Symmetric+
Asymmetric

A — B

A use B’s public key to encrypt the symmetric key

Send to B

B uses his/her private key to decrypt the cipertext

A can send the messages to B by using symmetric encryption

CSC116 Authentication

Who Are You?

Authentication

Authentication is the process that companies use to confirm that only the right people, services, and apps with the right permissions can get organizational resources. It's an important part of cybersecurity because a bad actor's number one priority is to gain unauthorized access to systems.

Why is Authentication Important?

Scenario: A University Student Database Without Authentication

Imagine a university has an **online student database** that stores:

- **Student names**
- **Grades**
- **Course enrollments**
- **Personal information (email, phone number, etc.)**

If this database **has no authentication**, anyone can access and modify the data.

Biometrics

Ear Shape Recognition

Voice Recognition

How a computer make authentication？

Knowledge-Based Authentication

Passwords/PINs: The most common method, where users enter a secret string.

Security Questions: Personal questions (e.g., "What is your first pet's name?").

One-Time Passcodes (OTP): Codes sent via SMS, email, or an authenticator app. Multi-Factor Authentication (MFA)

Smart Cards: Cards with embedded chips for authentication.

Hash function (SHA-256/SHA-3 Hashing**)**

A **hash function** is a **mathematical algorithm** that converts any input (such as patient records, medical notes, or passwords) into a fixed-length string of characters, called a **hash value**.

Demo

https://guggero.github.io/blockchain-demo/#!/hash

Hash collision

https://colab.research.google.com/drive/1R6MktlqvXBRpoQBWRi1mJjeIhpKzrnjJ?usp=sharing

If you remember the moment when, after many years of hard work and a long voyage

e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855

You stand in the center of your room

e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855

For CSer: How to solve the Collision?: https://www.youtube.com/watch?v=td0h7cv4cc0

Hash Functions in Modern Cryptography

- The **MD (Message Digest) family** (MD2, MD4, MD5) was developed by **Ronald Rivest** at MIT in the late 1980s and early 1990s.
- The **SHA (Secure Hash Algorithm) family** was developed by the **National Security Agency (NSA)** in the 1990s.
- The **SHA-3 standard (Keccak)** was developed by **Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche** and was adopted by NIST in 2015.
- SHA-256 (Secure Hash Algorithm 256-bit) was designed by the **National Security Agency (NSA)** and published by the **National Institute of Standards and Technology (NIST)** as part of the **SHA-2 family** in **2001**.

Algorithm	Hash Length	Security Level
MD5	32 hex chars	❌ Weak (Fast, easily cracked)
SHA-1	40 hex chars	❌ Weak (Considered broken)
SHA-256	64 hex chars	✅ Stronger but still vulnerable to brute-force
bcrypt	Variable	✅ Strong (Slow by design)
PBKDF2	Variable	✅ Strong (Key stretching)
Argon2	Variable	✅ Strong (Memory-hard function, best for modern security)

