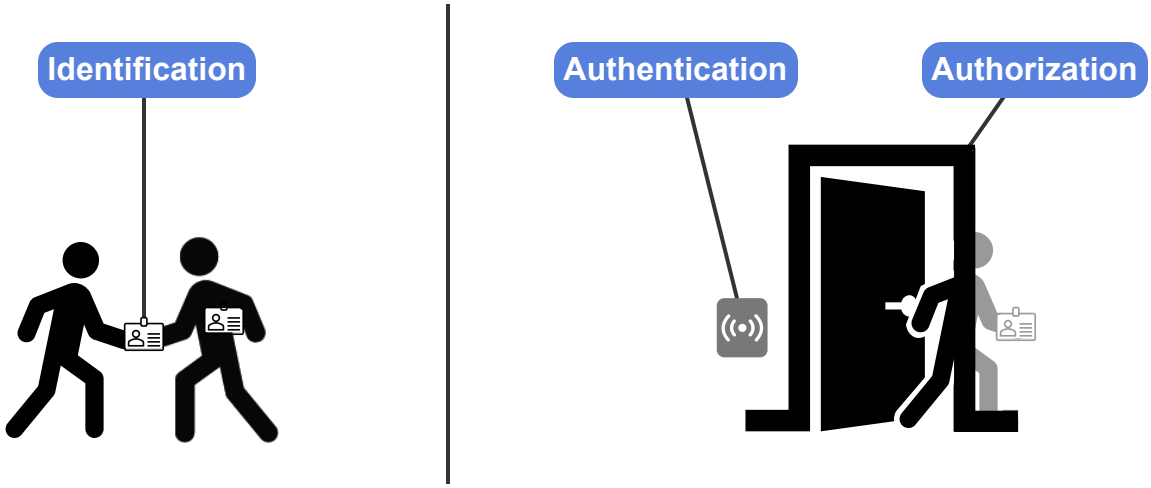
2.1 Principles

Identity and access management (IAM)

Identity and Access Management (IAM), also known as ***Identity Management (IdM)***, is a framework of technologies and policies for managing user identities in a system and controlling user access to the system's resources. The three main objectives of IAM is the identification, authentication, and authorization of users.

PARTICIPATION ACTIVITY 2.1.1: Identity and access management.



Animation content:

Static image: An employee is handed an ID card. The employee scans the ID card on a scanner next to a door and the door is unlocked. The employee opens the door and enters.

Animation captions:

1. An employee is issued an ID card as identification.
2. The card scanner authenticates the scanned ID card in the company system.
3. After authentication, the employee is granted authorization to enter and the door is unlocked.

Identification

A ***user*** is typically a person but could be an application, a process, or a device. ***Identification*** is the act of a user claiming an identity. A user can claim an identity by various means, including:

- ***Username***: a unique sequence of characters assigned to a user.
- ***Certificate***: a digital credential that binds the identity of a user to a cryptographic key.
- ***Token***: a physical device assigned to a user that can generate a unique code.
- ***SSH key***: a cryptographic key-pair owned by a user.
- ***Smart card***: a card with an embedded microchip that is assigned to a user.

Identity proofing is the process of verifying a user's identity during account creation. Ex: Registering for an online banking account requires identity verification by a government-issued ID.

An identity is associated with attributes within a system. An ***attribute*** is a specific characteristic of an identity. An attribute can be an identity's name, email address, location, or password.

PARTICIPATION ACTIVITY 2.1.2: Identification.

How to use this tool

Certificate **IAM** **SSH key** **Identification**

	The act of a user claiming an identity.
	A framework of technologies and policies for managing user identities in a system and controlling user access to the system's resources.
	A cryptographic key-pair owned by a user.
	A digital credential that binds the identity of a user to a cryptographic key.

Reset

PARTICIPATION ACTIVITY 2.1.3: Identification.

1) A user claiming an identity is always a person.
   - False
   - True

2) A certificate is a physical device that can generate a unique code.
   - False
   - True

3) An attribute is a specific characteristic of an identity.
   - False
   - True

4) A token is a cryptographic key-pair owned by a user.
   - False
   - True

Authentication

Authentication is the act of verifying or proving a user's claim to an identity. A user may provide different types of evidence to prove the user's claim to an identity such as the user's possession of a device, the user's location, or the user's knowledge of a password or a PIN. The different types of evidence that a user can provide to prove the user's claim to an identity are known as ***authentication factors***.

A user and a system exchange authentication information by using an authentication protocol. An ***authentication protocol*** is a type of computer communications protocol designed for securely transferring authentication information between two parties. Numerous authentication protocols exist, including password authentication protocol (PAP), challenge handshake authentication protocol (CHAP), Kerberos, EAP, IEEE 802.1X, RADIUS, and TACACS+.

PARTICIPATION ACTIVITY 2.1.4: Authentication.

How to use this tool

Authentication factors | Authentication protocol | Authentication

	The act of verifying or proving a user's claim to an identity.
	The different types of evidence that a user can provide to prove the user's claim to an identity.
	A type of computer communications protocol designed for securely transferring authentication information between two parties.

Reset

PARTICIPATION ACTIVITY 2.1.5: Authentication.

1) Authentication is the act of claiming an identity.
   - False
   - True

2) A user may provide a PIN to prove the user's claim to an identity.
   - False
   - True

3) Authentication ensures that an authorized user is prevented from accessing a protected resource.
   - False
   - True

©zyBooks 12/12/24 18:00 2172291
Daren Diaz
OUCYBS3213FreezeFall2024

4) An authentication protocol is used for securely transferring authentication information between two parties.
   - False
   - True

Authorization

Authorization is the act of granting a user access to a system resource. Authorization occurs after a user is identified and authenticated by a system.

Controlling access to a system resource involves a subject and an object. A ***subject*** is an entity that wants to access a system resource. A subject is typically a person but could be an application, a process, or a device. An ***object*** is a system resource that a subject wants to access. An ***access control model***, or ***authorization model***, is a set of technology-independent rules for controlling access to an object by a subject. The commonly used access control models are discretionary access control (DAC), mandatory access control (MAC), attribute-based access control (ABAC), role-based access control, and rule-based access control.

PARTICIPATION ACTIVITY 2.1.6: Authorization.

How to use this tool

Subject **Authorization** **Object** **Access control model**

	The act of granting a user access permissions to a system resource.
	An entity that wants to access a system resource.
	A system resource that a subject wants to access.
	A set of technology-independent rules for controlling access of a subject to an object.

Reset

PARTICIPATION ACTIVITY 2.1.7: Authorization.

1) Authorization is the act of verifying or proving a user's claim to an identity.

   - ○ False

- True

2) A subject is a system resource.

- False
- True

3) An access control model is a set of technology-independent rules for controlling access to a subject by an object.

- False
- True

4) Authorization occurs after a user is identified and before the user is successfully authenticated.

- False
- True

2.2 Authentication: Factors

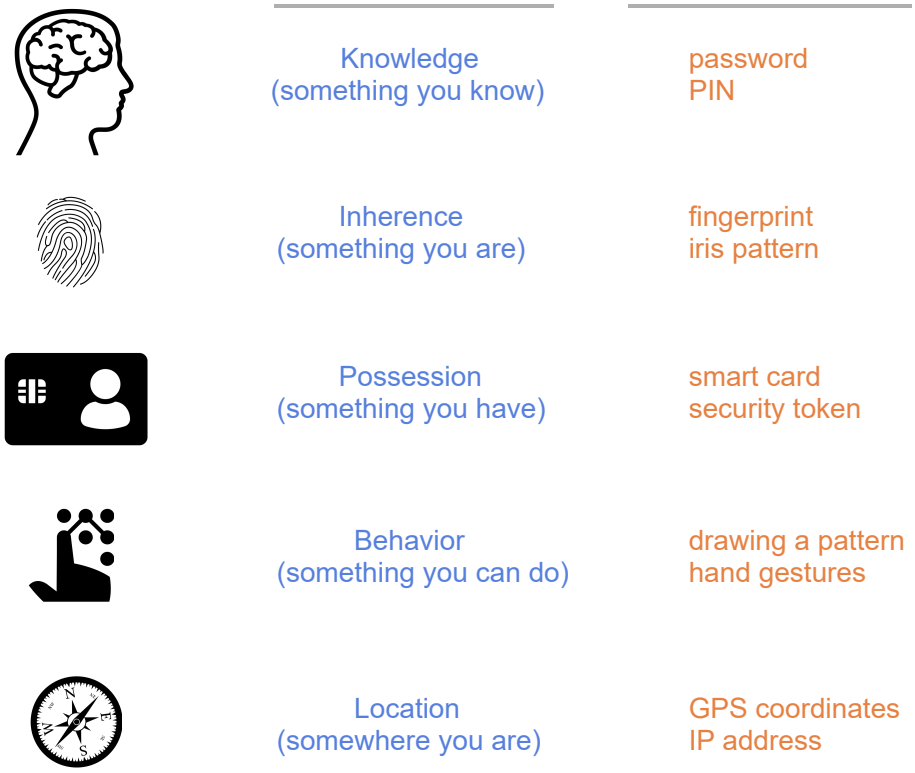
Authentication factors

An ***authentication factor*** is a type of evidence that a user provides to prove the user's claim to an identity. Five common authentication factors exist:

- ***Knowledge factor*** is a piece of information that a user knows. The most common types of knowledge factor is a password and a personal identification number (PIN). A knowledge factor is ***something you know***.
- ***Possession factor*** is something a user possesses or owns. The most common types of possession factors are a smart card and a mobile phone. A possession factor is ***something you have***.
- ***Inherence factor*** is a unique physical characteristic of a user. An inherence factor includes fingerprints, voiceprints, retina, and iris patterns. An inherence factor is ***something you are***.
- ***Location factor*** is information on a user's current location. A user's location can be determined by different means, including the GPS coordinates or the IP address of a user's device. A location factor is ***somewhere you are***.

- ***Behavior factor*** is an action a user performs. The most common types of behavior factor is a hand gesture or drawing of a specific pattern onto a device's screen. A behavior factor is ***something you can do***.

PARTICIPATION ACTIVITY | 2.2.1: Authentication factors.



Authentication factor	Examples
Knowledge (something you know)	password PIN
Inherence (something you are)	fingerprint iris pattern
Possession (something you have)	smart card security token
Behavior (something you can do)	drawing a pattern hand gestures
Location (somewhere you are)	GPS coordinates IP address

Animation content:

Static image: A table with three columns: icon, authentication factor, and examples. First row icon: Brain. First row authentication factor: Knowledge (something you know). First row examples: Password and PIN. Second row icon: Fingerprint. Second row authentication factor: Inherence (something you are). Second row examples: Fingerprint and iris pattern. Third row icon: Smart card. Third row authentication factor: Possession (something you have). Third row examples: Smart card and security token. Fourth row icon: Finger drawing pattern. Fourth row authentication factor: Behavior (something you can do). Fourth row examples: Drawing a pattern and hand gestures. Fifth row icon: Compass. Fifth row authentication factor: Location (somewhere you are). Fifth row examples: GPS coordinates and IP address.

Animation captions:

1. A knowledge factor is "something you know" such as a password or a PIN.
2. An inherence factor is "something you are" such as a fingerprint or an iris pattern.
3. Possession factor is "something you have" such as a smart card or a security token.
4. Behavior factor is "something you can do" such as drawing a pattern or hand gestures.
5. Location factor is "somewhere you are" such as a geolocation.

PARTICIPATION ACTIVITY 2.2.2: Authentication factors.

How to use this tool

Behavior factor **Location factor** **Possession factor** **Knowledge factor** **Inherence factor**

	A piece of information that a user knows.
	Something a user possesses or owns.
	A unique physical characteristic of a user.
	An action a user performs.
	The information on a user's current location.

Reset

PARTICIPATION ACTIVITY 2.2.3: Authentication.

1) A password is a _____ factor.

- knowledge
- possession
- inherence

2) A smart card is a _____ factor.

- location
- knowledge
- possession

3) A user's hand gesture is a _____ factor.

- behavior
- possession
- inherence

4) A fingerprint is a _____ factor.

- possession
- knowledge
- inherence

5) A user's GPS coordinates is a _____ factor.

- behavior
- knowledge
- location

6) A user's iris pattern is a _____ factor.

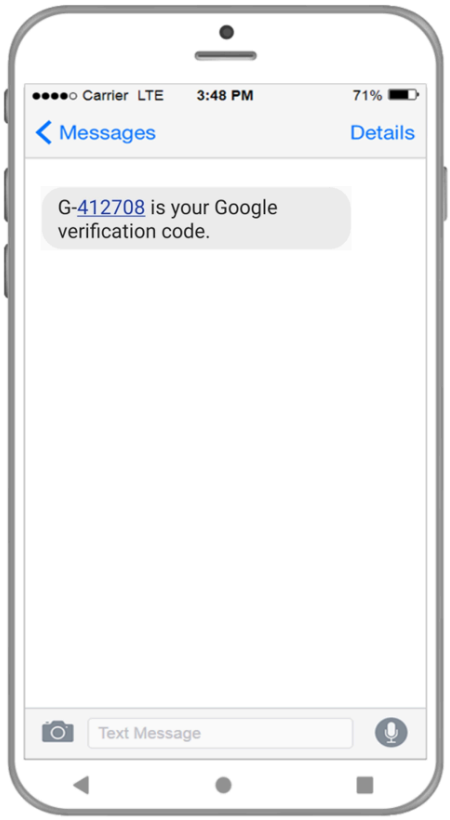
- knowledge
- possession
- inherence

Single and multi-factor authentication

Single-factor authentication (SFA) is an authentication method that uses only one authentication factor to verify a claimed identity. SFA is the most basic form of authentication and is most often implemented as a password. Ex: A secured system using SFA may only require a user's smart card (*possession factor*) or a user's password (*knowledge factor*).

Two-factor authentication, also known as ***2FA***, is an authentication method that verifies a user's claimed identity by using two different authentication factors. Ex: A secured system using 2FA may require a user's password (*knowledge factor*) and a verification code sent to a user's mobile phone (*possession factor*).

Example 2.2.1: A Google verification code on a mobile phone.



A smart card can be used for two-factor authentication (2FA). ***Smart card authentication*** refers to the use of a smart card for authenticating to a secured system. Smart card authentication uses a user's possession of a smart card (*possession factor*) and a user's PIN (*knowledge factor*).

Multi-factor authentication (MFA) is an authentication method that uses two or more authentication factors to verify a claimed identity. A 2FA is a type of MFA. The security of a system improves as the number of authentication factors to verify a claimed identity increases. However, a greater number of authentication factors negatively impacts user experience and reduces system usability.

PARTICIPATION ACTIVITY 2.2.4: Single and multi-factor authentication.

How to use this tool

2FA SFA MFA

	An authentication method that uses only one authentication factor.

	An authentication method that uses more than one authentication factor.
	An authentication method that uses two authentication factors.

Reset

PARTICIPATION ACTIVITY 2.2.5: Single and multi-factor authentication.

1) How many authentication factors are used by a secured system that requires a fingerprint and a password for authentication?
   - One
   - Two
   - Three

2) How many authentication factors are used by a secured system that requires a fingerprint, a password, and a PIN for authentication?
   - One
   - Two
   - Three

3) How many authentication factors are used by a secured system that requires a password and a PIN for authentication?
   - One
   - Two
   - Three

4) How many authentication factors are used by a secured system that requires a PIN, a fingerprint, and a smart card for authentication?
   - One
   - Two

- Three

5) How many authentication factors are used by a secured system that requires a fingerprint and an iris scan for authentication?
   - One
   - Two
   - Three

6) How many authentication factors are used by a secured system that requires a PIN and a user's location for authentication?
   - One
   - Two
   - Three

Knowledge-based authentication

Knowledge-based authentication (KBA) is an authentication method that requires knowledge of a user's personal information to validate a user's claimed identity. KBA is commonly used in a multi-factor authentication system. Two types of KBA exist:

- ***Static KBA***, also known as ***shared secrets*** or ***shared secret questions***, is based on a pre-agreed set of shared questions between an authentication system and a user. A user selects answers to a set of static questions during the registration process with an authentication system. Ex: The name of a user's best friend or a user's birth city.
- ***Dynamic KBA*** is based on knowledge questions that are not set by a user beforehand. In dynamic KBA, authentication questions are compiled from a user's public and private information, such as credit reports and history of financial transactions. Ex: A user's personal loan balances or credit card limits.

Dynamic KBA is more effective than static KBA because of the breadth and depth of questions that reference both a user's current and a user's historical information. An authentication system that uses dynamic KBA may generate diversionary questions designed to trick a user. Ex: Presenting a list of phone numbers and asking a user to select a number that the user never had.

PARTICIPATION ACTIVITY 2.2.6: Knowledge-based authentication.

How to use this tool

Static KBA | Dynamic KBA | Shared secrets

	Is based on a pre-agreed set of shared questions between an authentication system and a user.
	Is based on knowledge questions compiled from public and private information.
	A user's answers to a set of questions by an authentication system.

Reset

PARTICIPATION ACTIVITY 2.2.7: Knowledge-based authentication.

1) Which type of KBA authentication question is based on a pre-agreed set of shared secrets?
   - Static
   - Dynamic

2) In which type of KBA does a user select an authentication question?
   - Static
   - Dynamic

3) A question asking for a user's credit card balances is likely generated by an authentication system using which type of KBA?
   - Static
   - Dynamic

4) A diversionary authentication question is compiled by which type of KBA?
   - Static
   - Dynamic

CHALLENGE ACTIVITY | 2.2.1: Authentication factors.

581480.4344582.qx3zqy7

Start

Select the authentication factor used in each scenario.

A security system uses a retina scanner to control access to an office.

Pick

A bank sends a verification code to a user's mobile phone.

Pick

1 2

Check Next

2.3 Authentication: Methods

One-time password

A common implementation of the possession authentication factor is the use of one-time passwords. A ***one-time password (OTP)*** is an authentication code that can only be used once. An OTP is a 6 to 10 digit code and is commonly used by an authentication system that deploys 2FA.

An OTP is more secure than a password. An OTP prevents a user from sharing the user's password with another party and from using the same password on multiple systems. An OTP removes the need for having a complex password that is difficult for a user to remember.

Two types of OTP exist:

- ***Time-based one-time password (TOTP)*** is a one-time password that changes periodically. A TOTP is generated using a timer and a secret key. A change in a TOTP is based on an increment

of time called a ***timestep***. A timestep is between 30 to 180 seconds. A TOTP is valid for the duration of a timestep.

- ***HMAC-based one-time password (HOTP)***, or ***event-based OTP***, is a one-time password that changes based on an event. An HOTP is generated using a counter and a secret key. An HOTP counter is incremented by an event such as the press of a button on an HOTP generator device. An HOTP is valid until the HOTP is used for authentication or until a new HOTP is generated.

An OTP can be generated by a software application or a hardware device, or delivered to a user by email, Short Message Service (SMS), or phone call.

PARTICIPATION ACTIVITY 2.3.1: One-time password.

How to use this tool

OTP HOTP TOTP

	An authentication code that can only be used once and can be of two types.
	A one-time password that changes periodically without a user's action.
	A one-time password that changes periodically or by a user's action.

