



CyBOK

Authentication, Authorisation & Accountability

Dieter Gollmann
Hamburg University of Technology

bristol.ac.uk



© Crown Copyright, The National Cyber Security Centre 2021. This information is licensed under the Open Government Licence v3.0. To view this licence, visit <http://www.nationalarchives.gov.uk/doc/open-government-licence/>.

When you use this information under the Open Government Licence, you should include the following attribution: CyBOK AAA Knowledge Area Issue 1.0 © Crown Copyright, The National Cyber Security Centre 2021, licensed under the Open Government Licence <http://www.nationalarchives.gov.uk/doc/open-government-licence/>.

The CyBOK project would like to understand how the CyBOK is being used and its uptake. The project would like organisations using, or intending to use, CyBOK for the purposes of education, training, course development, professional development etc. to contact it at contact@cybok.org to let the project know how they are using CyBOK.

bristol.ac.uk

CyBOK

Security is a fashion industry

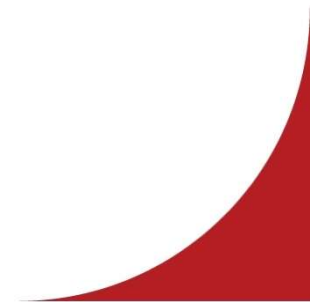
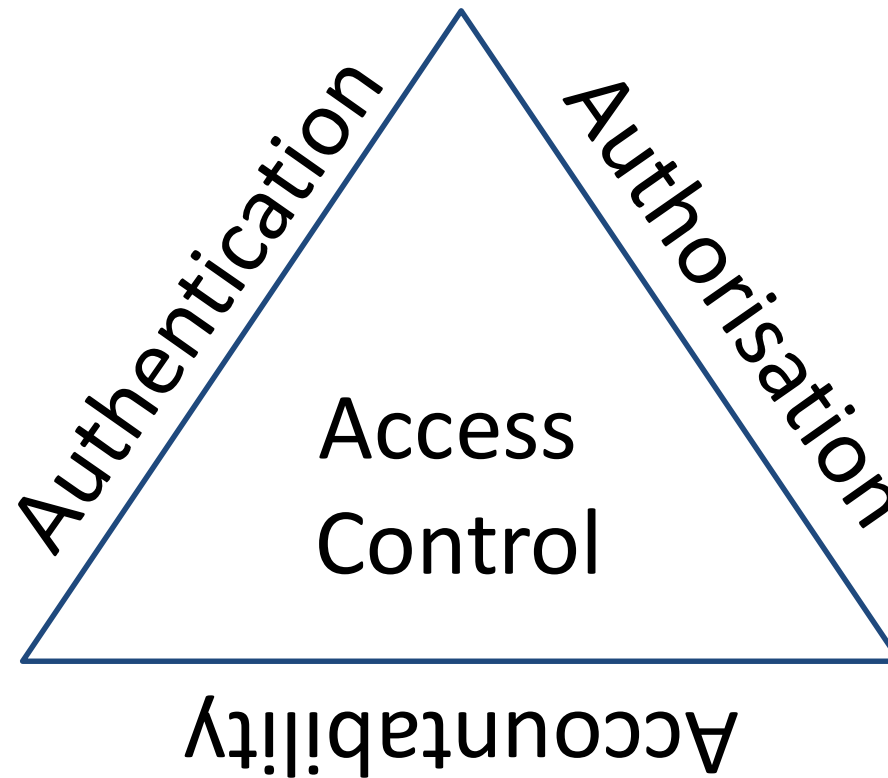
CyBOK

We don't do new things in computing; we just rename the old stuff

Rear Admiral Grace Murray Hopper



Authentication, Authorisation & Accountability




Access Control


- Access control: who is allowed to do what?
- Who: a person, a machine, an app, ...
- What: an action on an object
- Action: read, write, execute, ...
- Object: a file, a directory, data, a service, ...
- Security policy: expresses who is allowed to do what



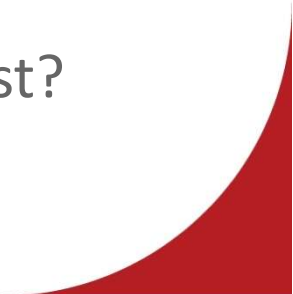
Principals & Subjects

- Security policies are a collection of **rules**
 - A rule grants or denies **access rights** to a **principal**
 - **Permission** is a synonym for access right
 - In security policies on users, the principals are **user identities**
 - When a user logs in to a system, a process is spawned that runs under the user identity of that person
 - Requests to protected resources are issued by this process
 - In the terminology of access control, the process is the **subject**
 - Subjects “speak for” principals
- 


Principals & Subjects

- *Because access control structures identify principals, it is important that principal names be globally unique, human-readable and memorable, easily and reliably associated with known people. [Morrie Gasser, 1990]*
 - This was true thirty years ago
 - Multi-user operating systems were the main use case for access control then
 - The principals in today's web applications are domains
 - The principals on today's smartphones are the apps
 - Deciding which resources an app may access is still a matter of access control
- 


Authentication

- Access decisions are based on the answers to three questions
 - Who issued the request?
 - To be precise, what is known about the origin of a request?
 - Verifying evidence about the origin of a request is called **authentication**
 - What is requested?
 - Which rules are applicable when deciding on the request?
- 


User Authentication

- Users can be authenticated based on
 - Something they know (password, PIN)
 - Something they hold (a smart card, a phone, a token)
 - Who they are (biometrics: fingerprint, face, gait, ...)
 - What they do (writing on a pad, typing on a keyboard, swiping a touchscreen, gait, ...)
 - Where they are (geographic location, on premise or off-premise, in a control room, reachable on a registered phone number)
 - Multi-factor authentication (MFA) combines authentication modes
- 