Features of Hash function

1 The same input will always produce the same hash value.

2 No matter how long or short the input is, the output hash is always of fixed length.

3 It is **impossible** to reverse-engineer the original input from the hash value.

4 Different inputs should produce different hash values.

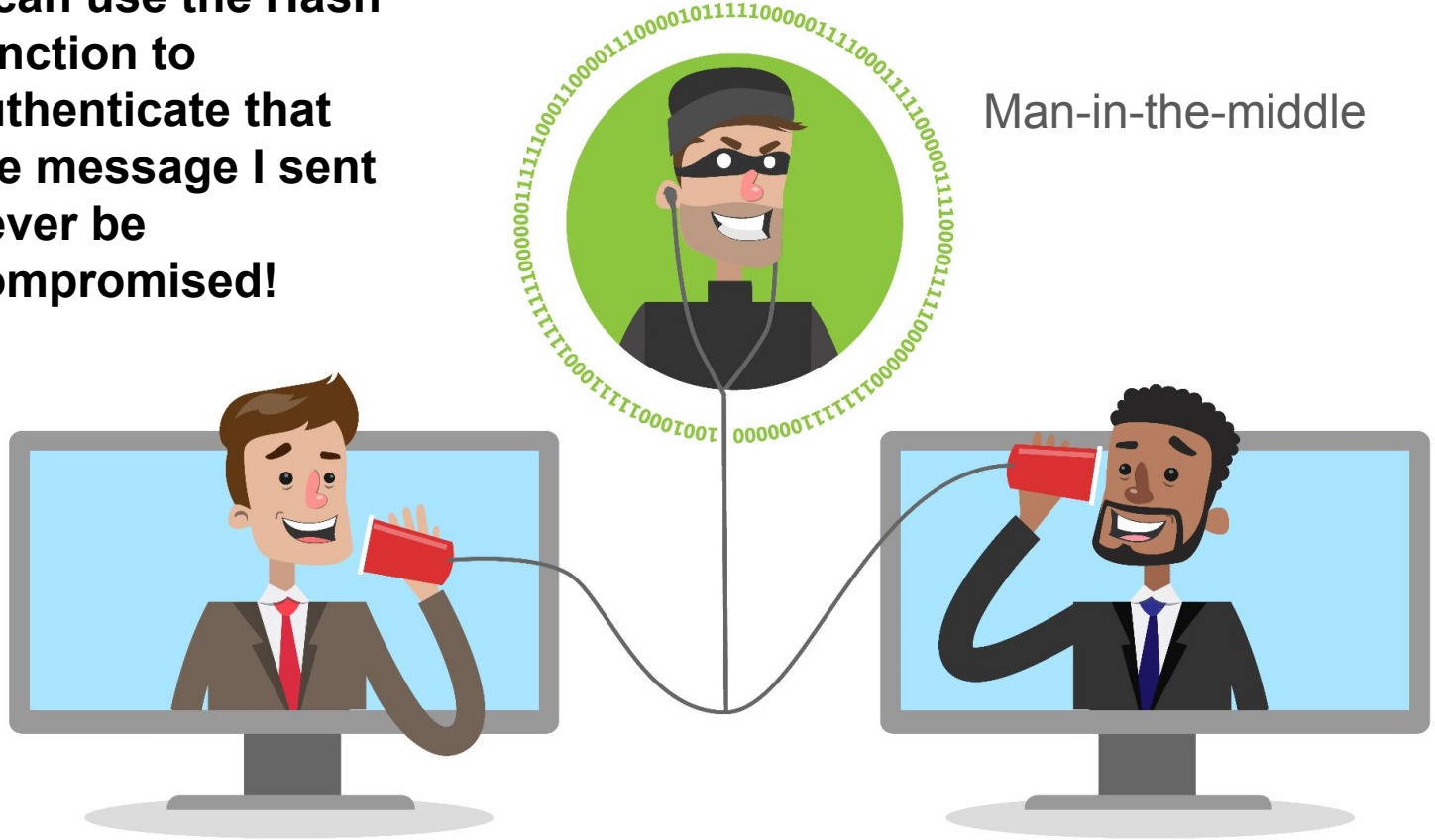
5 Hash functions are designed to compute hashes quickly, even for large amounts of data.