Reset

PARTICIPATION ACTIVITY 2.3.2: One-time password.

1) Which type of OTP is based on time?

- TOTP
- HOTP

2) Which type of OTP is based on an event such as a user's action?

- TOTP
- HOTP

3) Which type of OTP uses a counter and a secret key?

- TOTP
- HOTP

4) In which type of OTP is an authentication code valid until the authentication code is used?

- TOTP
- HOTP

5) An OTP can only be generated by a software application.

- False
- True

6) An OTP can be delivered to a user by email, SMS, or phone call.

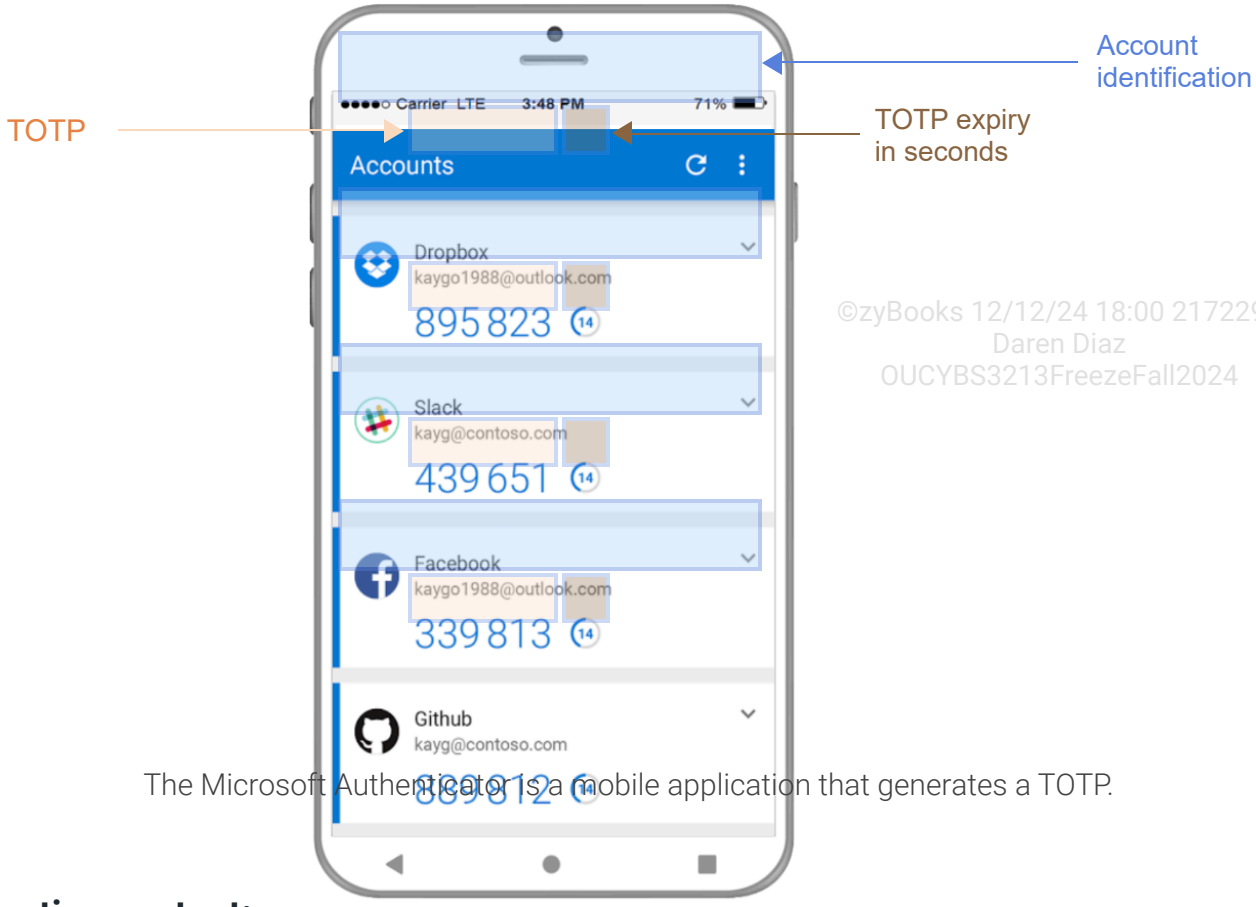
- False
- True

Software-based OTP

An ***authentication application***, also known as a ***software token***, is an application that generates an OTP. Ex: Google's Authenticator is a mobile authentication application that can generate a TOTP or an HOTP. Google's Authenticator is used in Google's 2-Step Verification.

An authentication application must be configured for use with an authentication server. An authentication server sends a secret key to an authentication application. The secret key is used by the authentication application to generate OTPs for all future authentication requests to the authentication server.

PARTICIPATION ACTIVITY 2.3.3: Microsoft Authenticator.



The Microsoft Authenticator is a mobile application that generates a TOTP.

Animation content:

Static image: A mobile phone showing the Microsoft Authenticator app with four accounts: Dropbox, Slack, Facebook, and Github. Each account name is highlighted in blue. The first account name is labeled "Account identification." Below each account name is a box with six numbers and a box with a countdown timer. The first box with six numbers is labeled "TOTP." The first countdown timer is labeled "TOTP expiry in seconds."

Animation captions:

1. The application lists the user's account identification (email address) at each service provider (Dropbox, Slack, Facebook, and GitHub).
2. The TOTP generated by the Microsoft Authenticator application for each service provider. In this case, the TOTP consists of 6 numbers.
3. The time left to the expiry of each TOTP (14 seconds). Upon expiry, a new set of TOTP numbers will be generated.

An OTP can be delivered to a user by an email, SMS, or a phone call. A ***push notification*** is a notification sent by an authentication server to a mobile device associated with a user. A push notification informs a user of an authentication attempt with the user's credentials. An authentication attempt is approved if the user performs a specific action, such as pushing a button on the user's phone or opening an application. A push notification uses a possession factor (a user's mobile phone) to validate a user's identity.

PARTICIPATION ACTIVITY 2.3.4: Software-based OTP.

How to use this tool

Software token **Push notification** **Seed**

	A software application that generates an OTP
	A notification sent by an authentication server to a mobile device associated with a user
	A secret key shared between an authentication server and a software token

Reset

PARTICIPATION ACTIVITY 2.3.5: Software-based OTP.

1) An authentication application can generate a valid OTP for any authentication server.
   - False
   - True

2) An authentication server that sends a push notification to validate a user's identity uses a knowledge factor.
   - False
   - True

3) An OTP can be delivered to a user by push notification, SMS, email or phone call.
   - False
   - True

4) In push notification, a user approves an authentication attempt by pushing a specific button on the user's mobile device.

- False
- True

Hardware-based OTP

A ***security token***, also known as a ***token key***, ***security key***, or ***password key***, is a hardware device that generates an OTP. A security token has an encoded secret key that is only shared with an authentication server. A security token does not store an OTP on a networked device such as a mobile phone or an email client, and is thus more secure than an authentication method that relies on SMS or email for OTP delivery. Ex: RSA Security's SecurID is a security token.

Example 2.3.1: RSA SecurID token.

Credit: RSA Security, LLC.[1]

An OTP can be pre-generated and used when a hardware or software security token and a network connection does not exist. A ***static code*** is a pre-generated OTP. A static code can be generated by a hardware or software-based OTP generator and saved on a secured storage media for later use.

PARTICIPATION ACTIVITY 2.3.6: Hardware-based OTP.

How to use this tool

Secret key **Static code** **Token key**

	A hardware device that generates an OTP
	A pre-generated OTP

	A value shared between a security token and an authentication server

Reset

PARTICIPATION ACTIVITY 2.3.7: Hardware-based OTP.

1) A token key is also known as a password key.
   - False
   - True

2) A static code is an pre-generated OTP.
   - False
   - True

3) A security key is less secure than an authentication method that relies on SMS or email for OTP delivery.
   - False
   - True

4) A security token has an encoded secret key that is shared with an authentication server.
   - False
   - True

5) A password key stores an OTP on a networked device.
   - False
   - True

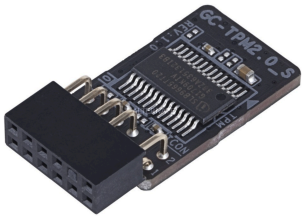
Hardware-based security modules

A hardware device can be authenticated to ensure that an unauthorized operation is not performed on data and systems. Two hardware-based technologies support device authentication: Trusted Platform Module (TPM) and Hardware Security Module (HSM).

A ***Trusted Platform Module (TPM)*** is a secure processor that performs cryptographic operations. A TPM can be embedded on a computer motherboard or installed on a computer motherboard that has a TPM port. A TPM has a burned-in RSA key-pair which can be used to authenticate a hardware device. A TPM can store cryptographic keys, passwords, and certificates. A TPM is tamper-resistant and can help prevent unauthorized modifications to firmware and software as part of a secure (trusted) boot process.

An operating system can use a TPM for full disk encryption (FDE). ***Full disk encryption (FDE)*** is the encryption of an entire drive including all user and operating system files and software programs. Ex: Microsoft Windows BitLocker is a full disk encryption software that can store a drive's encryption key in a TPM. A Windows user can encrypt the user's drive using BitLocker.

Example 2.3.2: Trusted Platform Module (TPM).



Credit: Gigabyte Technology.[2]

A ***hardware security module (HSM)*** is a tamper-resistant external device or a plug-in expansion card that provides cryptographic services. An HSM can store a hardware device's certificate for authentication, cryptographic keys, and passwords. Ex: HSMs are used for real-time authentication and authorization of credit and debit card transactions in the payment card industry. An HSM contains one or more cryptoprocessors for creating, securing, storing, and managing encryption keys. A ***cryptoprocessor*** is a dedicated microprocessor that performs cryptographic operations such as encryption.

Example 2.3.3: Hardware security module (HSM).

Credit: Thales.[3]

A TPM and an HSM can be certified against a security standard such as FIPS-140. The Federal Information Processing Standards (FIPS) 140 publication series, also known as ***FIPS-140***, is the U.S. government computer security standard that specifies the requirements for cryptographic modules.

PARTICIPATION ACTIVITY 2.3.8: Hardware-based security modules.

How to use this tool

TPM **FDE** **Cryptoprocessor** **HSM**

	A secure processor that performs cryptographic operations
	Encryption of an entire drive including all user and operating system files and software programs
	A dedicated microprocessor that performs cryptographic operations
	A tamper-resistant external device or a plug-in expansion card that provides cryptographic services

Reset

PARTICIPATION ACTIVITY 2.3.9: Hardware-based security modules.

1) A _____ is a plug-in expansion card.

- TPM
- HSM
- TPM or HSM

2) A _____ can perform a cryptographic operation such as encryption.

- TPM
- HSM
- TPM or HSM

3) A _____ can store keys, certificates, and passwords.

- TPM
- HSM
- TPM or HSM

4) A _____ is used in full disk encryption (FDE).

- TPM
- HSM
- TPM or HSM

5) A _____ can obtain certification against a security standard such as FIPS-140.

- TPM
- HSM
- TPM or HSM

Secure enclave

*A **secure enclave** is a tamper-resistant hardware component isolated within a system or device that provides a secure environment for cryptographic operations. A secure enclave ensures sensitive data remains encrypted and inaccessible to other software, including the operating system. Secure enclaves enable devices to comply with data protection regulations such as regulations for digital rights management and secure payment processing.*

(*1) RSA Security, LLC. "RSA SecureID". https://www.rsa.com/en-us/products/rsa-securid-suite/rsa-securid-access/modern-authentication-methods.

(*2) Gigabyte Technology. "Gigabyte Technology TPM". https://www.gigabyte.com/us/Motherboard/GC-TPM20_S#ov

(*3) Thales HSM. "Thales". https://www.thales.com

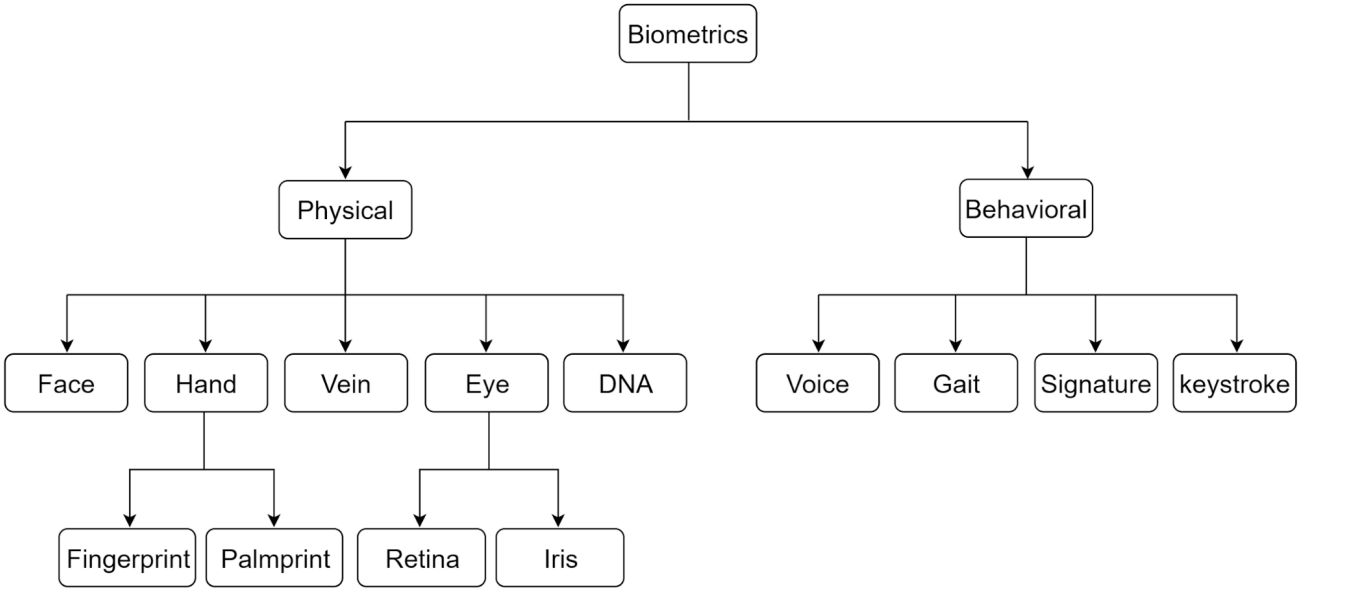
2.4 Authentication: Biometrics

Biometrics

Biometrics are measurements of the unique characteristics of an individual. Ex: A fingerprint or a voice. Two types of biometrics exist:

- ***Physical biometrics*** are measurements of an individual's physical characteristics, such as a fingerprint, palmprint, face, retina, iris, and vein patterns.
- ***Behavioral biometrics*** are measurements of an individual's behavioral characteristics, such as voice, gait, signature, and keystroke.

Figure 2.4.1: Biometrics.



PARTICIPATION ACTIVITY 2.4.1: Biometrics.

1) Fingerprint is a _____ biometric.
   - physical
   - behavioral

2) Gait is an individual's walking pattern.
   Gait is a _____ biometric.

- physical
- behavioral

3) Iris pattern is a _____ biometric

- physical
- behavioral

4) Retina pattern is a _____ biometric.

- physical
- behavioral

5) Face is a _____ biometric.

- physical
- behavioral

6) Voice is a _____ biometric.

- physical
- behavioral

7) Vein pattern is a _____ biometric.

- physical
- behavioral

Biometric authentication is an authentication method that uses physical and behavioral biometrics to verify a claimed identity. Biometric authentication is more secure than knowledge or token-based authentication. A password or a PIN may be shared or guessed, and a token may be lost or stolen. However, a biometric is unique to an individual and cannot be transferred, lost, or forgotten. A biometric is nearly impossible to duplicate.

PARTICIPATION ACTIVITY 2.4.2: Biometrics.

How to use this tool

Physical biometrics | **Behavioral biometrics** | **Biometric authentication** | **Biometrics**

The biometric measurements of an individual's fingerprint, face, retina, iris,

	and vein patterns.
	The biometric measurements of an individual's voice and gait.
	An authentication method that uses physical and behavioral biometrics to verify a claimed identity.
	The measurements of the unique characteristics of an individual.

Reset

Biometric factors

A ***biometric factor***, also known as an ***inherence factor***, is a unique physical or behavioral characteristic of an individual. A biometric factor is ***something you are***. Seven biometric factors are commonly used for authentication:

- A ***fingerprint*** is the pattern of ridges and valleys on an individual's finger. A finger is scanned using an ultrasonic, optical, or capacitive scanner.
- A ***retina*** is a layer of nerve cells lining the back of an eye. A ***retina pattern*** is the blood vessel pattern in an individual's retina. A retina is scanned using low-energy infrared light.
- An ***iris*** is the colored portion of an eye that surrounds a pupil. An ***iris pattern*** is the pattern of an individual's iris. An iris is scanned using video camera technology.
- ***Facial recognition*** involves measuring the distance between various points on an individual's face to create a faceprint. A ***faceprint*** is a digitally recorded representation of an individual's face. An individual's face measurements may include the length of a jawline, shape of a cheekbone, and width of a nose. A video, infrared, or thermal camera is used to scan an individual's face.
- ***Voice recognition*** uses a microphone to measure and analyze the rhythms, patterns, and sounds of an individual's voice.
- ***Vein recognition*** uses an infrared-light scanner to capture and analyze an individual's blood vessel patterns. Unlike a fingerprint scanner, a vein scanner does not have to be in contact with an individual's skin.
- ***Gait*** is an individual's walking pattern. ***Gait analysis*** is the study of an individual's body movements and muscle activity during motion.

PARTICIPATION ACTIVITY 2.4.3: Biometric factors.

How to use this tool

Iris | Retina | Gait | Fingerprint | Faceprint

	The colored portion of an eye that surrounds a pupil.
	The layer of nerve cells lining the back of an eye.
	An individual's walking pattern.
	The pattern of ridges and valleys on an individual's finger.
	A digitally recorded representation of an individual's face.

Reset

PARTICIPATION ACTIVITY 2.4.4: Biometric factors.

1) Which biometric factor is scanned by a thermal camera?
   - Vein
   - Fingerprint
   - Face

2) Which biometric factor involves rhythms and patterns of sound?
   - Voice
   - Retina
   - Face

3) Which biometric factor relates to an individual's body movements and muscle activity during motion?
   - Voice

- Gait
- Iris

4) Which biometric factor is scanned using a video camera?

- Retina
- Fingerprint
- Iris

5) Which biometric factor is scanned using an ultrasonic scanner?

- Retina
- Fingerprint
- Vein

Biometric errors

The performance of a biometric system is measured using three metrics:

- ***False rejection rate (FRR)***, or ***type I error***, is the percentage of valid biometric measures that are rejected by a biometric system.
- ***False acceptance rate (FAR)***, or ***type II error***, is the percentage of invalid biometric measures that are accepted by a biometric system.
- ***Crossover error rate (CER)***, also known as ***equal error rate (EER)***, is the rate at which FRR and FAR are equal.

The effectiveness or ***efficacy rate*** of a biometric system is a measure of the system's ability to minimize false acceptance and prevent false rejection. The FIDO Alliance, which certifies biometric systems, has established an FRR threshold of 3% (3 in 100) and a FAR threshold of 0.01% (1 in 10,000).

