

Information security

Andrea Pintér-Husztí

2024. december 13.

Contents

1 AAA

- Basic concepts
- Entity authentication
- Authorization
- Accounting

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ 🔍 ↺

Basic concepts

RFC (Request For Comments) 4949:

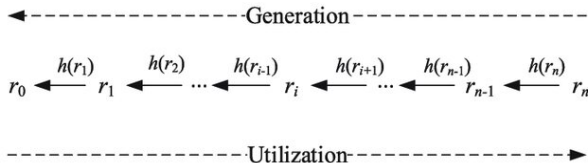
- **Identification:** Presenting an identifier to the system.
- **Authentication:** The process that verifies an identity. Presenting verification information that corroborates the binding between the entity and the identifier.
- **Authorization:** A process of specifying access rights to resources.
- **Accounting:** Keeping track of user activity while users are logged in to a network.

Means of Authentication

- **Something the individual knows:** password, a personal identification number (PIN), or answers to a prearranged set of questions
- **Something the individual possesses:** electronic keycards, smart cards, and physical keys. This type of authenticator is referred to as a token.
- **Biometric**
 - **Something the individual is (static biometrics):** fingerprint, retina, face, DNS
 - **Something the individual does (dynamic biometrics):** voice pattern, handwriting characteristics, typing rhythm

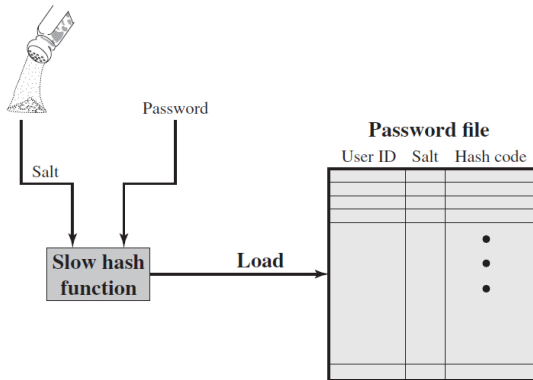
Password-based authentication

- Local password-based authentication
- One-time passwords: time synchronized solution (token generates fresh digits at regular intervals), hash chain based solution

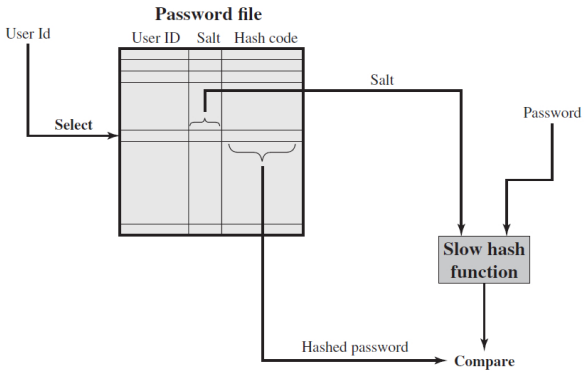


- Remote password-based authentication

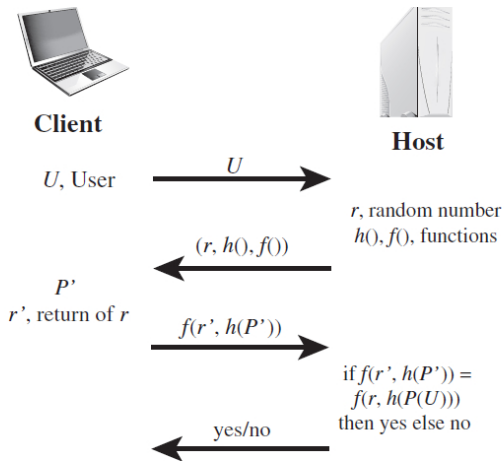
Unix password-based authentication - registration



Unix password-based authentication - verification



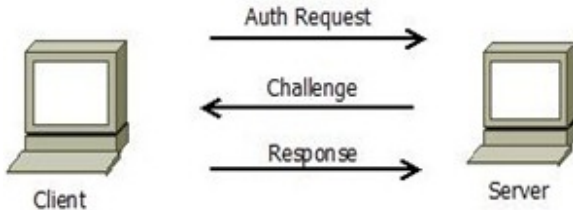
1. *Journal of the American Medical Association*, 2000; 283: 2689-2694.