6 Hash functions allow for quick verification of data integrity.

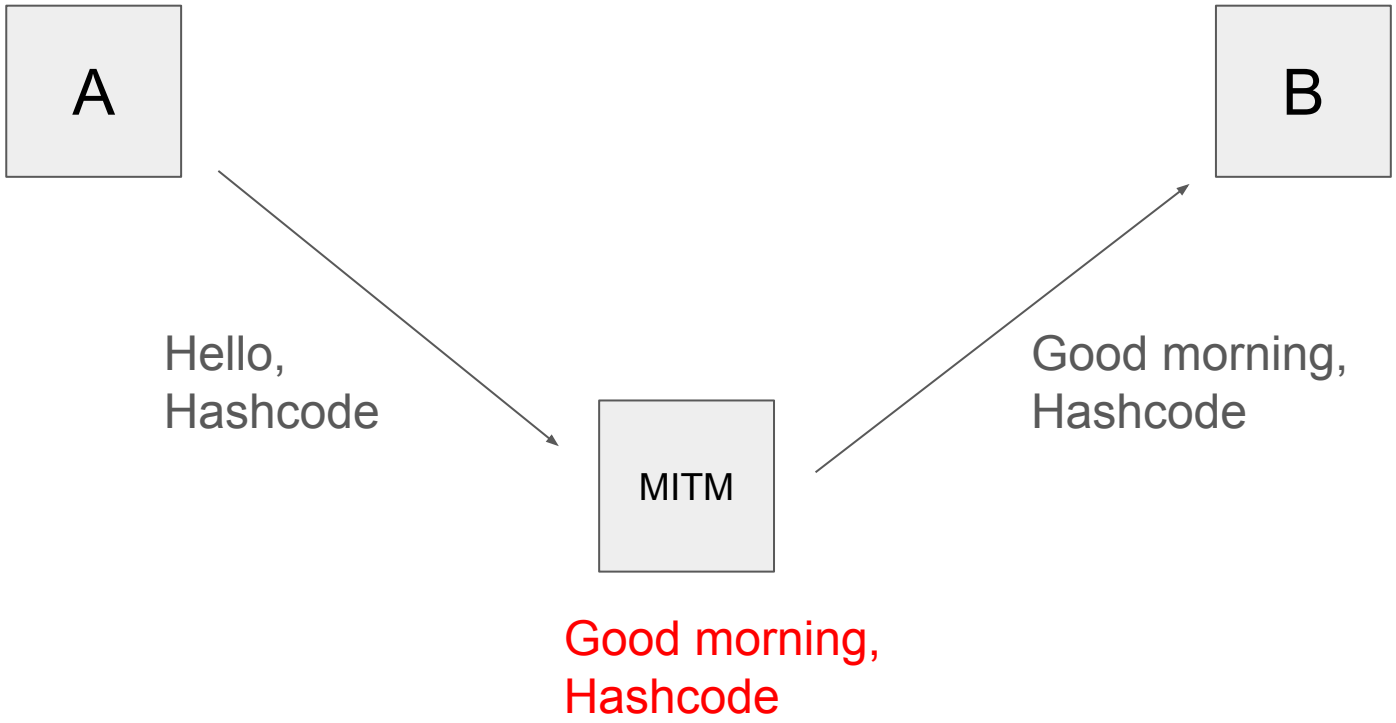
7 A small change in the input results in a huge change in the hash.