PARTICIPATION ACTIVITY 2.4.5: Biometric errors.

How to use this tool

False rejection rate (FRR) **False acceptance rate (FAR)** **Crossover error rate (CER)**

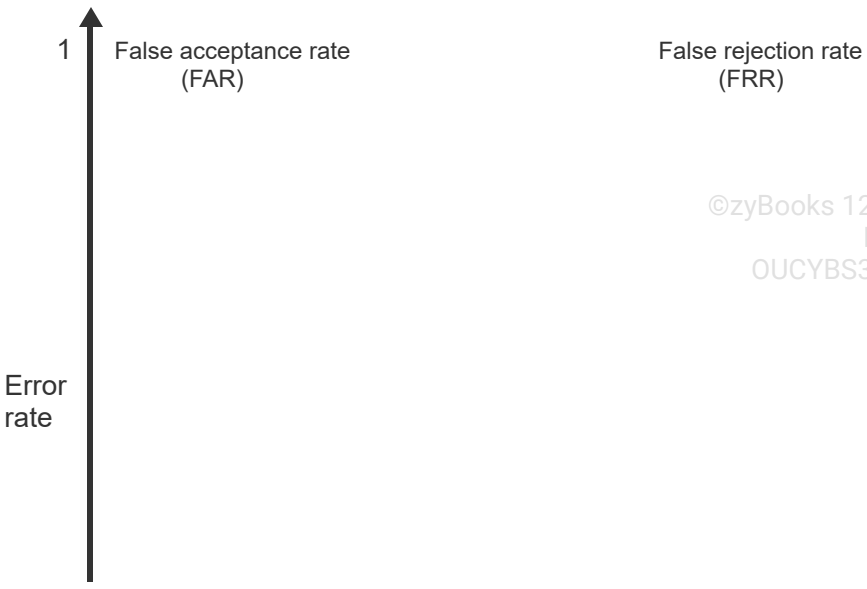
Efficacy rate

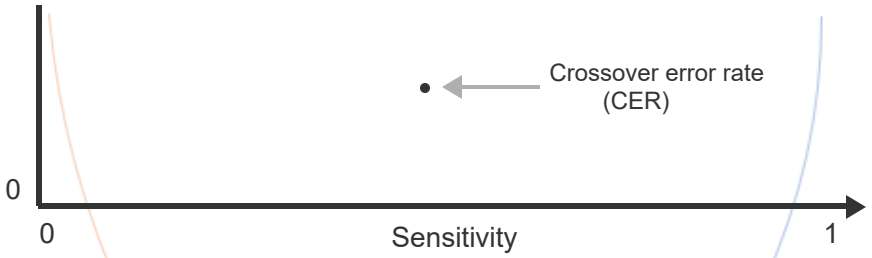
	The percentage of valid biometric measures that is rejected by a biometric system.
	The percentage of invalid biometric measures that is accepted by a biometric system.
	The rate at which a biometric system's false rejection rate (FRR) and false acceptance rate (FAR) are equal.
	A measure of a biometric system's ability to minimize false acceptance and prevent false rejection.

Reset

As the likelihood of false rejection (FRR) is decreased, the rate of false acceptance (FAR) is increased. The ***relative operating characteristic (ROC)*** graph is a visual characterization of the trade-off between an FRR and a FAR. On an ROC graph, the CER is the point at which FAR and FRR intersect or crossover. CER is used to compare the accuracy of different biometric systems. A biometric system with the lowest CER is the most accurate.

PARTICIPATION ACTIVITY 2.4.6: Crossover error rate.





Animation content:

Static image: A graph. The x-axis is labeled "Sensitivity" and goes from 0 to 1. The y-axis is labeled "Error rate" and goes from 0 to 1. A blue curved line labeled "False rejection rate (FRR)" starts at (0, 0) and curves upward to (1, 1). An orange curved line labeled "False acceptance rate (FAR)" starts at (0, 1) and curves downward to (1, 0). The curves intersect at about (0.5, 0.2). The intersection is labeled "Crossover error rate (CER)."

Animation captions:

1. A biometric system's false rejection rate (FRR) increases as the biometric system's sensitivity increases.
2. A biometric system's false acceptance rate (FAR) decreases as the biometric system's sensitivity increases.
3. The crossover error rate (CER) is the point that FAR and FRR intersect or crossover. The CER is the rate at which both FRR and FAR are equal.

PARTICIPATION ACTIVITY 2.4.7: Biometric errors.

1) Which is the most important error rate when comparing different biometric systems?
   - FAR
   - FRR
   - CER

2) Which of the following provides the best indication of a biometric system's accuracy?
   - Low FAR
   - Low FRR

- Low CER

3) Which of the following statements is true?

- FRR is increased as a biometric system's sensitivity increases
- FRR is decreased as a biometric system's sensitivity increases
- FRR remains the same as a biometric system's sensitivity increases

4) Which of the following statements is true?

- FAR increases as a biometric system's sensitivity increases
- FAR decreases as a biometric system's sensitivity increases
- FAR remains the same as a biometric system's sensitivity increases

CHALLENGE ACTIVITY 2.4.1: Authentication (biometrics).

581480.4344582.qx3zqy7

Start

What is the biometric type of each biometric factor?

Signature	Pick
Keystroke	Pick
Gait	Pick
Vein pattern	Pick
Retina pattern	Pick

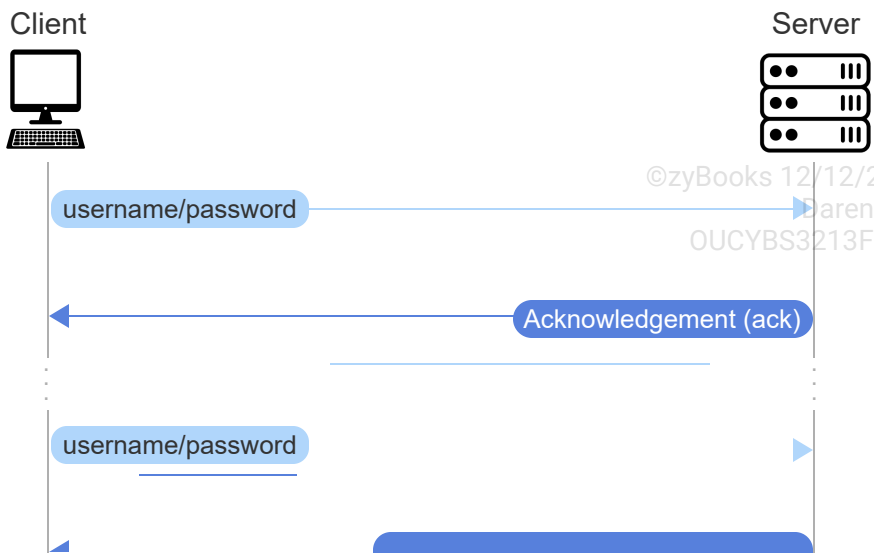
2.5 Authentication protocols: PAP and CHAP

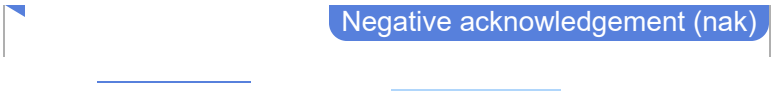
Password authentication protocol (PAP)

Password authentication protocol (PAP) is an authentication protocol used for authenticating a client to a server over a point-to-point connection. PAP authentication is only done once per session and at the time of the initial connection establishment.

PAP uses a two-way handshake protocol. A client sends the client's username and password to a server. The server sends an authentication acknowledgement (ack) to the client if a client's username and password are correct, or a negative acknowledgement (nak) if a client's username or password is incorrect. PAP does not encrypt usernames and passwords, and is thus not considered a secure authentication protocol.

PARTICIPATION ACTIVITY 2.5.1: Password authentication protocol (PAP).





Animation content:

Static image: A computer icon on the left labeled "Client." A server icon on the right labeled "Server." Messages are shown passing between the client and the server. The first message says "username/password" and goes from the client to the server. The second message says "Acknowledgement (ack)" and goes from the server to the client. The third message says "username/password" and goes from the client to the server. The fourth message says "Negative acknowledgement (nak)" and goes from the server to the client.
Step 1: A client sends a username and a password to a server.
A computer icon on the left labeled "Client." A server icon on the right labeled "Server." A message saying "username/password" is shown going from the client to the server.
Sep 2: If the username and password are correct, the server sends an acknowledgment (ack) to the client.
A message saying "Acknowledgement (ack)" is shown going from the server to the client.
Step 3: If the username and password are incorrect, the server sends a negative acknowledgment (nak) to the client.
The messages are grayed out. A new message saying "username/password" is shown going from the client to the server. A new message saying "Negative acknowledgement (nak)" is shown going from the server to the client.

Animation captions:

1. A client sends a username and a password to a server.
2. If the username and password are correct, the server sends an acknowledgment (ack) to the client.
3. If the username and password are incorrect, the server sends a negative acknowledgment (nak) to the client.

PARTICIPATION ACTIVITY 2.5.2: Password authentication protocol (PAP).

1) How often is a client authenticated to a server?

   - ○ Once per session
   - ○ Multiples times during a session

2) What type of message is sent from a server to a client if a password is correct but not the username?

- Acknowledgement (ack)
- Negative acknowledgement (nak)

3) How is a username and a password sent from a client to a server?

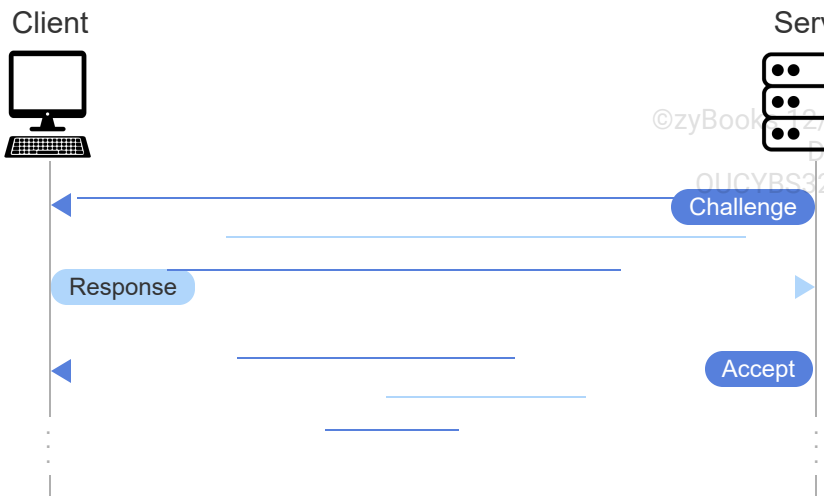
- encrypted
- unencrypted (plaintext)

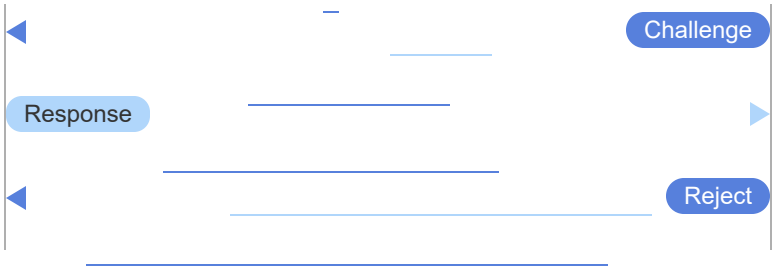
Challenge handshake authentication protocol (CHAP)

Challenge handshake authentication protocol (CHAP) is an authentication protocol that uses a shared secret to authenticate a client to a server over a point-to-point connection. Unlike PAP, CHAP periodically re-authenticates a client during a communication session.

CHAP uses a three-way handshake protocol. A server sends a randomly generated challenge string to a client. The client combines the server's challenge string with a secret shared with the server. The client computes the hash value of the combined string and sends the hash value to the server. The server compares the hash value received from the client with the server's own calculated hash value of the challenge string and the shared secret with the client. If the hash values are equal, the client is authenticated to the server. If the hash values are not equal, the client's authentication attempt fails.

PARTICIPATION ACTIVITY 2.5.3: Challenge handshake authentication protocol (CHAP).





Animation content:

Static image: A computer icon on the left labeled "Client." A server icon on the right labeled "Server." Messages are shown passing between the client and the server. The first message is labeled "Challenge" and goes from the server to the client. The second message is labeled "Response" and goes from the client to the server. The third message is labeled "Accept" and goes from the server to the client. The fourth message is labeled "Challenge" and goes from the server to the client. The fifth message is labeled "Response" and goes from the client to the server. The sixth message is labeled "Reject" and goes from the server to the client.
Step 1: A server sends a challenge to a client.
A computer icon on the left labeled "Client." A server icon on the right labeled "Server." A message labeled "Challenge" is shown going from the server to the client.
Sep 2: A client responds to the server's challenge.
A message labeled "Response" is shown going from the client to the server.
Step 3: If the response is correct, the server sends an accept message to the client.
A message labeled "Accept" is shown going from the server to the client.
Step 4: If the response is not correct, the server sends a reject message to the client.
The messages are grayed out. A new message labeled "Challenge" is shown going from the server to the client. A new message labeled "Response" is shown going from the client to the server. A new message labeled "Reject" is shown going from the server to the client.

Animation captions:

1. A server sends a challenge to a client.
2. A client responds to the server's challenge.
3. If the response is correct, the server sends an accept message to the client.
4. If the response is not correct, the server sends a reject message to the client.

PARTICIPATION ACTIVITY 2.5.4: PAP and CHAP.

1) In which authentication protocol is a shared secret used between a client

and a server?

- PAP
- CHAP

2) In which authentication protocol is a user periodically authenticated during a communication session?

- PAP
- CHAP

3) In which authentication protocol is a password sent from a client to a server?

- PAP
- CHAP

4) In which authentication protocol is a hash function used by a client and a server?

- PAP
- CHAP

5) In which authentication protocol is a client authenticated only once and at the beginning of a communication session?

- PAP
- CHAP

2.6 Authentication protocols: Kerberos

Kerberos

Kerberos is an authentication protocol that uses a ticket-based mechanism to authenticate a user and enable a user to access a network service. Kerberos uses UDP port 88 by default. Ex: Kerberos is the default authentication protocol in Windows Server 2019. A ***Kerberos realm*** is a network that uses Kerberos authentication.

A Kerberos authentication involves three entities:

- Server
- Client
- Key Distribution Center (KDC)

A server hosts a service that a user wants to access from a client. A ***Key Distribution Center (KDC)*** is a trusted third-party that authenticates a user and enables a user to access a service hosted on a server. A KDC consists of an Authentication Server (AS) and a Ticket-Granting Server (TGS). An ***Authentication Server (AS)*** authenticates a user. A ***Ticket-Granting Server (TGS)*** issues a ticket to a user that enables a user to access a service.

PARTICIPATION ACTIVITY 2.6.1: Kerberos.

1) Kerberos uses _____ by default.
   - UDP port 1812
   - UDP port 88

2) A(n) _____ issues a ticket to a user that enables a user to access a service.
   - Authentication Server (AS)
   - Ticket-Granting Server (TGS)

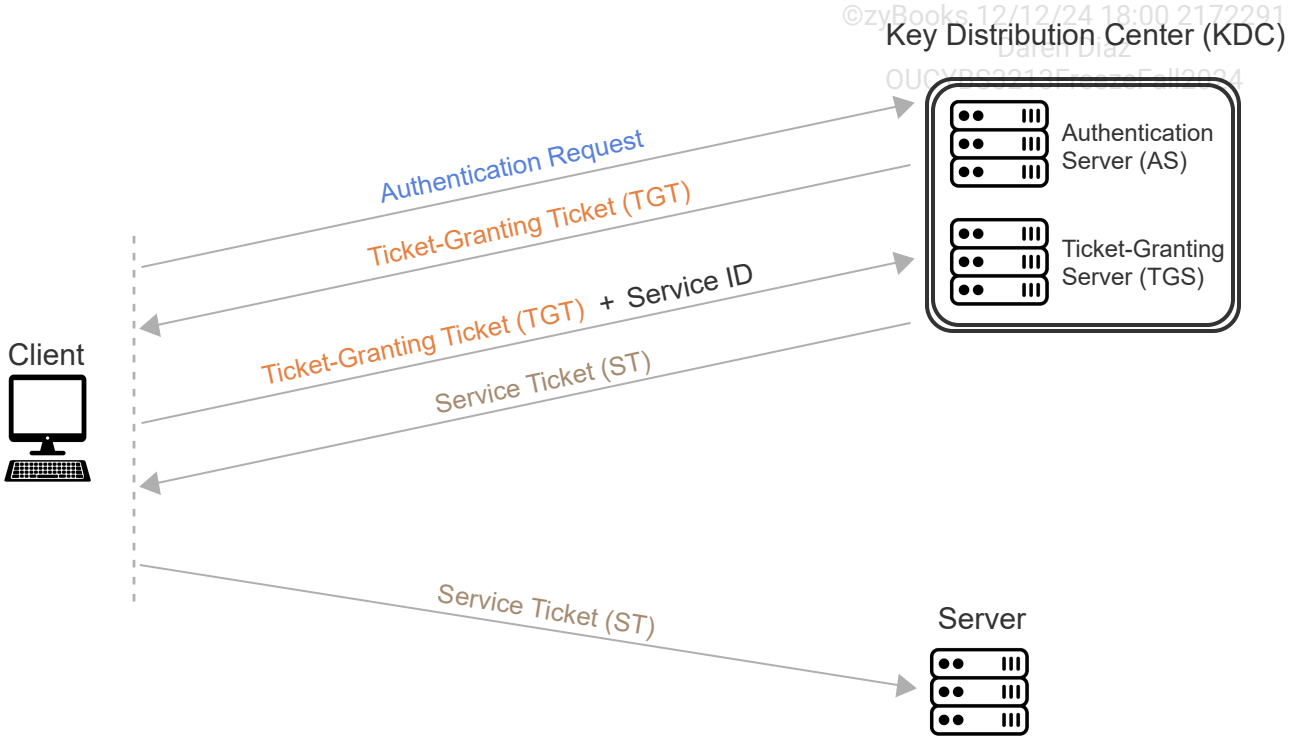
3) A(n) _____ authenticates a user.
   - Authentication Server (AS)
   - Ticket-Granting Server (TGS)

Kerberos authentication

A user that wants to authenticate to a network sends an authentication request message to an Authentication Server (AS). An authentication request message contains a user's identity information and credentials. An AS validates a user's identity and sends a user a Ticket-Granting Ticket (TGT). A ***Ticket-Granting Ticket (TGT)*** is a ticket that enables a user to request access to a service from a TGS.

A user that wants to access a service sends the user's TGT and the service's identification number (ID) to a TGS. The user's TGT proves to the TGS that the user is previously authenticated. A TGS responds by sending a Service Ticket (ST) to the user. A ***Service Ticket (ST)*** is an encrypted message that provides proof that a user is authorized to access a service. An ST is only valid for the duration of a single session. A user that wants to access the same service later should request a new ST from a TGS.