(a) Protocol for a password

Challenge-response protocols

- Computer system generates a challenge, such as a random string of numbers.
- The client device generates a response based on the challenge.
- Examples: based on symmetric primitives (encryption, MAC), asymmetric primitives (encryption, digital signature)



- Use of a smart card as a national identity card.
- eID card can provide stronger proof of identity and be used in a wide variety of applications.



eID - Electronic Identity Cards

Printed information:

- Personal data: name, date of birth, and address
- Document number: An alphanumerical nine-character unique identifier of each card.
- Card access number (CAN): A six-digit decimal random number printed on the face of the card. This is used as a password.
- Machine readable zone (MRZ): Three lines of human- and machine-readable text on the back of the card. This may also be used as a password.

elD functions

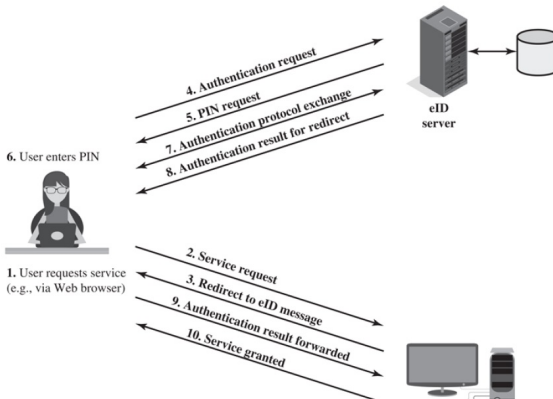
- ePass: This function is reserved for government use and stores a digital representation of the cardholder's identity. This function is similar to, and may be used for, an electronic passport. Other government services may also use ePass. The ePass function must be implemented on the card.
- eID: This function is for general-purpose use in a variety of government and commercial applications. The eID function stores an identity record that authorized service can access with cardholder permission. Citizens choose whether they want this function activated.
- eSign: This optional function stores a private key and a certificate verifying the key; it is used for generating a digital signature.

eID functions

Table 3.4 Electronic Functions and Data for eID Cards

Function	Purpose	PACE Password	Data	Uses
ePass (mandatory)	Authorized offline inspection systems read the data.	CAN or MRZ	Face image; two fingerprint images (optional); MRZ data	Offline biometric identity verification reserved for government access
eID (activation optional)	Online applications read the data or access functions as authorized.	eID PIN	Family and given names; artistic name and doctoral degree; date and place of birth; address and community ID; expiration date	Identification; age verification; community ID verification; restricted identification (pseudonym); revocation query
	Offline inspection systems read the data and update the address and community ID.	CAN or MRZ		
eSign (certificate optional)	A certification authority installs the signature certificate online.	eID PIN	Signature key; X.509 certificate	Electronic signature creation
	Citizens make electronic signature with eSign PIN.	CAN		

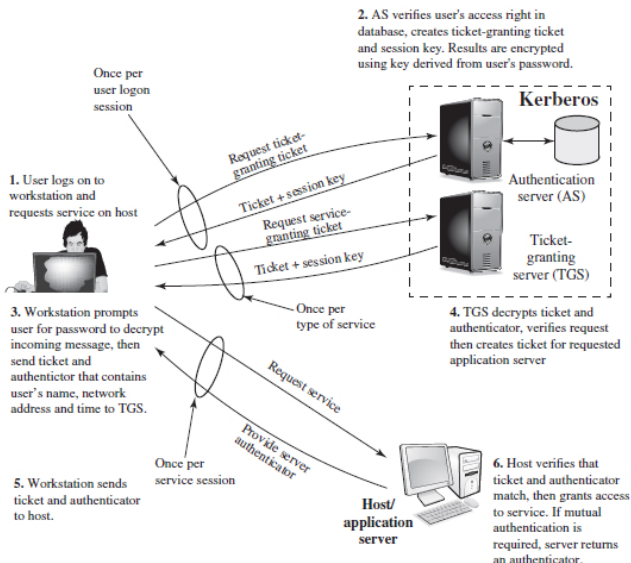
elD functions



Distributed systems remote authentication: Kerberos

- Authentication is not done by several servers, but by one authentication server.
- Developed by MIT, Internet standard.
- Kerberos requires the user to authenticate to all services and all servers to authenticate to the client.
- Protocol participants: clients, application servers and a Kerberos server.
- The authentication server (AS) stores the passwords of all clients in a central database.
- If AS authenticates the client and notifies the application server. AS exchanges a secret key with each application server.

