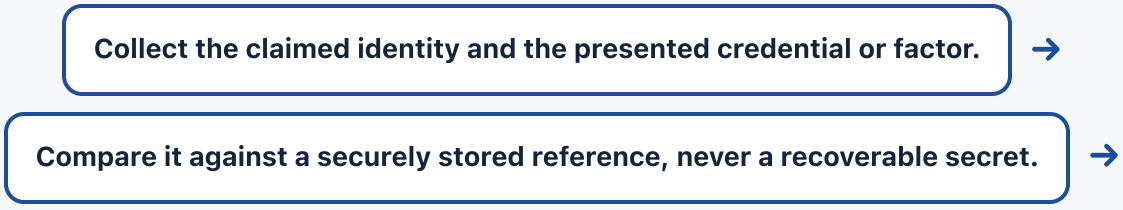
At a conceptual level, authentication compares the credentials presented against a securely stored reference and confirms they match without ever exposing the stored secret. Passwords should never be stored in a recoverable form; they are stored using a strong, slow password hashing method with per-user salting so that a stolen store is hard to crack. Multi-factor authentication adds a second, independent factor so that a stolen password is not sufficient. After a successful login, the system issues a session identifier or token that must be protected, scoped, and expired appropriately. Additional measures such as rate limiting and lockouts slow guessing, and monitoring for unusual logins helps catch takeover attempts. Passwordless approaches based on strong cryptographic keys are increasingly used to remove the weakest link, the reusable password.

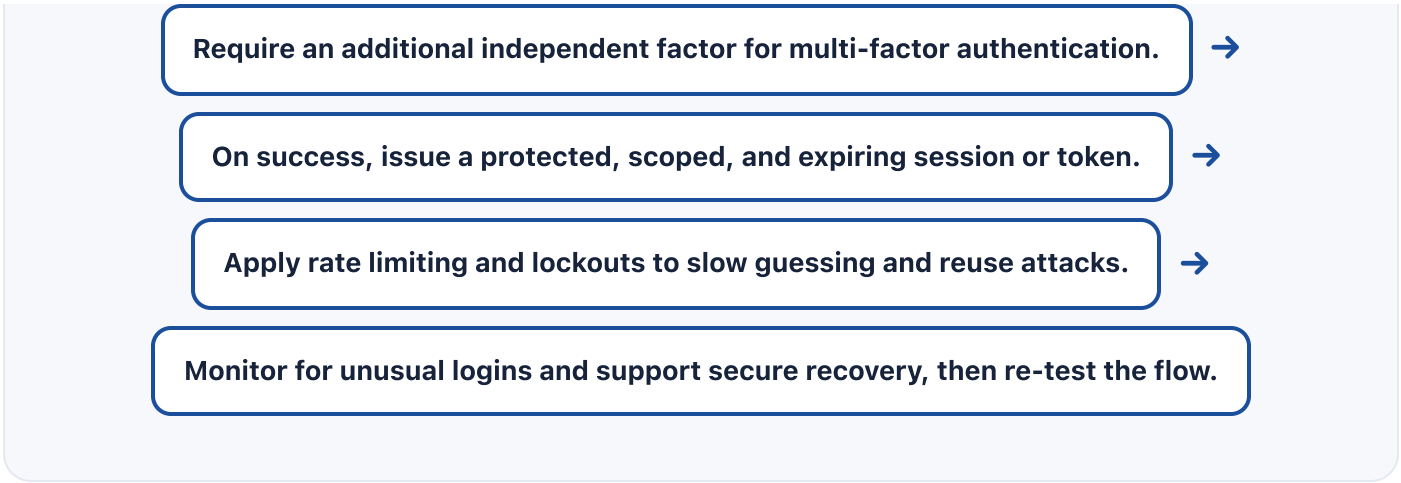
Architecture Diagram



Strong authentication combines independent factors so a single stolen credential is not enough to gain access.