PARTICIPATION ACTIVITY 2.6.2: Kerberos.



Animation content:

Static image: A computer icon labeled "Client" on the left, a box labeled "Key Distribution Center (KDC)" in the top-right corner, and a server icon labeled "Server" below the Key Distribution Center. A server labeled "Authentication Server (AS)" and a server labeled "Ticket-Granting Server (TGS)" are in the KDC. An arrow labeled "Authentication Request" points from the Client to the KDC. An arrow labeled "Ticket-Granting Ticket (TGT)" points from the KDC to the Client. An arrow labeled "Ticket-Granting Ticket (TGT) + Service ID" points from the Client to the KDC. An arrow labeled "Service Ticket (ST)" points from the KDC to the Client. An arrow labeled "Service Ticket (ST)" points from the Client to the Server.

Step 1: A user sends an authentication request to a Key Distribution Center (KDC).

A computer icon labeled "Client" on the left, a box labeled "Key Distribution Center (KDC)" in the top-right corner, and a server icon labeled "Server" below the Key Distribution Center. A server icon labeled "Authentication Server (AS)" and a server icon labeled "Ticket-Granting Server (TGS)" are in the KDC. An arrow labeled "Authentication Request" appears pointing from the Client to the KDC.

Step 2: A KDC's Authentication Server (AS) authenticates a user and sends a user a Ticket-Granting

Ticket (TGT).
An arrow labeled "Ticket-Granting Ticket (TGT)" appears pointing from the KDC to the Client.
Step 3: A user that wants to access a service sends the user's TGT and the service's ID to a KDC's Ticket-Granting Server (TGS).
An arrow labeled "Ticket-Granting Ticket (TGT) + Service ID" appears pointing from the Client to the KDC.
Step 4: A Ticket-Granting Server (TGS) sends a user a Service Ticket (ST) for the user's requested service.
An arrow labeled "Service Ticket (ST)" appears pointing from the KDC to the Client.
Step 5: A user sends the Service Ticket (ST) to the Server hosting the service and gains access to the service.
An arrow labeled "Service Ticket (ST)" appears pointing from the Client to the Server.

Animation captions:

1. A user sends an authentication request to a Key Distribution Center (KDC).
2. A KDC's Authentication Server (AS) authenticates a user and sends a user a Ticket-Granting Ticket (TGT).
3. A user that wants to access a service sends the user's TGT and the service's ID to a KDC's Ticket-Granting Server (TGS).
4. A Ticket-Granting Server (TGS) sends a user a Service Ticket (ST) for the user's requested service.
5. A user sends the Service Ticket (ST) to the Server hosting the service and gains access to the service.

PARTICIPATION ACTIVITY | 2.6.3: Kerberos.

How to use this tool

Key Distribution Center (KDC) **Ticket-Granting Ticket (TGT)** **Service Ticket (ST)**

Authentication Server (AS) **Ticket-Granting Server (TGS)**

	A ticket that enables a user to request access to a service from a TGS
	A trusted third-party that authenticates a user and enables a user to access a service hosted on a server

	Authenticates a user and sends a Ticket-Granting Ticket (TGT) to a user
	An encrypted message that provides proof that a user is authorized to access a service.
	Issues a ticket to a user that enables a user to access a service

Reset

PARTICIPATION ACTIVITY 2.6.4: Kerberos.

1) A user sends a Ticket-Granting Ticket (TGT) and a _____ to a Ticket-Granting Server (TGS) to request access to a service.
   - Key Distribution Center (KDC)
   - Service ID
   - Service Ticket (ST)

2) A Ticket-Granting Server (TGS) sends a user a _____ that enables a user to access a service.
   - Service ID
   - Ticket-Granting Ticket (TGT)
   - Service Ticket (ST)

3) An Authentication Server (AS) sends a user a _____ after validating a user's identity.
   - Ticket-Granting Server (TGS)
   - Ticket-Granting Ticket (TGT)
   - Service Ticket (ST)

4) A Ticket-Granting Server (TGS) sends a _____ to a user after a user requests access to a service.
   - Service ID

- Ticket-Granting Ticket (TGT)
- Service Ticket (ST)

5) A Ticket-Granting Server (TGS) and an _____ are the two components of a Key Distribution Center (KDC).
   - Ticket-Granting Ticket (TGT)
   - Authentication Server (AS)
   - Service Ticket (ST)

2.7 Authentication protocols: EAP and IEEE 802.1X

Extensible Authentication Protocol (EAP)

The ***Extensible Authentication Protocol (EAP)*** is an authentication framework for transporting different types of authentication protocols. EAP supports multiple authentication methods, including certificate-based, password-based, and multi-factor authentication.

EAP defines the format of authentication messages. Four EAP message types exist:

- ***EAP Request*** is a message sent from a server to a client requesting information from the client.
- ***EAP Response*** is a message sent from a client to a server in response to a server's EAP Request message.
- ***EAP Success*** is a message sent from a server to a client if the client's authentication is successful.
- ***EAP Failure*** is a message sent from a server to a client if the client's authentication is not successful.

An authentication method that is supported by EAP implements EAP messages using the protocol's messaging formats. EAP allows for a new authentication method to be compatible with existing wireless or point-to-point connection technologies. Over 40 different EAP authentication methods have been defined.

PARTICIPATION ACTIVITY 2.7.1: Extensible Authentication Protocol (EAP).

1) EAP is an authentication protocol.

- False
- True

2) EAP only supports password-based authentication.

- False
- True

3) EAP supports mutual authentication.

- False
- True

Four commonly used EAP authentication methods exist:

- ***Extensible Authentication Protocol-Transport Layer Security***, or ***EAP-TLS***, is an authentication protocol that uses the Transport Layer Security (TLS) and certificates for mutual authentication. EAP-TLS requires both a server and a client certificate.
- ***Extensible Authentication Protocol-Flexible Authentication via Secure Tunneling***, or ***EAP-FAST***, is an authentication protocol that uses a Protected Access Credential (PAC) to establish a Transport Layer Security (TLS) tunnel between a server and a client. A ***Protected Access Credential (PAC)*** is a security credential that is generated by a server and holds information specific to a client. EAP-FAST supports but does not require certificates.
- ***Extensible Authentication Protocol-Tunneled Transport Layer Security***, or ***EAP-TTLS*** is an authentication protocol that uses Transport Layer Security (TLS) and a server certificate to establish a secure tunnel between a server and a client. The server authenticates the client through the TLS tunnel by using any authentication protocol, including legacy password methods such as PAP. EAP-TTLS requires a server certificate but not a client certificate.
- ***Protected Extensible Authentication Protocol***, or ***PEAP***, is an authentication protocol that encapsulates EAP messages within an encrypted and authenticated Transport Layer Security (TLS) tunnel. PEAP requires a server certificate but not a client certificate.

Table 2.7.1: Comparison of EAP methods.

	EAP-TLS	EAP-TTLS	EAP-FAST	PEAP
Mutual authentication	Supported	Supported	Supported	Supported
Server certificate	Required	Required	Optional	Required

Client certificate	Required	Optional	Optional	Optional
Standard	Open/RFC	Internet Draft	Internet Draft	Internet Draft

PARTICIPATION ACTIVITY 2.7.2: Extensible Authentication Protocol (EAP).

How to use this tool

EAP **EAP-TTLS** **EAP-FAST** **EAP-TLS** **PEAP**

	An authentication protocol that encapsulates EAP messages within an encrypted and authenticated Transport Layer Security (TLS) tunnel.
	An authentication protocol that uses a Protected Access Credential (PAC) to establish a Transport Layer Security (TLS) tunnel between a server and a client.
	An authentication protocol that uses the Transport Layer Security (TLS) and server and client certificates for mutual authentication.
	An authentication framework for transporting authentication protocols.
	An authentication protocol that uses Transport Layer Security (TLS) and a server certificate to establish a secure tunnel between a server and a client.

Reset

PARTICIPATION ACTIVITY 2.7.3: Extensible Authentication Protocol (EAP).

1) EAP-TLS requires both server and client certificates.
   - False
   - True

2) PEAP encapsulates EAP messages within an encrypted and authenticated TLS tunnel.
   - False
   - True

3) EAP-TTLS uses PAC to establish a TLS tunnel between a server and a client.
   - False
   - True

4) EAP-TTLS allows for client authentication by legacy password methods.
   - False
   - True

IEEE 802.1X

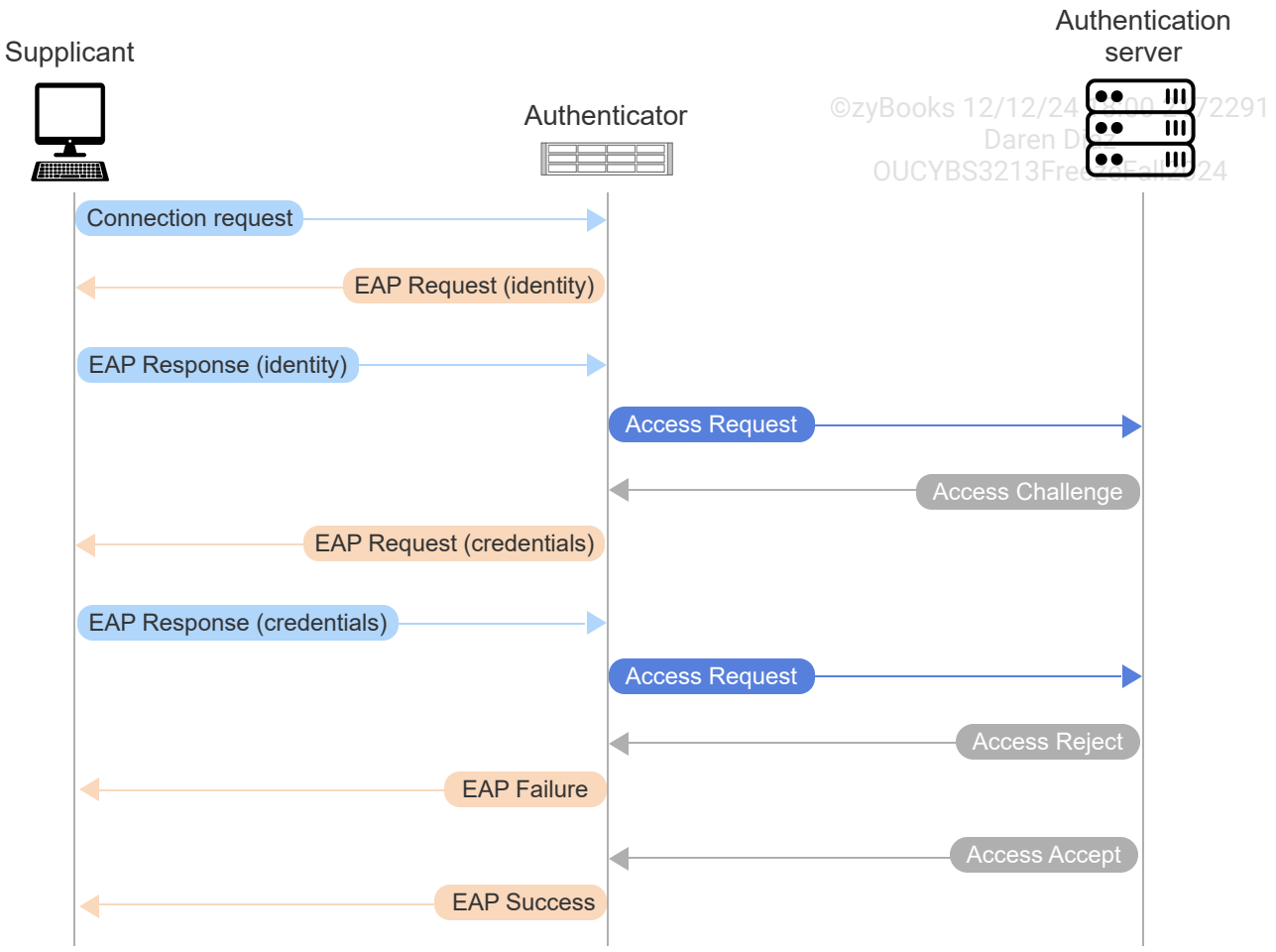
IEEE 802.1X is an IEEE standard for port-based access control. IEEE 802.1X is used for passing EAP messages over a wired or wireless network. ***EAP over LAN***, or ***EAPOL*** is the encapsulation of EAP over IEEE 802.1X. IEEE 802.1X allows a user to connect to a network port only after the user authenticates to the network.

The IEEE 802.1X authentication consists of three entities:

- A ***supplicant*** is a user or a device that wants to authenticate to a network.
- An ***authentication server*** is a server that authenticates a supplicant and makes an access control decision.
- An ***authenticator*** is a device that acts as a proxy for a supplicant and controls a supplicant's communication with an authentication server.

IEEE 802.1X uses EAP to facilitate communication between a supplicant, an authenticator, and an authentication server.

PARTICIPATION ACTIVITY | 2.7.4: IEEE 802.1X authentication.



Animation content:

Static image: An icon labeled "Supplicant," an icon labeled "Authenticator," and an icon labeled "Authentication server." A message labeled "Connection request" points from the Supplicant to the Authenticator. A message labeled "EAP Request (identity)" points from the Authenticator to the Supplicant. A message labeled "EAP Response (identity)" points from the Supplicant to the Authenticator. A message labeled "Access Request" points from the Authenticator to the Authentication server. A message labeled "Access Challenge" points from the Authentication server to the Authenticator. A message labeled "EAP Request (credentials)" points from the Authenticator to the Supplicant. A message labeled "EAP Response (credentials)" points from the Supplicant to the Authenticator. A message labeled "Access Request" points from the Authenticator to the Authentication server. A message labeled "Access Reject" points from the Authentication server to the Authenticator. A message labeled "EAP Failure" points from the Authenticator to the Supplicant. A message labeled "Access Accept" points from the Authentication server to the Authenticator. A message labeled "EAP Success" points from the Authenticator to the Supplicant.

Step 1: A supplicant sends a connection request to an authenticator. The authenticator sends an EAP Request message to the supplicant requesting the supplicant's identity.
An icon labeled "Supplicant," an icon labeled "Authenticator," and an icon labeled "Authentication server." A message labeled "Connection request" appears pointing from the Supplicant to the Authenticator. A message labeled "EAP Request (identity)" appears pointing from the Authenticator to the Supplicant.
Step 2: A supplicant sends the supplicant's identity to the authenticator in an EAP Response message.
A message labeled "EAP Response (identity)" appears pointing from the Supplicant to the Authenticator.
Step 3: The authenticator sends the supplicant's identity to the authentication server and sends the authentication server's response to the supplicant in an EAP Request message.
A message labeled "Access Request" appears pointing from the Authenticator to the Authentication server. A message labeled "Access Challenge" appears pointing from the Authentication server to the Authenticator. A message labeled "EAP Request (credentials)" appears pointing from the Authenticator to the Supplicant.
Step 4: The supplicant sends the supplicant's credentials in an EAP Response message to the authenticator.
A message labeled "EAP Response (credentials)" appears pointing from the Supplicant to the Authenticator.
Step 5: The authenticator sends the supplicant's credentials to the authentication server. If authentication is not successful, the authenticator sends an EAP Failure message to the supplicant.
A message labeled "Access Request" appears pointing from the Authenticator to the Authentication server. A message labeled "Access Reject" appears pointing from the Authentication server to the Authenticator. A message labeled "EAP Failure" appears pointing from the Authenticator to the Supplicant.
Step 6: The "Access Reject" and "EAP Failure" messages are grayed out. A message labeled "Access Accept" appears pointing from the Authentication server to the Authenticator. A message labeled "EAP Success" appears pointing from the Authenticator to the Supplicant.

Animation captions:

1. A supplicant sends a connection request to an authenticator. The authenticator sends an EAP Request message to the supplicant requesting the supplicant's identity.
2. A supplicant sends the supplicant's identity to the authenticator in an EAP Response message.
3. The authenticator sends the supplicant's identity to the authentication server and sends the authentication server's response to the supplicant in an EAP Request message.
4. The supplicant sends the supplicant's credentials in an EAP Response message to the authenticator.
5. The authenticator sends the supplicant's credentials to the authentication server. If authentication is not successful, the authenticator sends an EAP Failure message to the supplicant.

6. If the authentication is successful, the authenticator sends an EAP Success message to the supplicant.

PARTICIPATION ACTIVITY 2.7.5: IEEE 802.1X.

How to use this tool

EAPOL **IEEE 802.1X** **Supplicant** **Authenticator** **Authentication server**

	A user or a device that wants to authenticate to a network.
	A server that authenticates a supplicant and makes an access control decision.
	A device that acts as a proxy for a supplicant and controls a supplicant's communication with an authentication server.
	The encapsulation of EAP over IEEE 802.1X.
	An IEEE standard for port-based access control.

Reset

PARTICIPATION ACTIVITY 2.7.6: IEEE 802.1X.

1) Which EAP message is sent from an authenticator to a supplicant to request the supplicant's credentials?

- ○ EAP Request
- ○ EAP Response
- ○ EAP Success
- ○ EAP Failure

2) Which EAP message is sent to an authenticator from a supplicant to provide the supplicant's identity to the authenticator?

- EAP Request
- EAP Response
- EAP Success
- EAP Failure

3) Which EAP message is sent from an authenticator to a supplicant if the supplicant's authentication is successful?

- EAP Request
- EAP Response
- EAP Success
- EAP Failure

4) Which EAP message is sent from an authenticator to a supplicant if the supplicant's authentication is not successful?

- EAP Request
- EAP Response
- EAP Success
- EAP Failure

2.8 Authentication protocols: RADIUS and TACACS+

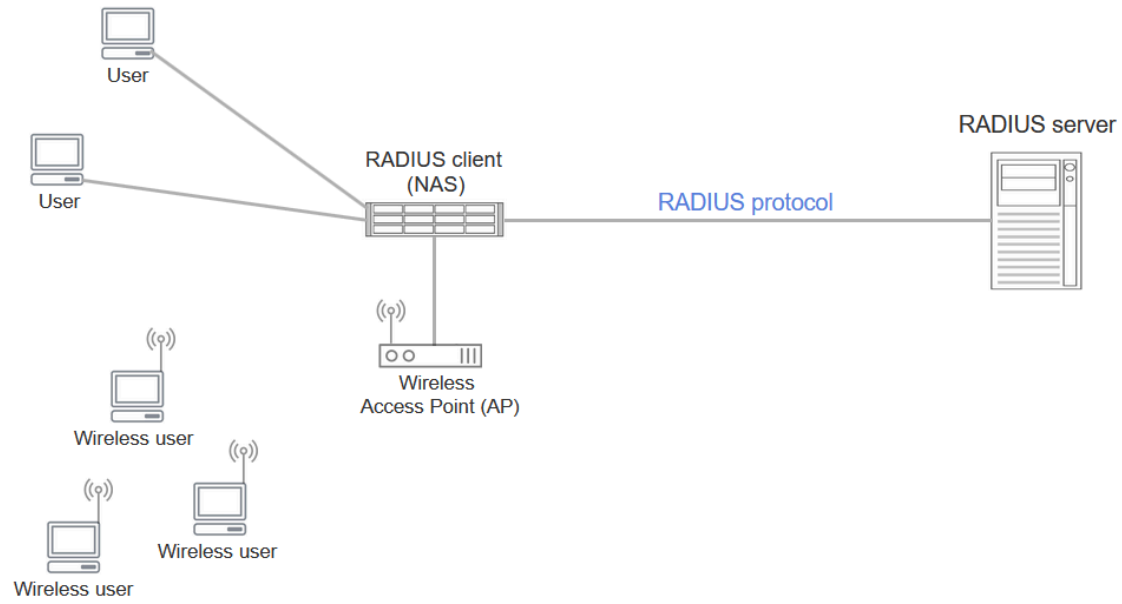
RADIUS

The ***Remote Authentication Dial-In User Service (RADIUS)*** is a networking protocol for centralized authentication, authorization, and accounting (AAA) services. The RADIUS protocol combines authentication and authorization services. RADIUS uses UDP port 1812 for authentication and

authorization, and UDP port 1813 for accounting. Ex: Windows Server 2019 can be configured to use the RADIUS protocol for authenticating and authorizing a user.