Distributed systems remote authentication: Kerberos



Kerberos authentication process

- A user logs on to a workstation and requests a service from a server.
- Client process starts: the client sends the ID and requests the TGT (ticket-granting ticket).
- AS returns: encryption of TGT and session key (single-use secret key), where the key is derived from the user's password.
- The client asks for the user's password, which it uses to decrypt the message. If the password is correct, the TGT and the session key are given.
- Neither the password nor the hash of it has been sent.

Kerberos ticket

- The ticket shows that AS has authenticated the user. The ticket contains: user ID, TGS ID, timestamp, expiry date, and the same session key. The ticket is encrypted with the symmetric key exchanged between AS and the server.
- This ticket could provide access directly to the application server, but then each server access would have to be addressed to the AS. Instead, the ticket is generated for TGS (ticket-granting server).
- With a ticket you can request more tickets. TGT can be used more than once.
- Attacker's goal: To obtain TGT
 - TGT encrypted with a shared secret key between AS and TGS is encrypted with a key derived from the password
 - short expiration time (8 hours)

Request a Kerberos service

- The user wants to access a service.
- The client process sends to TGS: provider server ID, TGT, authenticator, which is a session key-encrypted user ID+network address+timestamp.
- TGS decrypts the TGT ticket, checks it (validity period, IDs, etc.) and gets the session key.
- Decrypts the authenticator with the session key and verifies it.
- Ticket: session key distribution, authenticator: proves the identity of the client
- The authenticator has a short expiration time and can be used once. A new authenticator will be required for the next service request.

Kerberos service ticket

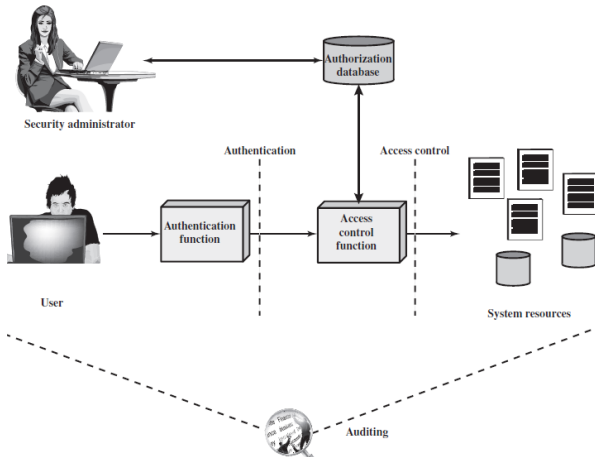
- The next 2 steps are the same as the previous 2.
- TGS sends a service-granting ticket, a new session key to the client. The whole message is encrypted with the old session key.
- The service-granting ticket is encrypted with a shared key between the TGS and the service server.
- Each time the client wants to use the service, it sends a service-granting ticket and a new authenticator.
- If the client requests authentication, the service server adds 1 to the received timestamp and encrypts it with the session key.
- The session key can be used to encrypt subsequent messages.

Basic concepts

RFC (Request For Comments) 4949:

- **Authentication** :The process that verifies an identity. Presenting verification information that corroborates the binding between the entity and the identifier.
- **Authorization**: A process of specifying access rights to resources.
- **IT Audit** An independent review and examination of system records and activities in order to test for adequacy of system controls, to ensure compliance with established policy and operational procedures, to detect breaches in security, and to recommend any indicated changes in control, policy, and procedures.

Access control



Access control

- An access control mechanism mediates between a user and system resources.
- The system must first authenticate an entity seeking access. The authentication function determines whether the user is permitted to access the system at all.
- Then the access control function determines if the specific requested access by this user is permitted.
- An auditing function monitors and keeps a record of user accesses to system resources.