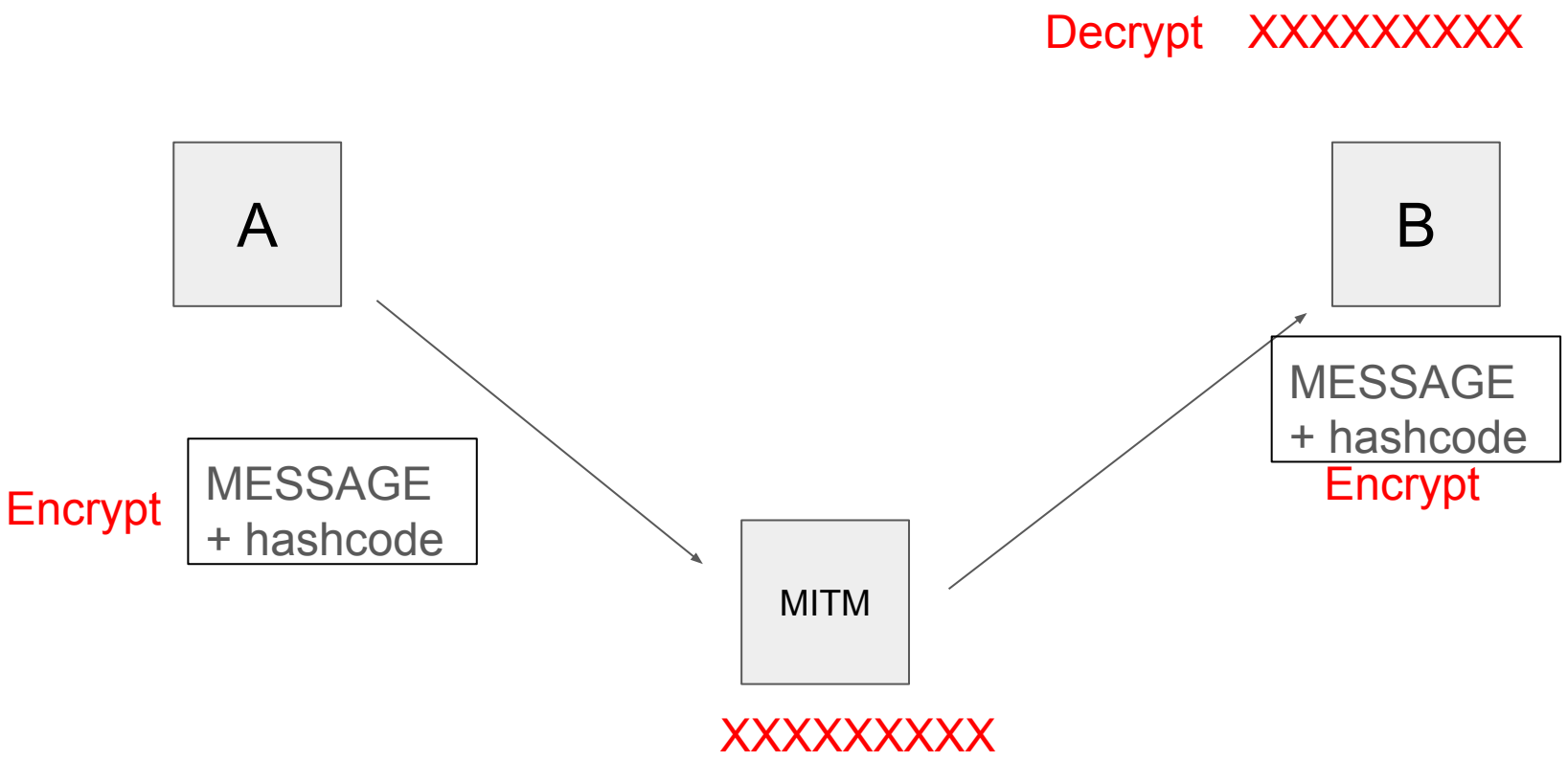
Hash Code can be used for Authenticate the messages

I can use the Hash function to authenticate that the message I sent never be compromised!



Man-in-the-middle


A
B
Hello,
Hashcode
Good morning,
Hashcode
MITM
Good morning,
Hashcode


Decrypt XXXXXXXXX
A
B
MESSAGE
+ hashcode
Encrypt
MESSAGE
+ hashcode
MITM
XXXXXXXXX

Does a university's online student database require authentication? (Yes/No)
✅ **Yes** (Otherwise, anyone could view or modify student information.)

Do you need authentication to post on social media platforms like Facebook or Twitter? (Yes/No)
✅ **Yes** (Users must log in before posting content.)

Do you need authentication to access a government website to check the weather forecast? (Yes/No)
❌ **No** (Weather forecasts are public information accessible to anyone.)

Do you need authentication to withdraw money from an ATM? (Yes/No)
✅ **Yes** (You must enter a PIN or use biometric authentication to verify your identity.)

Does opening a public library's online catalog require authentication? (Yes/No)
❌ **No** (Anyone can browse books, but borrowing may require authentication.)

Do you need authentication to unlock your smartphone? (Yes/No)
✅ **Yes** (Most smartphones require a password, PIN, fingerprint, or face recognition.)

Do online banking transactions require authentication? (Yes/No)
✅ **Yes** (Banks require passwords, OTPs, or biometrics for security.)

Do you need authentication to visit a company’s official homepage? (Yes/No)
❌ **No** (Company websites are usually public, but internal portals require authentication.)

Does accessing a private medical record require authentication? (Yes/No)
✅ **Yes** (Medical records contain sensitive information and must be protected.)

Single Sign-On (SSO)

Single sign-on (SSO) is an authentication scheme that allows a user to log in with a single SSO ID to any of several related, yet independent, software systems.

Single Sign-On (SSO)

Demo

Advantages

Reduces Password Management Burden – Users only need to remember one set of credentials, minimizing the risk of forgotten passwords.

Improves User Experience – Eliminates the need for repeated logins, allowing seamless navigation across multiple applications.