The RADIUS protocol is used for exchanging messages between a RADIUS server and a RADIUS client. A RADIUS server provides authentication and authorization services. A RADIUS client is a Network Access Server (NAS) that acts as an intermediary for a connection request from a user to a RADIUS server. A ***network access server (NAS)***, also known as ***remote access server (RAS)***, is an access gateway between an untrusted external network such as the Internet and a trusted internal network.

Figure 2.8.1: RADIUS protocol.



PARTICIPATION ACTIVITY 2.8.1: RADIUS.

1) Which two services are combined in the RADIUS protocol?

- Authentication and accounting
- Authorization and accounting
- Authentication and authorization

2) Which ports are used by the RADIUS protocol?

- TCP 80 and UDP 1813
- UDP 1812 and UDP 1813

- TCP 443 and TCP 22

3) A network access server (NAS) is an access gateway between which types of networks?

- Untrusted external network and untrusted internal network
- Untrusted external network and trusted internal network
- Trusted external network and trusted internal network

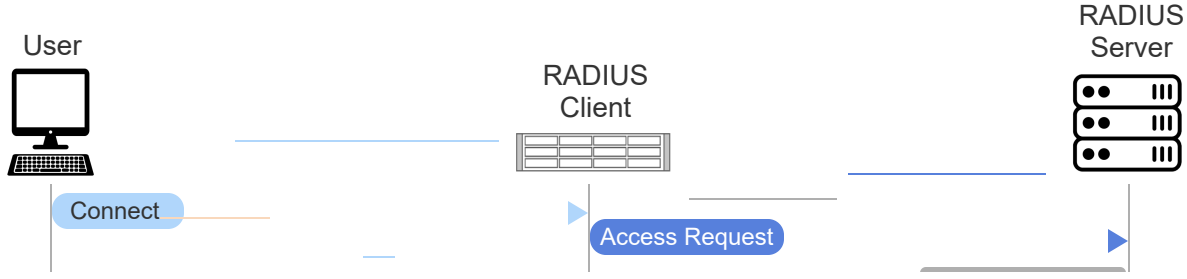
RADIUS authentication

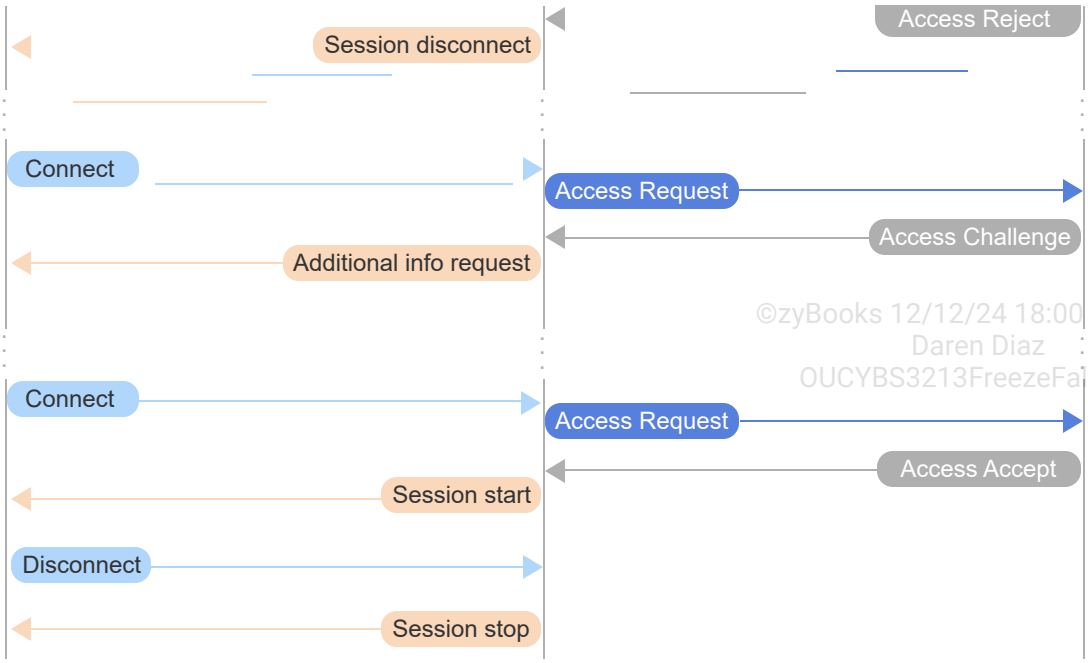
A user that wants to authenticate to a network connects to a RADIUS client. A RADIUS client prompts a user for the user's credentials and sends an *Access Request* message to a RADIUS server. An ***Access Request*** message is an authentication request that contains a user's credentials. A RADIUS server validates a user's credentials using an authentication protocol such as Challenge Handshake Authentication Protocol (CHAP). A RADIUS server responds to an *Access Request* message with one of the following messages:

- ***Access Reject*** is a message sent from a RADIUS server to a RADIUS client if a user's credentials are not valid.
- ***Access Challenge*** is a message sent from a RADIUS server to a RADIUS client if the RADIUS server requires additional information from a user, such as a user's PIN or a user's second password.
- ***Access Accept*** is a message sent from a RADIUS server to a RADIUS client if a user's credentials are valid.

If a user is successfully authenticated by a RADIUS server, the RADIUS server grants the user access to the user's authorized network resources.

PARTICIPATION ACTIVITY 2.8.2: RADIUS protocol.





Animation content:

Static image: An icon labeled "User," an icon labeled "RADIUS Client," and an icon labeled "RADIUS Server." A message labeled "Connect" points from the User to the RADIUS Client. A message labeled "Access Request" points from the RADIUS Client to the RADIUS Server. A message labeled "Access Reject" points from the RADIUS Server to the RADIUS Client. A message labeled "Session disconnect" points from the RADIUS Client to the User. A message labeled "Connect" points from the User to the RADIUS Client. A message labeled "Access Request" points from the RADIUS Client to the RADIUS Server. A message labeled "Access Challenge" points from the RADIUS Server to the RADIUS Client. A message labeled "Additional info request" points from the RADIUS Client to the User. A message labeled "Connect" points from the User to the RADIUS Client. A message labeled "Access Request" points from the RADIUS Client to the RADIUS Server. A message labeled "Access Accept" points from the RADIUS Server to the RADIUS Client. A message labeled "Session start" points from the RADIUS Client to the User. A message labeled "Disconnect" points from the User to the RADIUS Client. A message labeled "Session stop" points from the RADIUS Client to the User.
Step 1: A user connects to the RADIUS client. The RADIUS client sends the user's credentials to the RADIUS server in an "Access Request" message.
An icon labeled "User," an icon labeled "RADIUS Client," and an icon labeled "RADIUS Server." A message labeled "Connect" appears pointing from the User to the RADIUS Client. A message labeled "Access Request" appears pointing from the RADIUS Client to the RADIUS Server.
Step 2: The RADIUS server sends an "Access Reject" message to the RADIUS client if the user's credentials are not valid. The RADIUS client disconnects the user's session.
A message labeled "Access Reject" appears pointing from the RADIUS Server to the RADIUS Client. A message labeled "Session disconnect" appears pointing from the RADIUS Client to the User.
Step 3: The RADIUS server sends an "Access Challenge" message to the RADIUS client if the RADIUS server requires additional information from a user, such as a user's PIN.

The messages are grayed out. A message labeled "Connect" appears pointing from the User to the RADIUS Client. A message labeled "Access Request" appears pointing from the RADIUS Client to the RADIUS Server. A message labeled "Access Challenge" appears pointing from the RADIUS Server to the RADIUS Client. A message labeled "Additional info request" appears pointing from the RADIUS Client to the User.
Step 4: The RADIUS server sends an "Access Accept" message to the RADIUS client if the user's credentials are valid. The RADIUS client starts the user's session.

The previous four messages are also grayed out. A message labeled "Connect" appears pointing from the User to the RADIUS Client. A message labeled "Access Request" appears pointing from the RADIUS Client to the RADIUS Server. A message labeled "Access Accept" appears pointing from the RADIUS Server to the RADIUS Client. A message labeled "Session start" appears pointing from the RADIUS Client to the User.
Step 5: A user sends a disconnect message to the RADIUS client. The RADIUS client stops the user's session.
The four previous messages are also grayed out. A message labeled "Disconnect" appears pointing from the User to the RADIUS Client. A message labeled "Session stop" appears pointing from the RADIUS Client to the User.

Animation captions:

1. A user connects to the RADIUS client. The RADIUS client sends the user's credentials to the RADIUS server in an "Access Request" message.
2. The RADIUS server sends an "Access Reject" message to the RADIUS client if the user's credentials are not valid. The RADIUS client disconnects the user's session.
3. The RADIUS server sends an "Access Challenge" message to the RADIUS client if the RADIUS server requires additional information from a user, such as a user's PIN.
4. The RADIUS server sends an "Access Accept" message to the RADIUS client if the user's credentials are valid. The RADIUS client starts the user's session.
5. A user sends a disconnect message to the RADIUS client. The RADIUS client stops the user's session.

PARTICIPATION ACTIVITY 2.8.3: RADIUS.

How to use this tool

Access Request **Access Reject** **Access Challenge** **Access Accept**

	A message sent from a network access server (NAS) to a RADIUS

	server, requesting a RADIUS server to authenticate a user.
	A message sent from a RADIUS server to a RADIUS client after a RADIUS server authenticates a user.
	A message sent from a RADIUS server to a RADIUS client after a RADIUS server cannot validate a user's credentials.
	A message sent from a RADIUS server to a RADIUS client after a RADIUS server requests additional user information.

Reset

PARTICIPATION ACTIVITY 2.8.4: RADIUS protocol.

1) What is the message that is sent by a RADIUS server to a RADIUS client if a user's credentials are not valid?
   - Access Request
   - Access Accept
   - Access Reject
   - Access Challenge

2) What is the message that is sent by a RADIUS server to a RADIUS client to request additional information from a user such as a user's PIN or a user's second password?
   - Access Request
   - Access Accept
   - Access Reject
   - Access Challenge

3) What is the message that is sent by a RADIUS client to a RADIUS server that

includes a user's credentials?

- Access Request
- Access Accept
- Access Reject
- Access Challenge

4) What is the message that is sent by a RADIUS server to a RADIUS client if a user's credentials are valid?

- Access Request
- Access Accept
- Access Reject
- Access Challenge

TACACS+

Terminal Access Controller Access-Control System Plus (TACACS+) is a proprietary networking protocol developed by CISCO for centralized Authentication, Authorization, and Accounting (AAA) services. TACACS+ separates authentication, authorization, and accounting services. TACACS+ can be used to authenticate a user or a device. TACACS+ uses TCP port 49. Unlike RADIUS that only encrypts a user's password, TACACS+ encrypts the entire content of each AAA packet.

A TACACS+ client is a network access node such as a Network Access Server (NAS). A TACACS+ server contains authentication information for users and devices. A NAS obtains a user or device's credential and sends an authentication request to a TACACS+ server. A TACACS+ server responds with one of the following messages:

- ***Accept*** is a message sent from a TACACS+ server to a TACACS+ client if a user or device credentials are valid.
- ***Reject*** is a message sent from a TACACS+ server to a TACACS+ client if a user or device credentials are not valid.
- ***Error*** is a message sent from a TACACS+ server to a TACACS+ client if the TACACS+ server detects a network error during an authentication process.

TACACS+ is commonly used for authenticating an administrator account to a network device such as a switch or a router.

Table 2.8.1: Comparison of RADIUS and TACACS+ protocols.

	RADIUS	TACACS+
Functionality	Combines authentication and authorization	Separate authentication, authorization, and accounting
Standard	Open/RFC standard	Proprietary (Cisco)
Primary usage	Network access	Device administration
Security	Encryption of passwords only	Encryption of all AAA packets
Multi-factor authentication	Not supported	Supported
Transport protocol	UDP port 1812 for authentication and authorization, and UDP port 1813 for accounting	TCP port 49 for authentication, authorization, and accounting

PARTICIPATION ACTIVITY 2.8.5: TACACS+.

How to use this tool

Accept message **Error message** **Reject message**

	A message sent by a TACACS+ server to a TACACS+ client if a user's credentials are valid.
	The message sent by a TACACS+ server to a TACACS+ client if a user's credentials are not valid.
	The message sent by a TACACS+ server to a TACACS+ client if a TACACS+ server detects a network error during an authentication process.

Reset

PARTICIPATION ACTIVITY 2.8.6: TACACS+.

1) The _____ protocol combines authentication and authorization services.
   - RADIUS
   - TACACS+

2) The _____ protocol is proprietary.
   - RADIUS
   - TACACS+

3) The _____ protocol is commonly used for authenticating an administrator account to a network device.
   - RADIUS
   - TACACS+

4) The _____ protocol only encrypts a user's password.
   - RADIUS
   - TACACS+

5) The _____ protocol uses TCP port 49.
   - RADIUS
   - TACACS+

6) The _____ protocol supports multi-factor authentication.
   - RADIUS
   - TACACS+

2.9 Authentication and authorization on the Internet: SAML, OpenID, and OAuth

Security Assertions Markup Language (SAML)

Security Assertions Markup Language (SAML) is an XML-based standard for exchanging authentication and authorization information. SAML provides single sign-on (SSO) for web-based applications and is commonly used by web portals. ***Single sign-on (SSO)*** is an authentication method that enables a user to log into multiple systems with a single identity without having to authenticate to each system separately.

Three roles are defined in SAML:

- A ***principal*** is a human user.
- An ***identity provider (IdP)*** is an entity that creates, manages, and maintains identity information for a principal.
- A ***service provider (SP)*** is an entity that provides a service to a principal.

PARTICIPATION ACTIVITY 2.9.1: Security Assertions Markup Language (SAML).

1) SAML is used for exchanging _____ information.
   - ○ authentication
   - ○ authentication and authorization

2) An _____ is an entity that creates, manages, and maintains identity information for a principal.
   - ○ IdP
   - ○ SP

3) An _____ is an entity that provides a service to a principal.
   - ○ IdP
   - ○ SP

4) SSO is used for _____.
   - ○ authentication

- authorization

SAML assertion statements

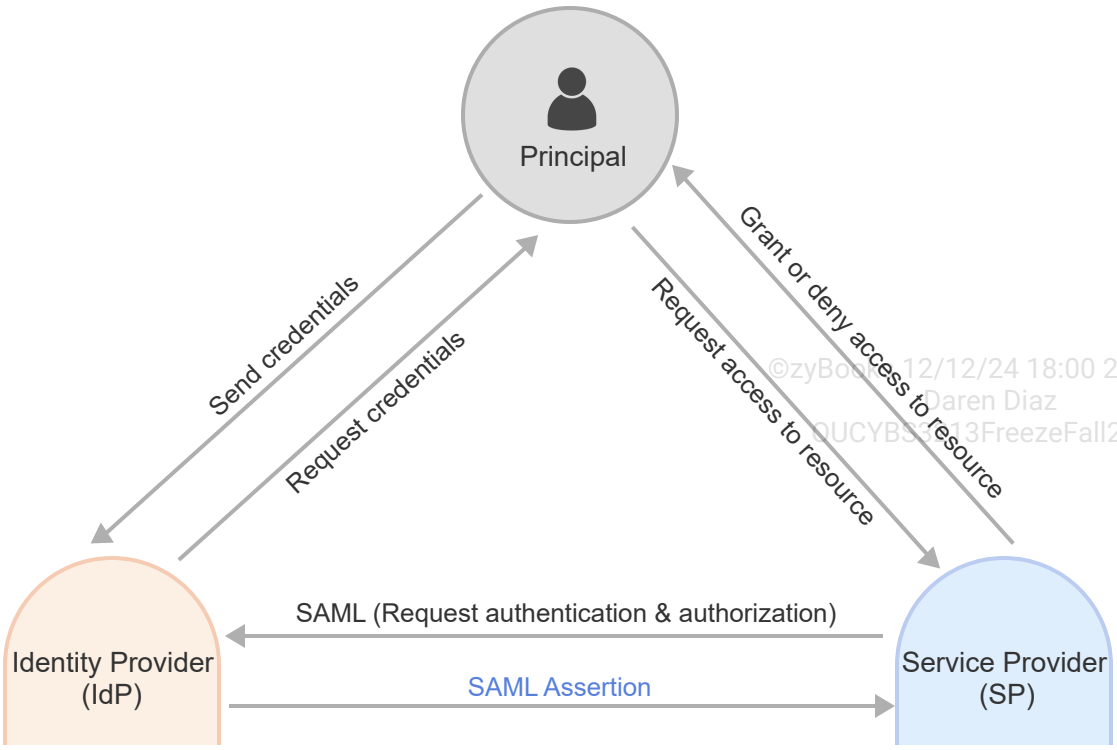
SAML is used between an IdP and an SP. When a principal requests an SP to perform a service, the SP requests an IdP for the principal's authentication and authorization information. In response to the SP's request, an IdP sends a SAML assertion to the SP. A ***SAML assertion*** is a digitally signed XML document that contains statements about a principal.

Three types of SAML assertion statements exist:

- An ***authentication statement*** asserts that a principal authenticated at a specific time using a specific authentication method.
- An ***attribute statement*** asserts that a principal is associated with a specific attribute. An attribute is a name-value pair.
- An ***authorization statement*** asserts that a principal is permitted to perform a specific action on a specific resource.

An SP makes an access control decision (decides whether to perform a principal's requested service) based on an IdP's assertion statements about a principal.

PARTICIPATION ACTIVITY 2.9.2: Security Assertions Markup Language (SAML).



Animation content:

Static image: A gray circle labeled "Principal," an orange circle labeled "Identity Provider (IdP)," and a blue circle labeled "Service Provider (SP)." An arrow labeled "Request access to resource" goes from the Principal to the Service Provider. An arrow labeled "SAML (Request authentication & authorization)" goes from the Service Provider to the Identity Provider. An arrow labeled "Request credentials" goes from the Identity Provider to the Principal. An arrow labeled "Send credentials" goes from the Principal to the Identity Provider. An arrow labeled "SAML Assertion" goes from the Identity Provider to the Service Provider. An arrow labeled "Grant or deny access to resource" goes from the Service Provider to the Principal.