Visual Workflow





Common Attacks

- Phishing that captures passwords and sometimes one-time codes
- Credential stuffing that reuses passwords leaked from other breaches
- Brute-force and password-spraying against weak or common passwords
- Session hijacking or fixation that steals or plants a session identifier
- Bypassing weak multi-factor methods or abusing insecure account recovery

Common Mistakes

- Storing passwords without a strong, slow, salted hashing method
- Relying on passwords alone without multi-factor authentication
- Allowing unlimited login attempts with no rate limiting or lockout
- Building account recovery weaker than the login it protects
- Leaking whether a username exists through different error messages

Best Practices

- Require multi-factor authentication, preferring phishing-resistant methods
- Store passwords with a strong, slow, salted password hashing function
- Support long passphrases and screen against known breached passwords
- Apply rate limiting, lockouts, and monitoring for suspicious logins
- Issue secure, scoped, expiring sessions and protect session identifiers
- Make account recovery as strong as the primary login path

Quick Checklist

- [x] Multi-factor authentication available and enforced for sensitive access
- [x] Passwords stored with a strong, slow, salted hashing method
- [x] Screening against known breached passwords in place
- [x] Rate limiting and account lockout configured
- [x] Sessions secure, scoped, and expiring, with protected identifiers
- [x] Account recovery reviewed to match login strength

Recommended Tools

Authenticator app or hardware security key
Provides a strong second factor, with keys being phishing-resistant

Password manager
Generates and stores unique strong passwords to prevent reuse

Identity provider with single sign-on
Centralizes authentication and enforces consistent policy

Breached-password screening service
Blocks passwords known to appear in prior leaks

Industry Standards

NIST SP 800-63
Digital identity guidelines covering authentication and credential strength

OWASP Application Security Verification Standard (ASVS)
Testable requirements for authentication and session management

OWASP Top 10
Identification and authentication failures are a recognized risk category

Career Relevance

Authentication is central to security engineers, identity and access management specialists, application security engineers, and penetration testers, who both build and test login systems. Developers must implement authentication and session handling correctly, and GRC and audit professionals assess whether authentication meets policy and standards. For the AI-Governance-Jobs.com audience, distinguishing authentication from authorization and explaining multi-factor is a common interview topic.

Interview Questions

- What is the difference between authentication and authorization?
- What are the three authentication factor categories, with an example of each?
- Why should passwords be stored with a slow, salted hashing function?
- Why is multi-factor authentication so effective against common attacks?
- What is credential stuffing, and how do you defend against it?

Related Certifications

CompTIA Security+ **ISC2 Certified in Cybersecurity (CC)** **ISC2 CISSP (for leadership tracks)**

Further Reading

- NIST SP 800-63 Digital Identity Guidelines
- OWASP Authentication Cheat Sheet
- CISA: Multi-Factor Authentication

Key Takeaways

- Authentication verifies identity; authorization decides what that identity can do.
- Factors are something you know, have, or are, combined for multi-factor.
- Multi-factor authentication is one of the highest-value security controls.
- Passwords must be stored with a strong, slow, salted hashing method.

- Secure session handling and strong recovery are part of authentication.

FAQ

Related Careers

RELATED ROLES

- Application Security Engineer
- Penetration Tester
- Secure Code Reviewer
- Security Engineer

RELATED CERTIFICATIONS

- CompTIA Security+
- ISC2 Certified in Cybersecurity (CC)
- ISC2 CISSP (for leadership tracks)

CURRENT OPENINGS

Live roles are on the job board.

Browse all jobs

GUIDES & SALARY

- Career guides
- Role roadmaps
- Salary data
- Career tools

SUGGESTED LEARNING PATH

1. Ground the basics with CS-001 Cybersecurity
2. Study this sheet: **Authentication**
3. Go deeper: Authorization
4. Go deeper: Secrets Management

5. Validate it: work toward **CompTIA Security+**
6. Find the role: browse current openings

RELATED SHEETS

CS-063 Authorization | CS-064 Secrets Management | CS-060 Cross-Site Request Forgery (CSRF) | CS-057 OWASP Top 10 | CS-001 Cybersecurity