Enhances Security – Reduces the risk of weak or reused passwords and supports strong authentication methods like Multi-Factor Authentication (MFA).

Centralized Identity Management – IT administrators can manage access permissions more effectively, improving compliance and security.

Lowers IT Maintenance Costs – Reduces the need for password resets and user account management, saving IT support resources.

Authentication, **Authorization**, Encryption

Authentication is used by a server when the server needs to know exactly who is accessing their information or site.

Authorization is a process by which a server determines if the client has permission to use a resource or access a file.

Encryption involves the process of transforming data so that it is unreadable by anyone who does not have a decryption key.

Hash Password

Store raw password is dangerous!

Hash Password

Storing passwords **in plaintext** is highly insecure. If a database is leaked or compromised, all user passwords are immediately exposed.

To improve security, we store **hashed passwords** instead of plaintext passwords. The idea is:

1. When a user creates a password, it is hashed and stored in the database.
2. When the user logs in, the entered password is hashed again and compared to the stored hash.
3. If the two hashes match, authentication is successful.

Rainbow Table Attack 🌈

- Attackers use **precomputed hash databases** to reverse hashes back into original passwords.
- If a database contains `MD5("123456")`, an attacker can quickly find that hash and determine the password.

Demo

https://gugger o.github.io/bl ockchain-de mo/#!/hash

```
password123
qwerty
letmein
```

Have a strong password please!

Question!

A company implements a security system where employees must use a username and password to log in. However, an attacker manages to steal a user's password through a phishing attack.

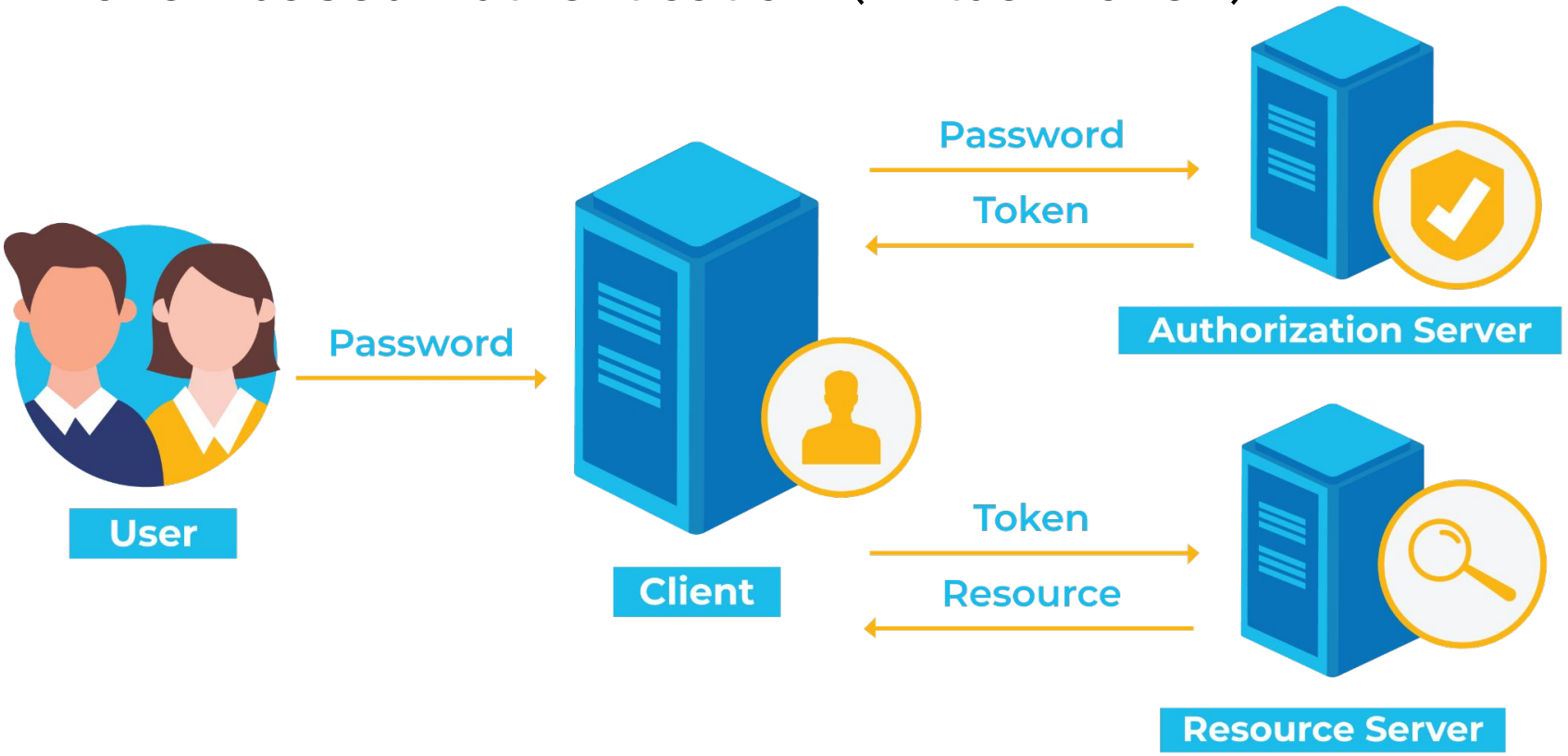
💡 **Question:**
Even though the company uses authentication, why is this system still vulnerable? What additional security measures could be implemented to prevent such attacks?