Animation captions:

1. A principal (user) requests access to a resource from a Service Provider (SP).
2. The Service Provider (SP) requests the Identity Provider (IdP) for the principal's authentication and authorization information.
3. The Identity Provider (IdP) requests the principal's credentials.
4. The principal sends the principal's credentials to the Identity Provider (IdP).
5. The Identity Provider (IdP) sends a SAML Assertion about the principal's authentication and authorization to the Service Provider (SP).
6. Based on the Identity Provider's SAML Assertion, the Service Provider grants or denies the principal's access request to a resource.

PARTICIPATION ACTIVITY 2.9.3: SAML.

How to use this tool

SAML attribute statement **SAML authorization statement** **SP**

SAML authentication statement **IdP**

	An entity that creates, manages, and maintains identity information for a principal

	An entity that provides a service to a principal
	A SAML statement that asserts that a principal authenticated at a specific time using a specific authentication method
	A SAML statement that asserts a principal is associated with a specific attribute
	A SAML statement that asserts a principal is permitted to perform a specific action on a specific resource.

Reset

PARTICIPATION ACTIVITY 2.9.4: Security Assertions Markup Language (SAML).

1) An _____ requests for authentication and authorization information on a principal.
   - IdP
   - SP

2) An _____ sends an SAML assertion about a principal.
   - IdP
   - SP

3) An _____ uses SAML assertions to make access control decisions about a principal.
   - IdP
   - SP

4) An _____ requests for a principal's credentials.
   - IdP
   - SP

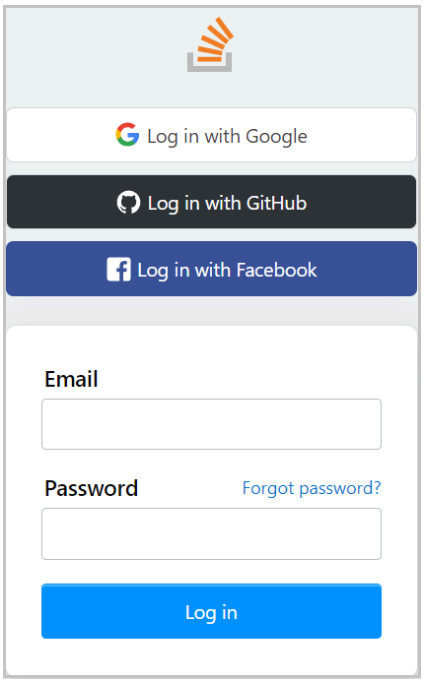
OpenID

OpenID is an open-standard protocol for decentralized authentication. OpenID enables a user to log into multiple websites without the need to have login credentials for each website. OpenID allows a client (a website or an application) to verify the identity of a user without having to manage the user's credentials. OpenID does not rely on a central authority for user authentication.

An ***OpenID Identity Provider (OpenID IdP)*** is a website that manages a user's identity information. Ex: Google and Facebook are OpenID IdPs. An ***OpenID acceptor***, or ***relying party (RP)***, is a website that accepts OpenID authentication. A user creates an account with an OpenID IdP and uses the account to sign into an OpenID acceptor. An OpenID acceptor redirects the user's authentication request to the OpenID IdP that manages the user's identity information. The OpenID IdP confirms the user's identity to the OpenID acceptor.

OpenID does not mandate a specific authentication method by which an OpenID IdP can authenticate a user. An OpenID IdP can use any authentication method, including legacy passwords, smart cards, or biometrics.

Example 2.9.1: Using an OpenID IdP to log into a website.



Credit: stackoverflow.com[1]

PARTICIPATION ACTIVITY 2.9.5: OpenID.

How to use this tool

OpenID **OpenID IdP** **OpenID acceptor**

	A website that manages a user's identity information.
	A website that accepts OpenID authentication.
	An open-standard protocol for decentralized authentication.

Reset

PARTICIPATION ACTIVITY 2.9.6: OpenID.

1) A user creates an account with an OpenID _____.
   - IdP
   - acceptor or relying party

2) An OpenID _____ is a website that manages a user's identity information.
   - IdP
   - acceptor or relying party

3) An OpenID _____ redirects a user's authentication request.
   - IdP
   - acceptor or relying party

4) An OpenID _____ confirms a user's identity.
   - IdP
   - acceptor or relying party

5) An OpenID _____ is a website that accepts OpenID authentication.

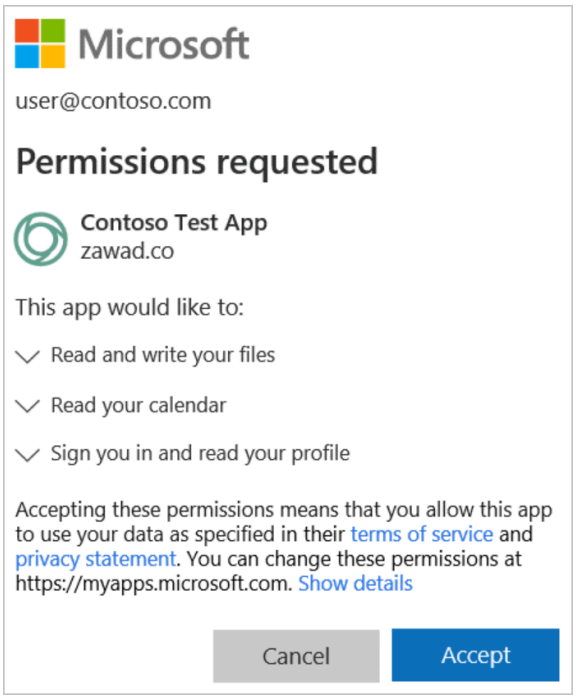
- IdP
- acceptor or relying party

OAuth

OAuth is an open-standard authorization protocol. OAuth enables a user to grant a client (a website or an application) access to the user's information at other websites without sharing the user's credentials with a client. OAuth is designed for use with HTTP.

OAuth provides a client secure delegated access to a user's information on behalf of the user. An authorization server issues access tokens to a client with the approval of the user who owns a resource. A client uses the access token to access a user's resource hosted by a resource server. Ex: Google and Microsoft use OAuth to allow a user to share information about the user's account with a third-party application or a website.

Example 2.9.2: An app requesting permissions using OAuth.



Credit: Microsoft Corporation.[2]

PARTICIPATION ACTIVITY 2.9.7: OAuth.

1) OAuth is an open-standard for authentication.

- False
- True

2) OAuth enables a user to grant an application access to the user's information at other websites without sharing the user's credentials with the application.

- False
- True

3) An OAuth authorization server issues access tokens to a user.

- False
- True

Identity federation

*An **identity federation** is a trust relationship established to exchange authentication information, allowing a user to have a single login across multiple systems. Ex: Identity federation exists between Google and Slack, a business messaging application, so a user can log into Slack with the user's Google credentials.*

***Interoperability** refers to the ability of different systems and organizations to communicate and collaborate. Interoperability enhances identity federation by ensuring that user credentials and security protocols are consistent across platforms, thereby improving operational efficiency and security. Ex: Interoperability allows healthcare providers to use a single set of login credentials across multiple hospital and clinic systems, streamlining secure access.*

(*1) stackoverflow.com "OpenID". https://stackoverflow.com/users/login

(*2) Microsoft Corporation. "OAuth". https://docs.microsoft.com/en-us/cloud-app-security/investigate-risky-oauth

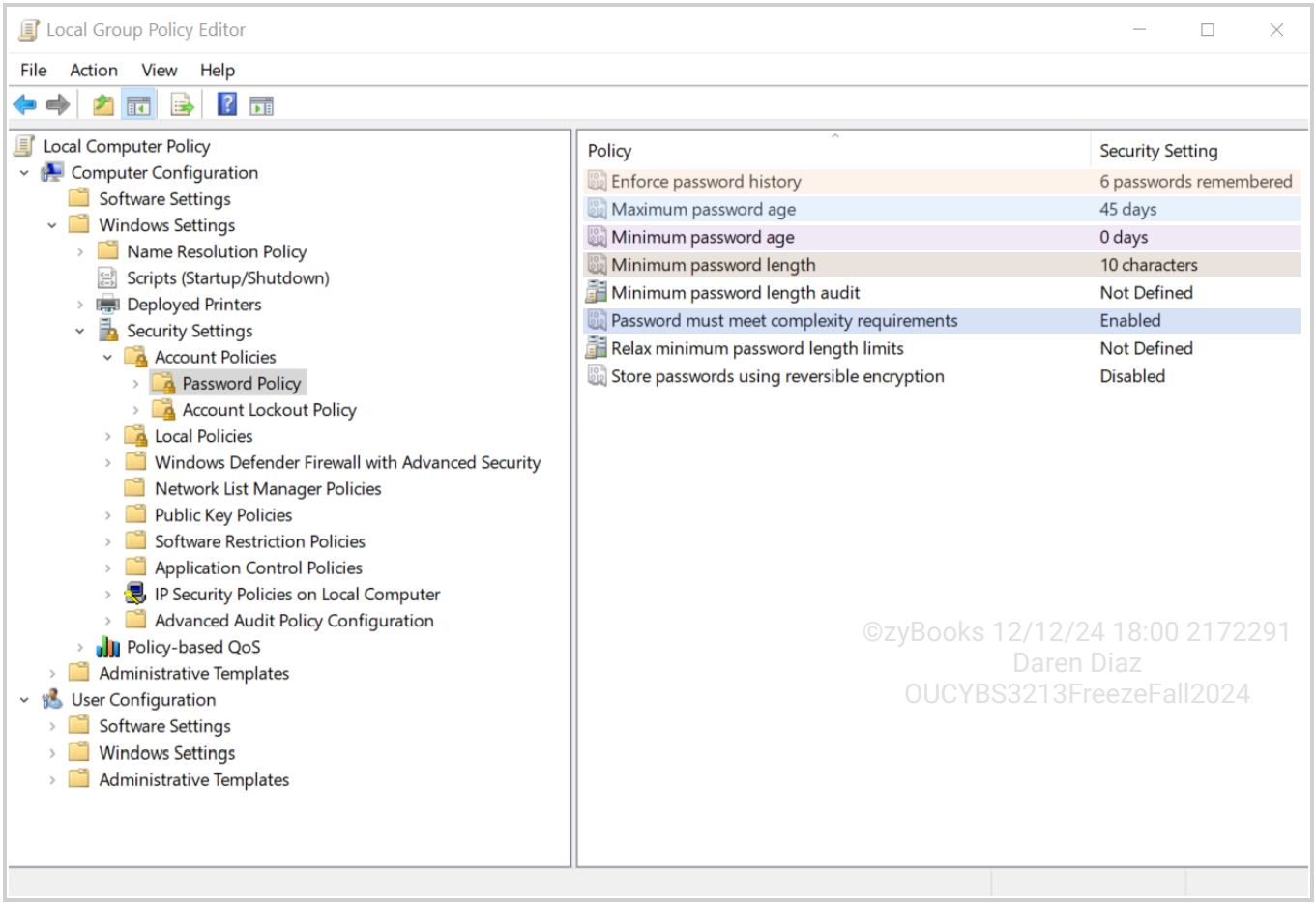
2.10 Accounts: Types and policies

Account policies

An ***account policy*** defines how a computer account should be created, used, maintained, disabled, and removed. An account is protected by a password. A ***password policy*** is a set of rules that define a password's properties. Three common password policies exist:

- ***Password complexity*** policy defines the requirements for a password's length and character types. A password complexity policy prevents a brute-force attack against a password.
- ***Password history*** policy defines how often an old password can be reused. A password history policy prevents password reuse. ***Password reuse*** is the act of reusing a previously used password.
- ***Password age*** policy, also known as ***password life-span*** policy, defines how often a password should be changed. A password age policy limits the time a compromised password can be used to access an account.

PARTICIPATION ACTIVITY 2.10.1: Password policy settings in Windows 10.



Animation content:

Static image: A screenshot of the Local Group Policy Editor in Windows 10. The left pane shows the files. The Password Policy file is highlighted and found under Computer Configuration, then Windows Settings, then Security Settings, then Account Policies. The Password Policy folder is opened in the right pane showing a list of policies and security settings. The policy "Enforce password history" is highlighted in orange and shows "6 passwords remembered" for the security setting. The policy "Maximum password age" is highlighted in blue and shows "45 days" for the security setting. The policy "Minimum password age" is highlighted in pink and shows "0 days" for the security setting. The policy "Minimum password length" is highlighted in brown and shows "10 characters" for the security setting. The policy "Password must meet complexity requirements" is highlighted in purple and shows "Enabled" for the security setting.

Animation captions:

1. "Enforce password history" defines how often an old password can be reused.
2. "Maximum password age" defines how often a password has to be changed.
3. "Minimum password age" defines how long a user must keep a password before a user can change a user's password.
4. "Minimum password length" defines the minimum number of characters in a password.
5. "Password must meet complexity requirements" requires a user to set a complex password that has a minimum length and different character types.

Authentication credentials such as passwords should be securely stored and maintained. A ***password vault***, also known as a ***password manager***, is a software application that stores, manages, and secures a user's passwords. A password vault enables a user to select and store complex passwords without having to memorize those passwords. Ex: 1Password and LastPass are popular password managers.

PARTICIPATION ACTIVITY 2.10.2: Account policies.

How to use this tool

Password complexity policy **Password age policy** **Account policy**

Password history policy

	A password policy that defines how often a password should be changed.
	A policy that defines how a computer account should be created, maintained, and removed from a system.
	A password policy that defines how often an old password can be reused.
	A password policy that defines the requirements for a password's length and character types.

Reset

PARTICIPATION ACTIVITY 2.10.3: Account policies.

1) A password _____ policy may require a password to be changed every 21 days.
   - complexity
   - history
   - age

2) A password _____ policy may require that a password include one lowercase letter.
   - complexity
   - history
   - age

3) A password _____ policy may require the length of a password to be a minimum of eight characters.
   - complexity
   - history
   - age

4) A password _____ policy may require that a password not be one of the four previously used passwords.

- complexity
- history
- age

5) A password _____ policy may require the length of a password to be a minimum of ten characters and include three uppercase letters.

- complexity
- history
- age

Account types

A computer user is associated with a computer account. A computer account contains information about a user and a user's permissions to a computer resource. Five types of accounts exist:

- A ***user account***, or ***end user account***, is an account that is assigned to an individual that wants to access a computer resource.
- A ***privileged account***, also known as an ***administrative account*** or ***root account***, is an account assigned to a system administrator. A privileged account has full authorization and control over a system. A privileged account is used for administering and maintaining a system's resources and users. Ex: The *root* account in Linux/Unix and the *administrator* account in Windows are privileged accounts.
- A ***shared account***, also known as a ***generic*** or ***group account/credentials***, is an account accessed by more than one user. A shared account is used by a group of users that want to access the same computer resources. A shared account's credentials are shared by all the users in a group.
- A ***guest account*** is an account that is assigned to a temporary user. A guest account has limited privileges.
- A ***service account*** is an account that is assigned to an application or service. A service account is not used for interactive login.

PARTICIPATION ACTIVITY 2.10.4: Account types.

How to use this tool

Service account | Shared account | Guest account | User account | Privileged account

	An account that is accessed by more than one user.
	An account that is assigned to a system administrator.
	An account that is assigned to an individual that wants to access a computer resource.
	An account that is assigned to a temporary user.
	An account that is assigned to an application or service.

Reset

PARTICIPATION ACTIVITY 2.10.5: Account types.

1) Which account type is assigned to a system administrator?

- user
- privileged
- shared
- guest
- service

2) Which account type is assigned to a database server?

- user
- privileged
- shared

- guest
- service

3) Which account type is assigned to a group of users that want to access the same computer resources?

- user
- privileged
- shared
- guest
- service

4) Which account type is assigned to an individual that wants to access a computer resource?

- user
- privileged
- shared
- guest
- service

5) Which account type is assigned to a temporary user?

- user
- privileged
- shared
- guest
- service

2.11 Accounts: Controls and maintenance

Account controls

Access to a user account can be controlled based on the physical characteristics of a user's device. One such physical characteristic is a device's geolocation. ***Geolocation*** is the use of location

technologies such as a GPS coordinate or an IP address to identify the location of an electronic device. ***Geotagging*** is the process of adding geographical identification to a device. Geotagging can be used to allow a device to access a system only if a device is at a specific GPS coordinate or has a specific IP address.

Geofencing is the practice of using GPS or radio frequency identification (RFID) to define a virtual perimeter for a geographic boundary. A geofenced area is a predetermined GPS-based area from which a user can login into a user's account.

PARTICIPATION ACTIVITY 2.11.1: Account controls.

1) A device can be geotagged by adding geographical identification to a device.
   - False
   - True

2) A GPS coordinate and an IP address can be used in geotagging.
   - False
   - True

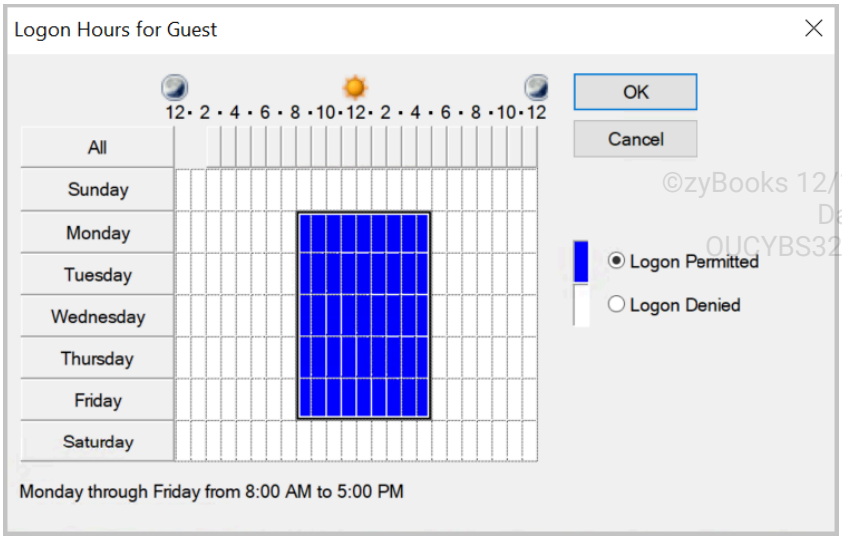
3) Geofencing is the practice of using an IP address to define a virtual perimeter for a geographic boundary.
   - False
   - True

A device's geolocation can be used to deny a user's login attempt from a new location. ***Impossible travel time*** is a metric that a system can use to deny a user's login attempt from a location that a user could not have travelled to in the elapsed time since a user's previous login location. Ex: A user's login attempt from Los Angeles at 9:05 AM is denied if the user's previous login was at 9:00 AM from Boston.

A device's ***network location*** is a device's connection point to a network. A network location can be determined by a device's media access control (MAC) address or IP address. A device's network location can be used to control a user's account access. Ex: A user account may be configured to only be accessible from an IP address assigned to a user's laptop.

Time of day restrictions specify when a user can login to a user's account. A time of day restriction is used to prevent a user from accessing a system outside work hours. Ex: A user may only be allowed to login to a user's account on Monday through Friday from 8:00 AM to 5:00 PM. ***Time-based login*** enables a user to login to a user's account at a specific time.