Access Control Policies

An access control policy, which can be embodied in an authorization database, dictates what types of access are permitted, under what circumstances, and by whom.

- Discretionary access control (DAC)
- Mandatory access control (MAC))
- Role-based access control (RBAC)
- Attribute-based access control (ABAC)

DAC

Discretionary access control (DAC):

- Data management is decentralised.
- Access to an object is essentially determined by the owner of the object, the creator of the object.
- The owner of the object may donate or withdraw access rights from other persons.
- The advantage is that there is no need for a central registration, it is the responsibility of the owner to keep the data at an appropriate level of protection. However, this freedom can in some cases be at the expense of security.

MAC

Mandatory access control (MAC)

- Security labels for all individuals and objects are defined from a single central location.
- The basic idea is to compare the security labels of resources (e.g. confidential, secret, top secret) with the privileges of the entities.
- The object owner cannot determine who can access the object and in what form.
- The security mechanism controls the access rights of each object and cannot be changed by any user.
- Only used in environments (government, military) where different levels of security classification exist.

RBAC

Role-based access control (RBAC):

- Controls access based on the roles that users have within the system and on rules stating what accesses are allowed to users in given roles.
- Each user can be assigned a different set of roles.
- Roles define and describe which operations can be performed on which objects in a given role.
- Rights are not defined according to persons, but according to roles (e.g. in the life of the company).

ABAC, CBAC

- Attribute-based access control (ABAC): The model is based on the attributes of the user, the resource and the current environmental conditions (e.g. time). ABAC can be used at the firewall, server, application, database and data levels.
- A Code-based Access Control (CBAC): Assigns access rights to executable files. The rules refer to the origin of the code, the identity of the code (e.g. the hash of the executable) or other properties of the executable, not the identity of the user who launched the executable.

Basic elements - Subjects

Subjects: an entity capable of accessing objects.

Basic access control systems typically define three classes of subject:

- Owner: This may be the creator of a resource.
- Group: a group is a set of users who have the same rights over certain resources. Groups are usually defined in terms of tasks shared by several users. A user can belong to more than one group.
- Other: the minimum right for those who can log in but are neither an owner nor a group member

Basic elements - Objects

Objects: a resource to which access is controlled.

In general, an object is an entity used to contain and/or receive information. Examples include records, blocks, pages, segments, files, portions of files, directories, directory trees, mailboxes, messages, and programs.

Basic elements - Access rights

Access rights: The access right defines how the actor can access the objects.

- Read: user may view information in a system resource. Read access includes the ability to copy or print.
- Write: User may add, modify, or delete data in system resource. Write access includes read access.
- Execute: User may execute specified programs.
- Delete: User may delete certain system resources, such as files or records.
- Create: User may create new files, records, or fields.
- Search: User may list the files in a directory or otherwise search the directory.

DAC

In the case of DAC, each object is specified by which actors have access to it. The objects and the operators can be specified:

- Access Control Matrix
- Access Control List
- Capability List
- Authorization table
- Extended Access Control Matrix

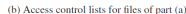
Access Control Matrix

Its columns correspond to each object and its rows to each subject. The cells of the matrix contain the access rights granted. This results in a sparse matrix.

		OBJECTS			
		File 1	File 2	File 3	File 4
SUBJECTS	User A	Own Read Write		Own Read Write	
	User B	Read	Own Read Write	Write	Read
	User C	Read Write	Read		Own Read Write

(a) Access matrix

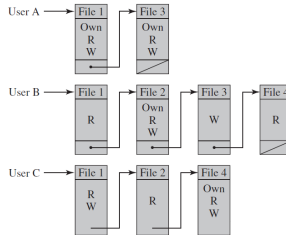
The columns of the matrix are sorted into a list.



Cons: Slow to check which files a user has access to.

Capability List

The rows of the matrix are sorted into a list.



(c) Capability lists for files of part (a)

Benefit: It is easy to control which files a user has access to. It uses little memory to store them.

Cons: Slow to check who has access to a file.