Strong Authentication

Strong Authentication refers to a robust method of verifying a user's identity to ensure secure access to systems, applications, or data. It typically involves multiple layers of security and relies on the principle of **multi-factor authentication (MFA)**. This approach requires a combination of two or more of the following factors:

1. **Something You Know**: A knowledge-based factor, such as a password, PIN, or security question.
2. **Something You Have**: A possession-based factor, like a security token, smart card, or a mobile device with a one-time password (OTP) generator.
3. **Something You Are**: A biometric-based factor, including fingerprints, facial recognition, retina scans, or voice recognition.

Token-based Authentication（Virtual Token）



okta

Token-based Authentication Pros

With tokens, authentication is separate from authorization. A user logs in once, receives a token, and then uses that token to access different services without needing to authenticate repeatedly.

Reduced Attacks: Since tokens expire after a certain period, even if they are stolen, they are only useful for a limited time.

No Need to Store Passwords: The server only verifies tokens without storing user credentials in memory.

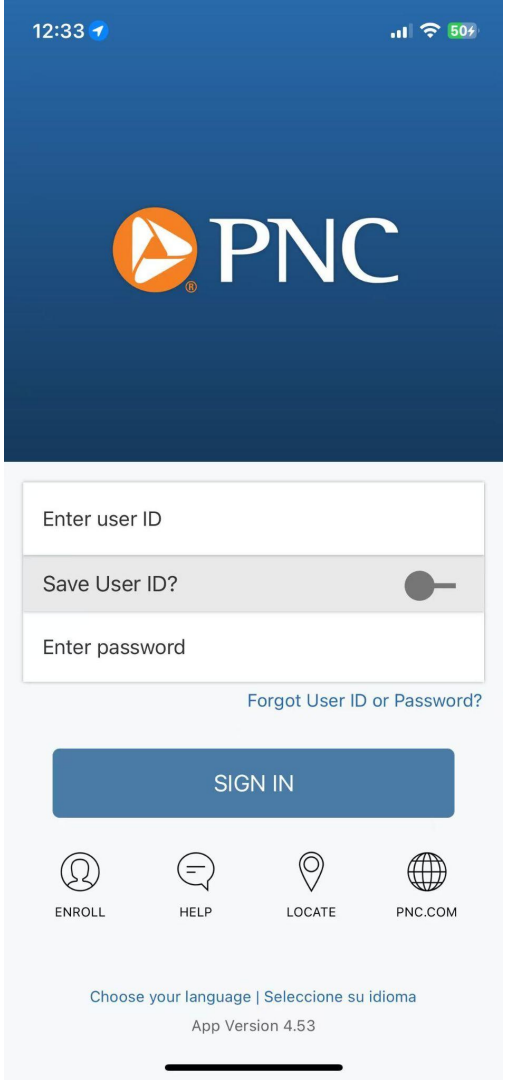
Smart card is a token （Physical Token）

A smart card can be considered a type of token, specifically a security token, as it contains an embedded chip that stores sensitive information and can be used for authentication purposes like logging into systems or making secure transactions, effectively acting as a digital key or "token" to access services.