Figure 2.11.1: Time of day settings in Windows Server 2019.



PARTICIPATION ACTIVITY 2.11.2: Account controls.

How to use this tool

Geolocation **Geofencing** **Geotagging** **Time-based login**

Impossible travel time

	The use of location technologies such as a GPS coordinate or an IP address to identify the location of an electronic device.
	The practice of using GPS or radio frequency identification (RFID) to define a virtual perimeter for a geographic boundary.
	The process of adding geographical identification to a device.
	An account control mechanism that enables a user to login to a user's

	account at a specific time.
	A metric that a system can use to deny a user's login attempt from a location that a user could not have travelled to in the elapsed time since a user's previous login location.

Reset

PARTICIPATION ACTIVITY 2.11.3: Account controls.

1) Which account control can be used to deny a user login attempt based on a user's distance from the user's previous login location?
   - Impossible travel time
   - Time of day
   - network location

2) Which account control can be used to only allow a user to login to a system on weekends?
   - lockout
   - Time of day
   - network location

3) Which account control can be used to only allow a user to login to a system from the user's mobile phone?
   - Impossible travel time
   - Time of day
   - network location

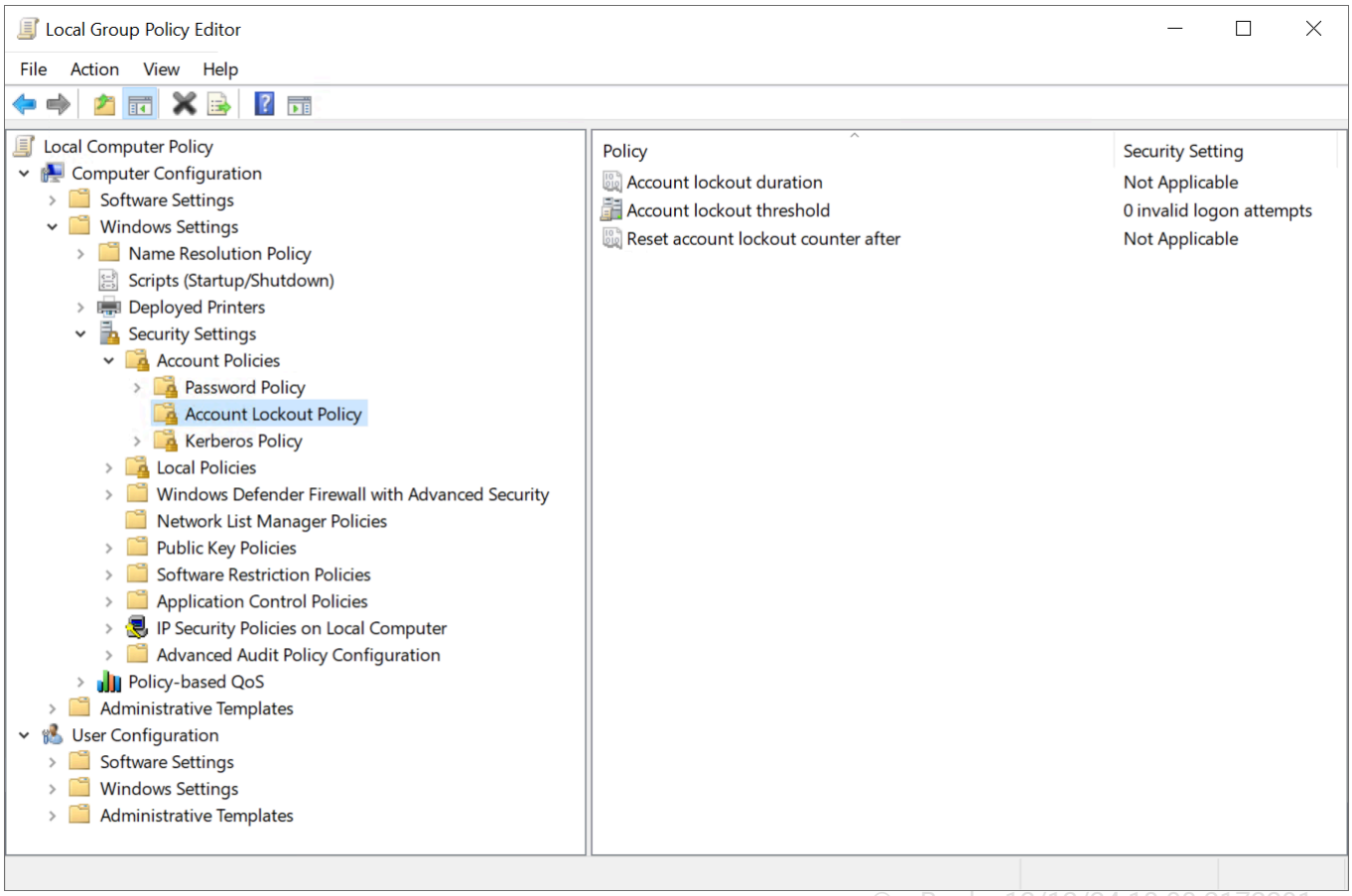
Account maintenance

Account controls are necessary but not sufficient to provide security for a system. An ***account audit*** establishes whether account controls are properly implemented and are effective in controlling access to a system. An account audit helps ensure that a user has access rights based on the user's needs

and a system's security policies. An account audit is performed periodically and can be manual or automatic.

An account that is temporarily not needed or is not secure can be disabled. A ***disabled account*** is an account that exists in a system but not accessible by the account owner. A user account can also be automatically locked out to prevent or slow down a brute-force attack against a user's credentials. An ***account lockout*** is the disabling of an account when the number of incorrect login attempts to an account exceeds a predefined number. A lockout account can be manually unlocked by an account administrator or automatically unlocked after a period of time, known as a ***lockout duration***.

PARTICIPATION ACTIVITY 2.11.4: Account lockout policy settings in Windows Group Policy.



Animation content:

Static image: A screenshot of the Windows Local Group Policy Editor. The left pane shows the files. The Account Lockout Policy file is highlighted and found under Computer Configuration, then Windows Settings, then Security Settings, then Account Policies. The Account Lockout Policy folder is opened in the right pane showing a list of policies and security settings. The policy "Account

lockout duration" shows "Not Applicable" for the security setting. The policy "Account lockout threshold" shows "0 invalid logon attempts" for the security setting. The policy "Reset account lockout counter after" shows "Not Applicable" for the security setting.

Animation captions:

1. A Windows environment uses the Group Policy Editor to configure Group Policy Objects (GPOs) for account maintenance and device settings.
2. "Account lockout duration" defines the number of minutes a locked out account remains locked out before automatically becoming unlocked.
3. "Account lockout threshold" defines the number of failed login attempts that causes a user account to be locked out.
4. "Reset account lockout counter after" defines the number of minutes that must pass after a failed logon attempt before the failed logon attempt counter is set to zero.

PARTICIPATION ACTIVITY 2.11.5: Account maintenance.

How to use this tool

Lockout duration | **Account lockout** | **Disabled account** | **Account audit**

	Establishes whether account controls are properly implemented and are effective in controlling access to a system.
	An account that exists in a system but not accessible by the account owner.
	The disabling of an account when the number of incorrect login attempts to an account exceeds a predefined number.
	The period of time that a lockout account cannot be accessed.

Reset

PARTICIPATION ACTIVITY 2.11.6: Account maintenance.

1) A disabled account can be the result of a security breach.
   - False
   - True

2) An account audit can be used to ensure that a system's security policies are followed.
   - False
   - True

3) A brute-force attack against a user's credentials cannot result in the user's account lockout.
   - False
   - True

Attestation

Attestation *is the process of verifying a device's hardware and software configurations to ensure trustworthiness. Attestation enhances account security by ensuring only compliant devices access network resources. Attestation can integrate with geofencing and time-based restrictions, adding a security layer to prevent potentially compromised devices from accessing sensitive data. Ex: Before allowing a laptop to access an organization's network, attestation verifies that the laptop's antivirus software is up-to-date and the laptop's firewall is enabled prior to granting network access.*

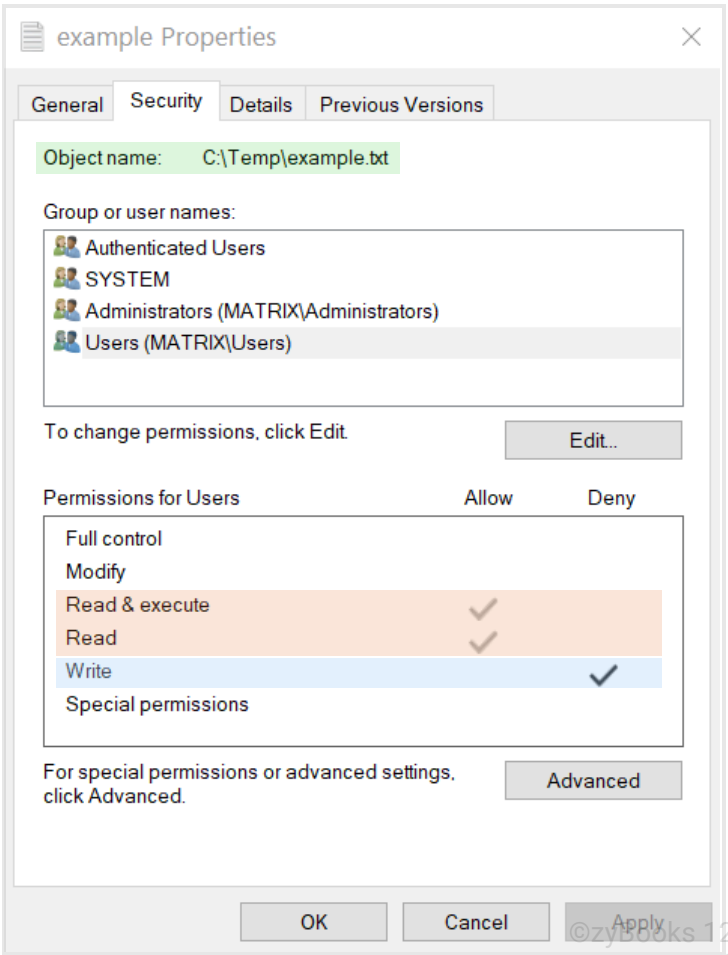
2.12 Access control models: DAC and MAC

Discretionary access control (DAC)

Discretionary access control (DAC) is an access control model in which access to an object is at the discretion or choice of the object's owner. An object's owner defines access to the object based on a subject's need for accessing the object. A DAC model is implemented using an access control list (ACL). An ***access control list (ACL)*** defines a subject's access type to an object. Ex: In DAC, a user's access to a file is at the discretion of the file's owner. A file's owner can grant a user *write permission* to the file or deny a user *write permission* to the file.

DAC is the access control model used by default in most operating systems including Linux/Unix and Windows.

PARTICIPATION ACTIVITY 2.12.1: Discretionary access control (DAC) in Windows 10.



Animation content:

Static image: A screenshot of an "example Properties" window in Windows 10. The Security tab is shown. The first line is highlighted in green and says "Object name: C:\Temp\example.txt". A box labeled "Group or user names" contains "Authenticated Users", "SYSTEM", "Administrators", and "Users". "Users" is highlighted in gray. A box labeled "Permissions for Users" has "Allow" and "Deny"

columns. The "Read & execute" and "Read" permissions are highlighted in orange and have check marks under "Allow." The "Write" permission is highlighted in blue and has a check mark under "Deny."

Animation captions:

1. The "Object name:" specifies a resource named "example.txt" (the object).
2. The users (the subjects) have "Read & execute" and "Read" permissions on the "example.txt" file (the object).
3. The users (the subjects) do not have "Write" permissions on the "example.txt" file (the object).

PARTICIPATION ACTIVITY 2.12.2: Discretionary access control (DAC).

How to use this tool

Discretionary access control (DAC) **Subject** **Access control list (ACL)**

Access control model **Object**

	Defines a subject's access type to an object
	An access control model in which access to an object is at the discretion or choice of an object's owner
	A resource that a user or a process wants to access
	A user or process that wants to access a resource
	A set of technology-independent rules for controlling access of subjects to objects

Reset

PARTICIPATION ACTIVITY 2.12.3: Discretionary access control (DAC).

1) In DAC, a database application's owner is a(n) _____.
   - object
   - subject

2) In DAC, a printer is a(n) _____.
   - object
   - subject

3) In DAC, a user's access type to a resource is defined by a(n) _____.
   - object
   - subject

4) In DAC, an ACL defines a user's access type to a(n) _____.
   - object
   - subject

Mandatory access control (MAC)

Mandatory access control (MAC) is an access control model in which access to an object is mandated by a set of rules and classification labels. A classification label represents a security domain. A ***security domain***, also known as a ***realm of security***, is a collection of subjects and objects that share a common security policy.

A subject is assigned a clearance level. An object is assigned a classification label. A ***MAC security policy*** defines a subject's access privileges to an object based on a subject's clearance level and an object's classification label. A central authority assigns clearance levels, classification labels, and defines security policies. Ex: A user is assigned *"secret"* clearance level. A file is assigned *"top secret"* classification label. A MAC security policy denies a user with *"secret"* clearance to access a file with *"top secret"* classification.

MAC is the most restrictive access control model and is typically used in government and military settings where data confidentiality is a high priority.

PARTICIPATION ACTIVITY 2.12.4: Mandatory access control (MAC).

How to use this tool

Mandatory access control (MAC) | **Realm of security** | **MAC security policy**

	An access control model in which access to an object is mandated by a set of rules and classification labels
	A collection of subjects and objects that share a common security policy
	Defines a subject's access privileges to an object based on a subject's clearance level and an object's classification label

Reset

PARTICIPATION ACTIVITY 2.12.5: Mandatory Access Control (MAC).

1) In MAC, an object's owner controls access to an object.
   - False
   - True

2) In MAC, a central authority assigns clearance levels, classification labels, and defines security policies.
   - False
   - True

3) In MAC, access to an object is mandated by a set of rules and classification labels assigned by a subject.
   - False
   - True

4) In MAC, a subject is assigned a classification label.

- False
- True

5) In MAC, an object is assigned a classification label.

- False
- True

SELinux

To increase security for confidential data on Linux machines, the US National Security Agency (NSA) developed SELinux. ***SELinux*** *is a security architecture that supports MAC implementation on Linux. SELinux was released to the open source community in 2000 and is now included in the Linux kernel. SELinux is most commonly used on Red Hat Enhanced Linux (RHEL) and Fedora Linux. SELinux has provided Android's MAC system since Android 4.3, released in 2013.*

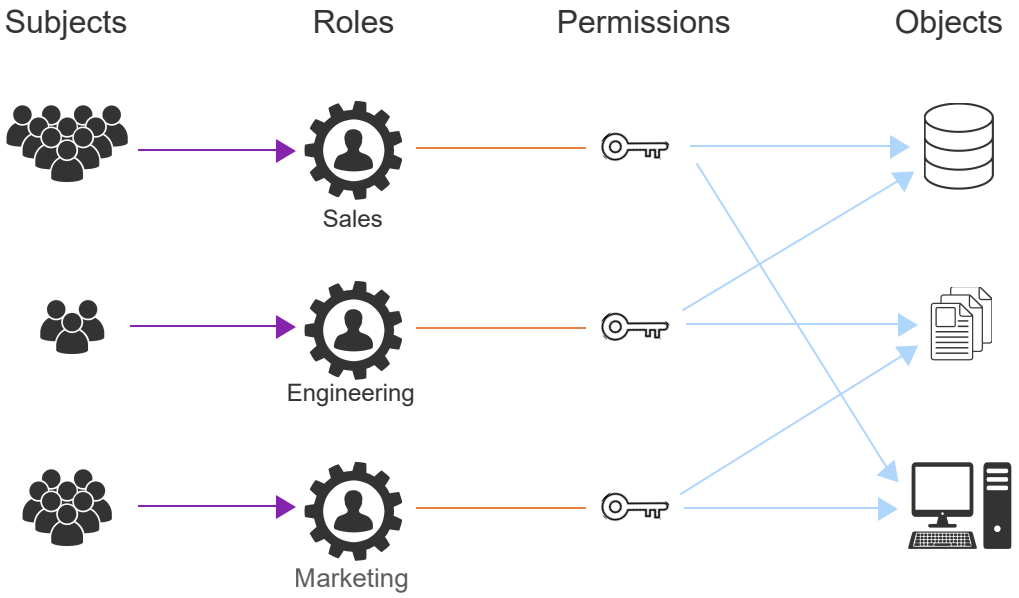
2.13 Access control models: Role, rule, and attribute-based

Role-based access control (RBAC)

Role-based access control (RBAC) is an access control model in which a subject's access rights to an object are based on the subject's role within a system. An ***RBAC role*** is a collection of objects and permissions that capture the requirements of a job function. A subject is assigned to a role, and thereby gains access rights to objects that are required to perform the subject's job. A subject can access an object only if a subject is assigned to a role. Ex: A sales role captures the requirements for a sales person, which may include access rights to a database and a computer. A user who is a sales person is assigned to the sales role, and thereby gains access to the database and the computer.

RBAC is a non-discretionary access control model. A ***non-discretionary access control model*** is an access control model in which access to an object is controlled by a central authority or administrator.

PARTICIPATION ACTIVITY 2.13.1: Role-based access control (RBAC).



Animation content:

Static image: A table with columns "Subjects," "Roles," "Permissions," and "Objects." The first row, Subjects column has an icon of ten people. The icon has an arrow pointing to an icon labeled "Sales" in the first row, Roles column. An orange line connects the Sales icon with a key icon in the first row, Permissions column. Two arrows are pointing from the key. One arrow points to a database icon in the first row, Objects column. The other arrow points to a desktop icon in the third row, Objects column. The second row, Subjects column has an icon of three people. The icon has an arrow pointing to an icon labeled "Engineering" in the second row, Roles column. An orange line connects the Engineering icon with a key icon in the second row, Permissions column. Two arrows are pointing from the key. One arrow points to the database icon in the first row, Objects column. The other arrow points to a file icon in the second row, Objects column. The third row, Subjects column has an icon of eight people. The icon has an arrow pointing to an icon labeled "Marketing" in the third row, Roles column. An orange line connects the Marketing icon with a key icon in the third row, Permissions column. Two arrows are pointing from the key. The first arrow points to the file icon in the second row, Objects column. The other arrow points to the desktop icon in the third row, Objects column.

Animation captions:

1. A role is a collection of objects and permissions that capture the requirements of a job function.
2. Subjects (users) are assigned to a role, and thereby gain access to objects.

PARTICIPATION ACTIVITY 2.13.2: Role-based access control (RBAC).

How to use this tool

An RBAC role **Non-discretionary access control model**

Role-based access control (RBAC)

	An access control model in which access to an object is based on a subject's role within a system
	A collection of objects and permissions that capture the requirements of a job function.
	An access control model in which access to objects is controlled by a central authority or administrator, not an object's owner

Reset

PARTICIPATION ACTIVITY 2.13.3: Role-based access control (RBAC).

1) In RBAC, who controls access to an object?