Authorization table

A row in the table contains a subject, an object and an access right.

Table 4.1 Authorization Table for Files in Figure 4.2

Subject	Access Mode	Object
A	Own	File 1
A	Read	File 1
A	Write	File 1
A	Own	File 3
A	Read	File 3
A	Write	File 3
B	Read	File 1
B	Own	File 2
B	Read	File 2
B	Write	File 2
B	Write	File 3
B	Read	File 4
C	Read	File 1
C	Write	File 1
C	Read	File 2
C	Own	File 4
C	Read	File 4
C	Write	File 4

Sorting by subject gives a capability list, sorting by object gives an access list.

Extended Access Control Matrix

Access rights are defined not only at the resource level, but also at the process and storage level. The * indicates that someone is granted not only the right granted, but also the ability to delegate the right gained.

		OBJECTS								
		Subjects			Files		Processes		Disk drives	
		S ₁	S ₂	S ₃	F ₁	F ₂	P ₁	P ₂	D ₁	D ₂
SUBJECTS	S ₁	control	owner	owner control	read*	read owner	wakeup	wakeup	seek	owner
	S ₂		control		write*	execute			owner	seek*
	S ₃			control		write	stop			

* = copy flag set

Figure 4.3 Extended Access Control Matrix

- Process: delete, stop, wake up
- Device: read, write, control (e.g. seek)
- Storage: read/write

Mandatory access control (MAC)

- We use it in information systems where more restrictive security requirements have to be met.
- Access control is uniform.
- All objects and users are assigned a security label.
- The tag consists of two parts: a security level and a domain identifier (e.g. TopSecret, finance)
- Secret levels: U (unclassified) $< C$ (classified) $< S$ (secret) $< TS$ (topsecret) (strict relation)

Management of rights

- Privileges are determined by the relationship between the object's security label and the user's security label.
- A user has access to an object if the user's *security level* is greater than or equal to the object's security level and its scope is (a subset of) the scope of the object.

Role-based access control (RBAC)

- RBAC is based on roles, typically linked to jobs. A user can have multiple roles.
- Active research area, published as a standard in 2009 (FIPS PUB 140-3).
- While the role of a user can change, the role is mostly relatively static.
- Role privileges for a given resource are less likely to change.

RBAC

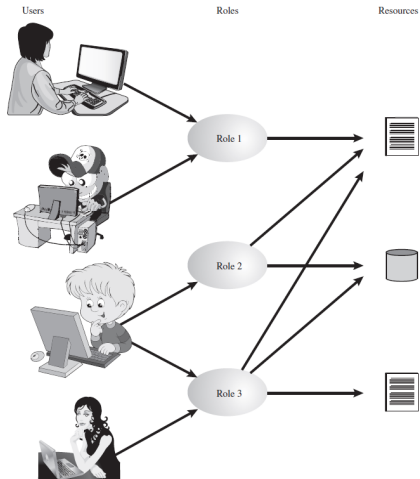


Figure 4.6 Users, Roles, and Resources

Assigning users to roles

	R_1	R_2	$\dots$	R_n
U_1	$\times$			
U_2	$\times$			
U_3		$\times$		$\times$
U_4				$\times$
U_5				$\times$
U_6				$\times$
$\vdots$				
U_m	$\times$			

RBAC - Extended Access Control Matrix

		OBJECTS								
		R ₁	R ₂	R _n	F ₁	F ₂	P ₁	P ₂	D ₁	D ₂
ROLES	R ₁	control	owner	owner control	read *	read owner	wakeup	wakeup	seek	owner
	R ₂		control		write *	execute			owner	seek *
	•									
	•									
	•									
	R _n			control		write	stop			

Figure 4.7 Access Control Matrix Representation of RBAC

Basic concepts

RFC (Request For Comments) 4949:

- **Authentication:** The process that verifies an identity. Presenting verification information that corroborates the binding between the entity and the identifier.
- **Authorization:** A process of specifying access rights to resources.
- **Accounting:** Keeping track of user activity while users are logged in to a network.