Case Study


PNC BANK
VirtualWallet
by PNC BANK
DEBIT
VISA
DEBIT
VISA


ATM

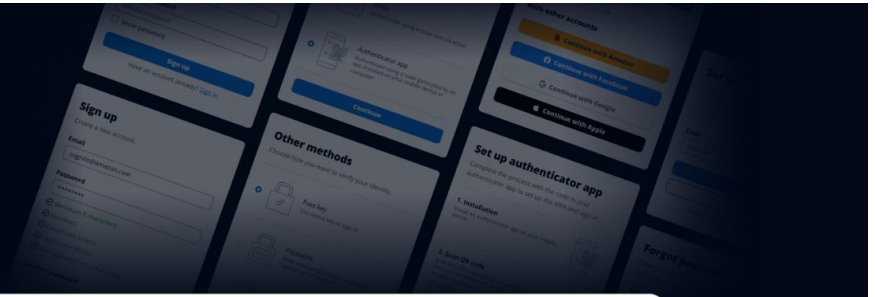

12:33
PNC
Enter user ID
Save User ID?
Enter password
Forgot User ID or Password?
SIGN IN
ENROLL
HELP
LOCATE
PNC.COM
Choose your language | Seleccione su idioma
App Version 4.53

Security, Identity & Compliance

Amazon Cognito

Secure identity and access management for apps

With Amazon Cognito, you can add registration and sign-in to your applications. You can also get temporary AWS credentials for access to your cloud resources.



Add sign-in and sign-up experiences to your app

With Amazon Cognito, you can add secure authentication and authorization to your application. Amazon Cognito scales to support millions of users. It's free for up to 10,000 monthly active users (MAUs) in your account. Learn more

Get started for free in less than five minutes

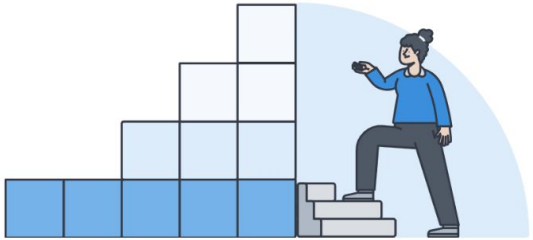
Grant app access to AWS services

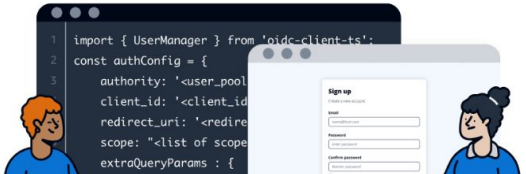
Get temporary credentials for AWS services with Amazon Cognito. You can grant resource access to signed-in users and anonymous guest users. This feature comes at no additional cost.

Get started for free

Scale to **millions of users** with a **99.9% availability** service level agreement

Scale to millions of users with a fully-managed, high-performance, and reliable user directory. Amazon Cognito user pools have a service level agreement (SLA) monthly uptime of 99.9% for each AWS region that Amazon Cognito operates in.





Easy setup with SDKs and managed login pages

Amazon Cognito has efficient tools for applications. Step through setup guidance for technologies like JavaScript, IOS, and Android. Apply custom branding to your managed login pages.

Amazon Cognito

User Pools = Authentication → Verify who the user is (Sign-in).

Identity Pools = Authorization → Grant AWS resource access (IAM roles, AWS credentials).

AI for Authentications

Facial Recognition Authentication

Behavioral Biometrics Authentication

Anomaly Detection in Authentication

Deep Learning for Phishing Detection

AI-Generated One-Time Passwords (OTP)

Fingerprint Authentication

Cashier-Free Checkout System

https://www.youtube.com/watch?v=NrmMk1Myrxc

Cashier-free checkout systems (e.g., Amazon Go, AiFi, Grabango) rely on a combination of **AI, computer vision, sensors, and RFID** (Radio Frequency Identification) tag to track what a user picks up and purchases.