- ○ An object's owner
- ○ A subject that has access rights to an object

- A central authority or administrator

2) What is an RBAC role?

- A collection of subjects and objects that capture the requirements of a job function.
- A collection of objects and permissions that captures the requirements of a job function.
- A collection of subjects, objects, and permissions that captures the requirements for a job function.

3) In RBAC, how does a subject gain access rights to an object?

- By being assigned access rights to the object by a central authority.
- By requesting access rights from the object's owner.
- By being assigned to a role that requires access rights to the object.

Attribute-based access control (ABAC)

An ***attribute-based access control (ABAC)*** model is an access control model in which access to an object is based on attributes and access control policies that define the allowable operations for a given attribute combination. ABAC is a non-discretionary access control model. ABAC is also known as ***policy-based access control (PBAC)***.

ABAC uses three types of attributes:

- ***Object attribute*** is a characteristic of an object such as name, creation date and time, sensitivity, and owner.
- ***Subject attribute*** is a characteristic of a subject such as name, role, organization, and security clearance.
- ***Environment attribute***, also known as the ***context of an access request***, is a characteristic of an access request environment such as time-of-access and current threat levels.

In ABAC, a subject is granted access rights to an object when a *"subject-object-environment"* attribute combination matches a defined access control policy. Ex: A user's access to a file is denied if the attribute combination of *"user's security clearance-file's creation time-current threat levels"* does not match a defined access control policy.

PARTICIPATION ACTIVITY 2.13.4: Attribute-based access control (ABAC).

How to use this tool

Attribute-based access control (ABAC) **Environment attribute** **Subject attribute** **Object attribute**

	An access control model in which a subject's access to an object is based on attributes and access control policies.
	A characteristic of an entity that may include name, creation date and time, sensitivity, and owner.
	A characteristic of an access request environment.
	A characteristic of an entity that may include name, role, organization, and security clearance.

Reset

PARTICIPATION ACTIVITY 2.13.5: Attribute-based access control (ABAC).

1) In ABAC, how is access to an object controlled?

- ○ An object's owner controls access to the object.

- A subject controls access to an object.
- Attributes and access control policies control access to an object.

2) In ABAC, which attribute type may be used to deny a user's access to a file based on a user's security clearance?
   - Object attribute
   - Subject attribute
   - Environment attribute

3) In ABAC, which attribute type may be used to deny a user's access to a file based on a current threat level?
   - Object attribute
   - Subject attribute
   - Environment attribute

4) In ABAC, which attribute type may be used to deny a user's access to a file based on a file's creation time?
   - Object attribute
   - Subject attribute
   - Environment attribute

Rule-based access control

A ***rule-based access control*** model is an access control model in which access to an object is based on a set of rules. A subject can only access an object if all the conditions specified in a rule are met. A rule defines circumstances in which a subject can access an object. Ex: A rule may allow access to an object from a specific IP address, restrict access to an object to business hours only, or deny access to an object from a mobile device. Rule-based access control is a non-discretionary access control model. A central authority or administrator defines a set of rules for a system.

Rule-based access control is commonly implemented in firewalls through the use of access control lists (ACLs). A firewall uses a set of rules to filter network traffic. Ex: A firewall rule may allow HTTPS traffic on TCP port 443 or deny SSH traffic on TCP port 22.

PARTICIPATION ACTIVITY 2.13.6: Rule-based access control.

1) In rule-based access control, who controls access to an object?

- An object's owner
- A subject that has access rights to an object
- A central authority or administrator

2) In rule-based access control, how does a subject gain access rights to an object?

- By requesting access rights from the object's owner.
- By meeting all the conditions defined in a rule.
- By being assigned to a role that requires access rights to the object.

3) In rule-based access control, how can a subject be restricted to access an object on weekends only?

- An object's owner grants the subject access to the object on weekends only.
- An object's owner would define a rule that specifies when a subject can access the object.
- A central authority would define a rule that specifies the times that a subject can access the object.

Table 2.13.1: Access control models.

Access control model	Permissioning basis	Authorization criteria	Non-discretionary access control

DAC	Object owner	Object owner	No
MAC	Classification of objects and subjects	Security policies	Yes
Role-based	Classification of roles	Roles	Yes
Rule-based	Evaluation of rules	Rules	Yes
Attribute-based	Evaluation of attributes	Attributes	Yes

2.14 Access control: Filesystem permissions, privileged access management, and conditional access

Filesystem permissions

Filesystem permissions control which users, groups, or services can perform an action on a file. An action on a file is reading, writing, modifying, or executing the file. Improper or insecure file permissions may be leveraged by a user to perform an unauthorized action on a file.

Filesystem permissions are implemented differently in each operating system. Ex: In Linux/Unix a file has read, write, and execute permissions. A file permission is denoted by a set of three characters: `r` for read, `w` for write, and `x` for execute. If an action is not permitted on a file, the corresponding letter is replaced by `-`. Ex: A file permission of `r-x` means that a user can read and execute the file, but not write to the file. Ex: A file permission of `rw-` means that a user can read and write to the file, but not execute the file.

Table 2.14.1: File permissions in Linux/Unix.

Symbolic representation	Permission

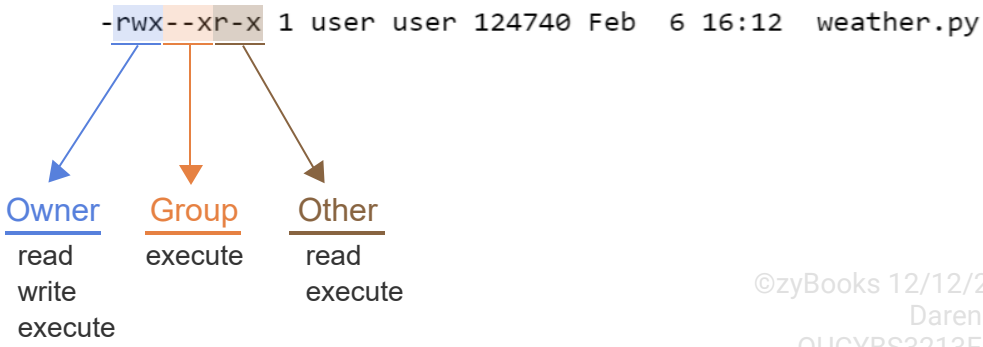
`---`	no permissions
`r--`	read
`-w-`	write
`--x`	execute
`rw-`	read and write
`r-x`	read and execute
`-wx`	write and execute
`rwx`	read, write, and execute

In Linux/Unix, three permissions classes exist: *owner*, *group*, and *other*. The `ls` command with option `-l` displays file permissions for the three classes. The first character of the `ls -l` output displays the file type and is not related to file permissions. The remaining nine characters are grouped in sets of three. The first set displays the permissions for the *owner* class, the second set for the *group* class, and the third set for the *other* class.

Ex: A file permissions of `-r-xrwxr-x` means that the *owner* class has read and execute permissions, the *group* class has read, write and execute permissions, and *other* class has read and execute permissions. Ex: A file permissions of `-rw-rw---x` means that the *owner* and *group* classes have read and write permissions, and the *other* class has execute permission.

PARTICIPATION ACTIVITY 2.14.1: Linux/Unix file permissions.


```
~$ ls -l
-rwxrw-rw- 1 user user  56700 Feb  6 16:13  hello.c
-rw-r--r-- 1 user user  22680 Feb  6 16:14  readme.txt
-rwx--xr-x 1 user user 124740 Feb  6 16:12  weather.py
~$
```



Animation content:

Static image: A box with five lines of text. The first line is "~$ ls -l". The second line is "-rwxrw-rw- 1 user user 56700 Feb 6 16:13 hello.c". The third line is "-rw-r--r-- 1 user user 22680 Feb 6 16:14 readme.txt". The fourth line is "-rwx--xr-x 1 user user 124740 Feb 6 16:12 weather.py" and is highlighted in green. The fifth line is "~$". Below this box is the text "-rwx--xr-x 1 user user 124740 Feb 6 16:12 weather.py". The text "rwx" is highlighted in blue with a blue arrow pointing to a list titled "Owner." The Owner list has "read," "write," and "execute." The text "--x" is highlighted in orange with an orange arrow pointing to a list titled "Group." The Group list has "execute." The text "r-x" is highlighted in brown with a brown arrow pointing to a list titled "Other." The Other list has "read" and "execute."
Step 1: The Owner class has read, write and execute permissions, and the Group and Other classes have read and write permissions to "hello.c".
A box with five lines of text. The first line is "~$ ls -l". The second line is "-rwxrw-rw- 1 user user 56700 Feb 6 16:13 hello.c". The third line is "-rw-r--r-- 1 user user 22680 Feb 6 16:14 readme.txt". The fourth line is "-rwx--xr-x 1 user user 124740 Feb 6 16:12 weather.py". The fifth line is "~$". The second line is highlighted in green. A copy of the second line appears below the box. The text "rwx" is highlighted in blue and a blue arrow appears pointing to a list titled "Owner." The Owner list has "read," "write," and "execute." The text "rw-" is highlighted in orange and an orange arrow appears pointing to a list titled "Group." The Group list has "read" and "write." The text "rw-" is highlighted in brown and a brown arrow appears pointing to a list titled "Other." The Other list has "read" and "write."
Step 2: The Owner class has read and write permissions, and the Group and Other classes have read permission to "readme.txt".
The green highlighting moves from the second line of text to the third line of text. The text below the box changes to a copy of the third line of text: "-rw-r--r-- 1 user user 22680 Feb 6 16:14 readme.txt". The text "rw-" is highlighted in blue. The text "execute" disappears from the Owner list so that the Owner list now has "read" and "write." The text "r--" is highlighted in orange. The text "write" disappears from the Group list so that the Group list now has "read." The text "r--" is highlighted in brown. The text "write" disappears from the Other list so that the Other list now has "read."
Step 3: The Owner class has read, write, and execute permissions, the Group class has execute permission, and the Other class has read and execute permissions to "weather.py".
The green highlighting moves from the third line of text to the fourth line of text. The text below the box changes to a copy of the fourth line of text: "-rwx--xr-x 1 user user 124740 Feb 6 16:12

weather.py". The text "rwx" is highlighted in blue. The text "execute" reappears on the Owner list so that the Owner list now has "read," "write," and "execute." The text "--x" is highlighted in orange. The text "read" disappears from the Group list and the text "execute" appears on the Group list so that the group list now reads "execute." The text "r-x" is highlighted in brown. The text "execute" appears on the Other list so that the Other list now reads "read" and "execute."

Animation captions:

1. The Owner class has read, write and execute permissions, and the Group and Other classes have read and write permissions to "hello.c".
2. The Owner class has read and write permissions, and the Group and Other classes have read permission to "readme.txt".
3. The Owner class has read, write, and execute permissions, the Group class has execute permission, and the Other class has read and execute permissions to "weather.py".

PARTICIPATION ACTIVITY 2.14.2: Filesystem permissions.

1) What is the group class permission for a file with permissions `-r-xrw-rwx`?
   - `r-x`
   - `rw-`
   - `rwx`

2) A file permission is `-rwxrw---x`. What action cannot be performed on the file by the *group* class?
   - Read
   - Write
   - Execute

3) A file permission is `-r-xrw-rwx`. What action can be performed on the file by all classes?
   - Read
   - Write
   - Execute

4) A file permission is `-r-xr--r--`. What action cannot be performed on

the file by any classes?

- Read
- Write
- Execute

5) A file can be read and executed by *owner*, *group*, and *other* classes and only written to by *owner* class. What is the file permission?

- `-r-xr--rwx`
- `-rw-r-xrw-`
- `-rwxr-xr-x`

6) A file can be executed by *owner*, *group*, and *other* classes and read only by *group* class. What is the file permission?

- `-rwxrw-r-x`
- `-rw-r--rwx`
- `---xr-x--x`

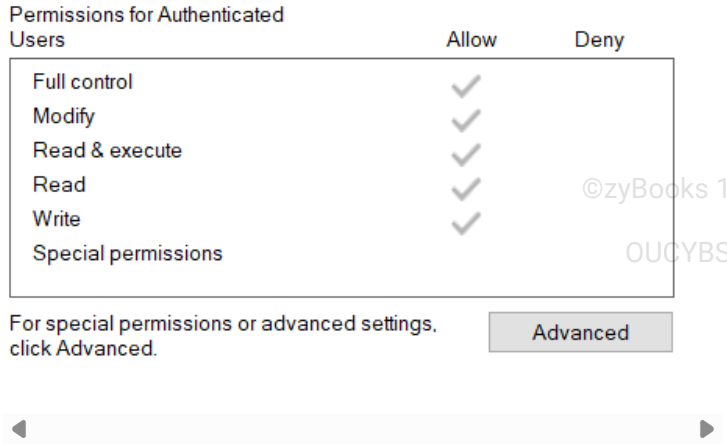
7) A file can be read by *owner*, *group*, and *other* classes, written to by *owner* class, and executed by *group* class only. What is the file permission?

- `-rwxrw-rwx`
- `-rw-r-xr--`
- `-r--rwxrw-`

File permissions in Windows can be configured in the *Security tab* of a file's properties. Five permissions exist for a file:

- ***Full control permission*** allows a user to read, write, modify, and delete a file, and change a file's permission.
- ***Modify permission*** allows a user to read and modify a file.
- ***Read & execute permission*** allows a user to read and execute a file.
- ***Read permission*** allows a user to read a file.
- ***Write permission*** allows a user to write to a file and modify a file's permissions.

Figure 2.14.1: File permissions in Windows 10.



PARTICIPATION ACTIVITY 2.14.3: Filesystem permissions.

1) A file permission for a user is *Modify*. Which action cannot be performed on the file by the user?
   - Read
   - Write
   - Execute

2) A file permission for a user is *Read & execute*. Which actions cannot be performed on the file by the user?
   - Read
   - Write
   - Execute

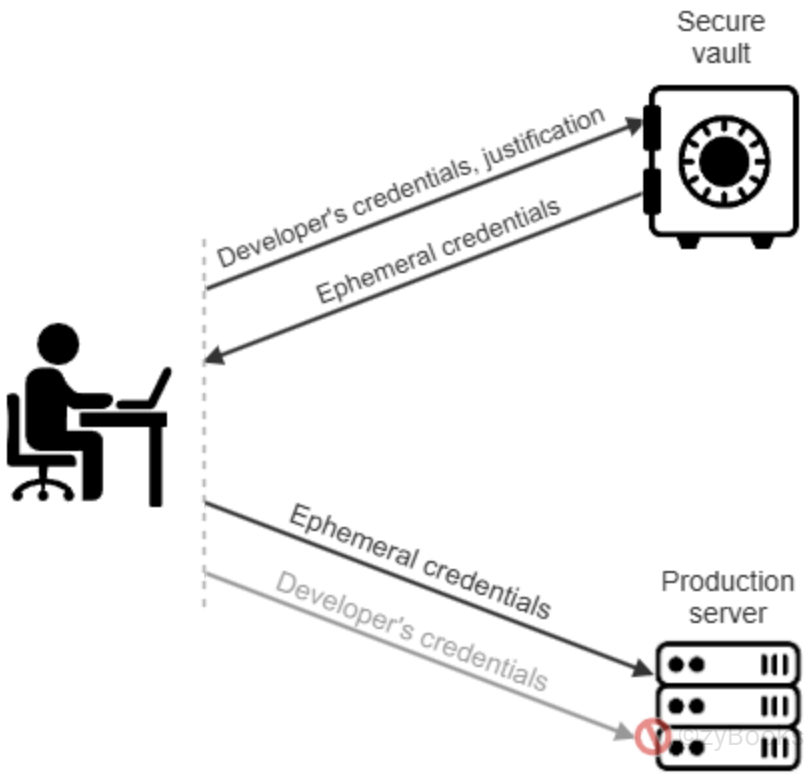
3) A file permission for a user is *Write*. Which action can be performed on the file by the user?
   - Read
   - Read and modify a file's permissions
   - Write and modify a file's permissions

Privileged access management (PAM)

Privileged access management (PAM), also known as ***privileged identity management (PIM)***, is the set of controls, tools, and processes for managing, securing, and monitoring a privileged account. A ***privileged account*** is an account that has elevated access rights in a system. Ex: A superuser account such as the *administrator* account in Windows and the *root* account in Linux/Unix is a privileged account.

A privileged account has greater security permissions, and thus has the potential to cause more harm than any other account type. A privileged account may have the authorization to override security controls and perform tasks such as configure networks, shut down systems, and provision accounts. The management of a privileged account should ensure that the principle of least privilege is enforced. The ***principle of least privilege*** states that a user should only be granted the minimum access rights required to perform the user's job.

PARTICIPATION ACTIVITY 2.14.4: PAM tools.



Animation content:

Static image:

Animation captions:

1. A developer needs to access a production server to troubleshoot a bug. The developer's account does not have production server privileges, so access is denied.
2. The developer requests just-in-time permissions from a secure vault that stores privileged account credentials. The developer provides justification for production server access.
3. The secure vault grants the developer temporary access to the production server based on security policies. The secure vault generates ephemeral credentials for the developer.
4. The developer is granted access to the production server using the ephemeral credentials, which are removed after the session.

PARTICIPATION ACTIVITY 2.14.5: Privileged access management (PAM).

How to use this tool

PAM or PIM **Privileged account** **Principle of least privilege**

	A set of controls, tools, and processes for managing, securing, and monitoring a privileged account.
	A principle that states that a user should only be granted the minimum access rights required to perform the user's job.
	An account that has elevated access rights in a system.

Reset

PARTICIPATION ACTIVITY 2.14.6: Privileged access management (PAM).

1) The management of a privileged account is more important than the management of other account types.
   - ○ False
   - ○ True

2) A PAM system manages, secures, and monitors all user accounts.

- False
- True

3) The principle of least privilege ensures that a user has the access rights required to perform the user's job.

- False
- True

Conditional access

Conditional access is access control based on a predefined access criteria or condition. An access criteria may include a subject's geolocation, device type, IP address, or a device's security state such as having an up-to-date anti-virus signature file or the latest security patch. An access control decision may include the blocking, limiting, or allowing access to an object, or requiring a subject to complete multi-factor authentication (MFA), or request the installation of a critical security patch on a device.

Conditional access is commonly used to enforce security policies. A system that supports conditional access analyzes real-time conditions and a system's security policies to make a security enforcement decision. Ex: Conditional access may be used to enforce a security policy that requires a user to complete a two-factor authentication (2FA) before granting a user access to an accounting application. Ex: Microsoft Azure Active Directory and Intune support conditional access.

PARTICIPATION ACTIVITY 2.14.7: Conditional access.

1) Conditional access can be used to control access to an object based on a device's IP address.

- False
- True

2) A security policy that requires a user to be inside a geofenced area before granting a user access to a database application cannot be implemented by conditional access.

- False
- True

3) A user's access to an accounting application can be denied by conditional access based on the user's device not having a critical security patch.

- False
- True

2.15 LAB: Account management (Walkthrough)

IT-Labs are not printable at this time.

2.16 LAB: Account management (Scenario)

IT-Labs are not printable at this time.

2.17 LAB: Securing accounts (Walkthrough)

IT-Labs are not printable at this time.