- Mechanisms are needed to securely transfer access requests, attributes, rules and decisions between nodes.
- These mechanisms are mostly based on cryptography.
 - Attribute-based encryption
In the use of cloud systems, the need to ensure that only those with a given attribute can access the data has been addressed.
 - Key-Policy Attribute-based Encryption (KP-ABE): the key is generated based on the specified attributes
 - Ciphertext-Policy Attribute-based Encryption (CP-ABE): the document is generated based on the specified attributes

Distributed systems: Authorization

- OAuth2 – OpenID Connect: authentication and authorization
- a way for internet users to grant websites or applications access to their information on other websites but without giving them the credentials (e.g. passwords).
- Participants: user (owner of the resources), resource server, authorization server (AS), client application
- Client application requests access to the services, resource server
- AS authenticates users, applications
- Assumption: client applications are registered to AS (public ID, symmetric secret key between client app. and AS for secure communication, registration of redirect_URI (browser is redirected to these URI))

OAuth2

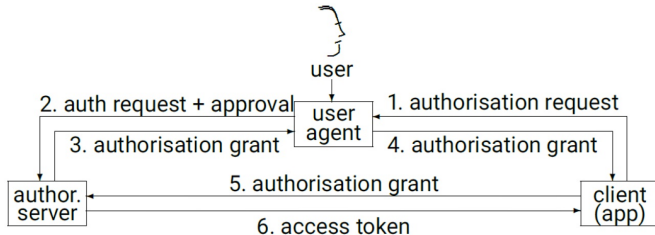
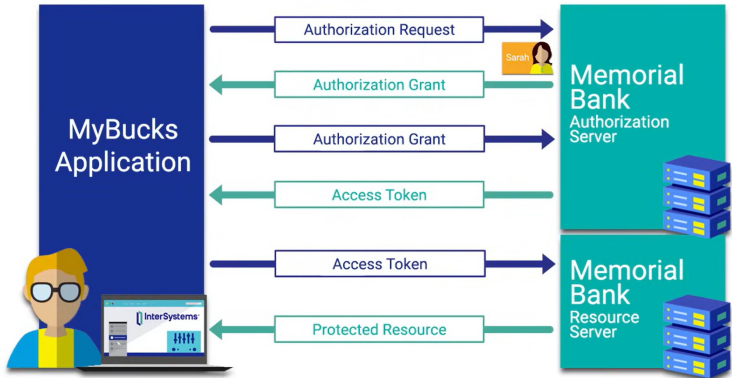


Figure 13.5: Message flow in OAuth 2.0.

Workflow of OAuth 2.0



OAuth2

- The user (browser) receives an authorization request from the client application, the request includes `redirect_URI`
- The user forwards the request to AS, indicating his approval, via a secret channel
- If authorized for the request, AS sends the authorization grant to the user and `redirect_URI`
- Client app. sends the received grant and `redirect_URI` to AS
- AS authenticates users, applications and sends an access token
- The access token is used by the client app to access resources.

- ◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡

Accounting

- Accounting processes: Processes that are carried out after the occurrence of events.
- Such a process could be a regular audit that checks, whether the organisation is in compliance with the regulations in force.
- Technical audit: examines log files (logs) for evidence of an attack.
- Operating systems, network devices, applications may also have log files.
- It can also be an incident-triggered scan that tries to identify vulnerabilities.

Accounting

- Administrators should have audit policy that specify which events are to be are logged. e.g. successful or unsuccessful user authentication
- Logging is as robust as the quality of the evidence logs collected during operations.
- Attackers may try to hide their tracks by, by deleting incriminating log entries.
- Protecting evidence against tampering is cryptographic e.g. the use of hash chains

Accounting

- Log files are a large amount of data, where many entries are not security relevant, so automated processing is required.
- Machine learning techniques can help to detect anomalies.
- Visualisation techniques try to draw the administrators' attention to the most relevant events.
- Accounting is only as strong as the organisational security policies on connecting devices, e.g. USB tokens, to internal systems, and policies on access to external web sites.
- Accounting is only as strong as the defences against software vulnerabilities that can be exploited to run code under a user identity without the user being aware of that fact, e.g. so-called drive-by-downloads.