Passwords, Old & New

- Advice on strong passwords
 - Mix upper and lower case characters, numerals, special characters
 - Don't write password down
 - Hide password during login
 - Change regularly
 - Threat model: user has one password, compromise at the user side or by guessing
- New advice, NIST SP 800-63
 - Long passphrases
 - Leave a copy of the password in a safe place
 - Passwords can be displayed when entered in a secure environment
 - Only change for a reason
 - Threat model: user has many passwords, may be compromised at server
- 

Biometrics

- Biometric feature must uniquely identify a person
 - Feature must be stable
 - A feature might change with age
 - Feature can be conveniently captured in operational settings
 - Feature cannot be spoofed during user authentication!
 - Liveness detection to thwart fake fingers, fake fingerprints taped to finger (James Bond in *Diamonds are Forever*), etc.
 - Biometrics suit local authentication: unlock smartphone, automated passport control
 - Not very suitable for remote authentication
 - Can be used as one of the factors in MFA
- 

Authentication Tokens

- Device that computes a one-time password (OTP) synchronised with the verifier, or a response to a challenge set by the verifier
 - Known as **tokens** or **security keys**, but both terms also have other meanings in IT security
 - Possession of device is necessary for successful authentication
 - Authentication based on “something you hold”
 - Small hand-held device with an LED display to show OTP
 - E.g., RSA SecureID and YubiKey
 - Could have a numeric keypad and a ‘sign’ button
 - PhotoTAN: challenge sent as QR code read from user’s screen
- 

FIDO UAF (Universal Authentication Framework)

- FIDO token supports public key cryptography (digital signatures)
- Token can create public key / private key pairs
- Public keys serve as identifiers, users can register an individual public key with each server (website)
- Private keys are used for generating digital signatures
- User authentication based on a challenge-response pattern
 - FIDO token digitally signs the response to the server's challenge
 - Response verified using the public key registered with the server




Behavioural Authentication

- Authentication based on “what you do”
 - Keystroke dynamics
 - Characteristic features of hand writing, speed and pen pressure
 - Voice recognition
- Smartphones are well equipped for behavioural authentication
- Challenge: can these features be spoofed during user authentication?

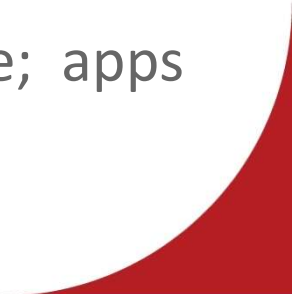


Continuous Authentication

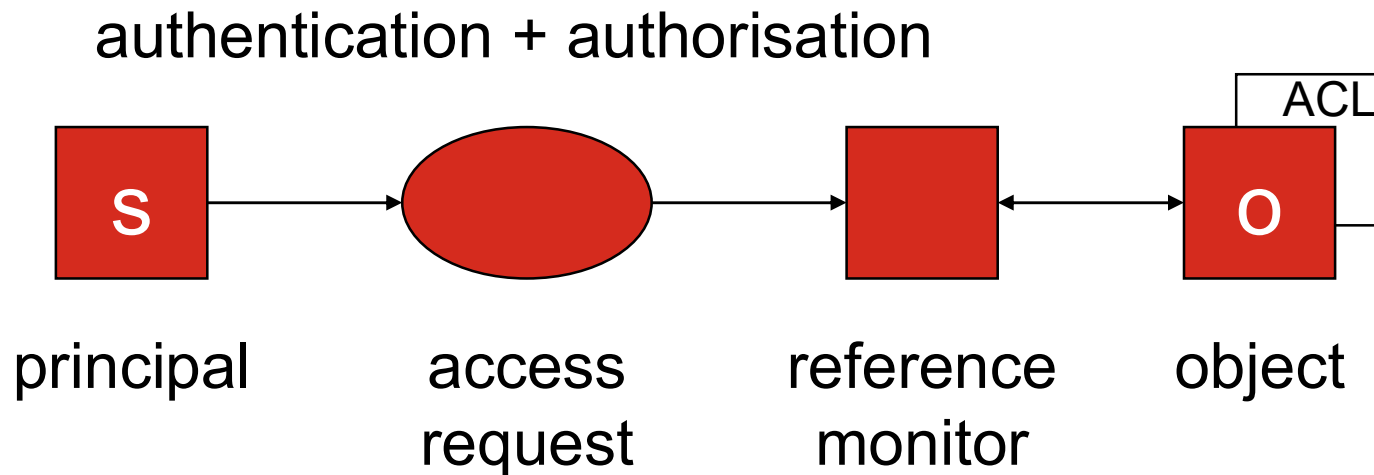
“Minimum Friction, Maximum Security”

- Don't authenticate a user only at the start of a session but continuously during the session
 - May use behavioural authentication; does not inconvenience the user with authentication ceremonies
 - Variations in user behaviour may cause false rejects
 - A cold may affect voice recognition
 - Requires a smooth fall-back when behavioural authentication fails
 - Strength of security depends on strength of liveness detection
 - Can synthesized speech or a voice imitator trick the system?
 - **Without a precise threat model, behavioural authentication can only offer uncertain security guarantees**
- 

Multi-Factor Authentication

- Combines user authentication modes to increase security
 - E.g., the European Payment Services Directive 2 (PSD2), prescribes two-factor authentication (2FA) for online payments
 - Two factors: password and authentication token for computing transaction numbers (TANs) uniquely tied to the transaction
 - If tokens are tied to one payment service only, customers have to carry multiple devices
 - Tokens that can be used with several services require some level of prior standardisation
 - Token could be a smartphone registered with the service; apps for several services can be installed on the same device
- 


Access Control = Authentication + Authorisation



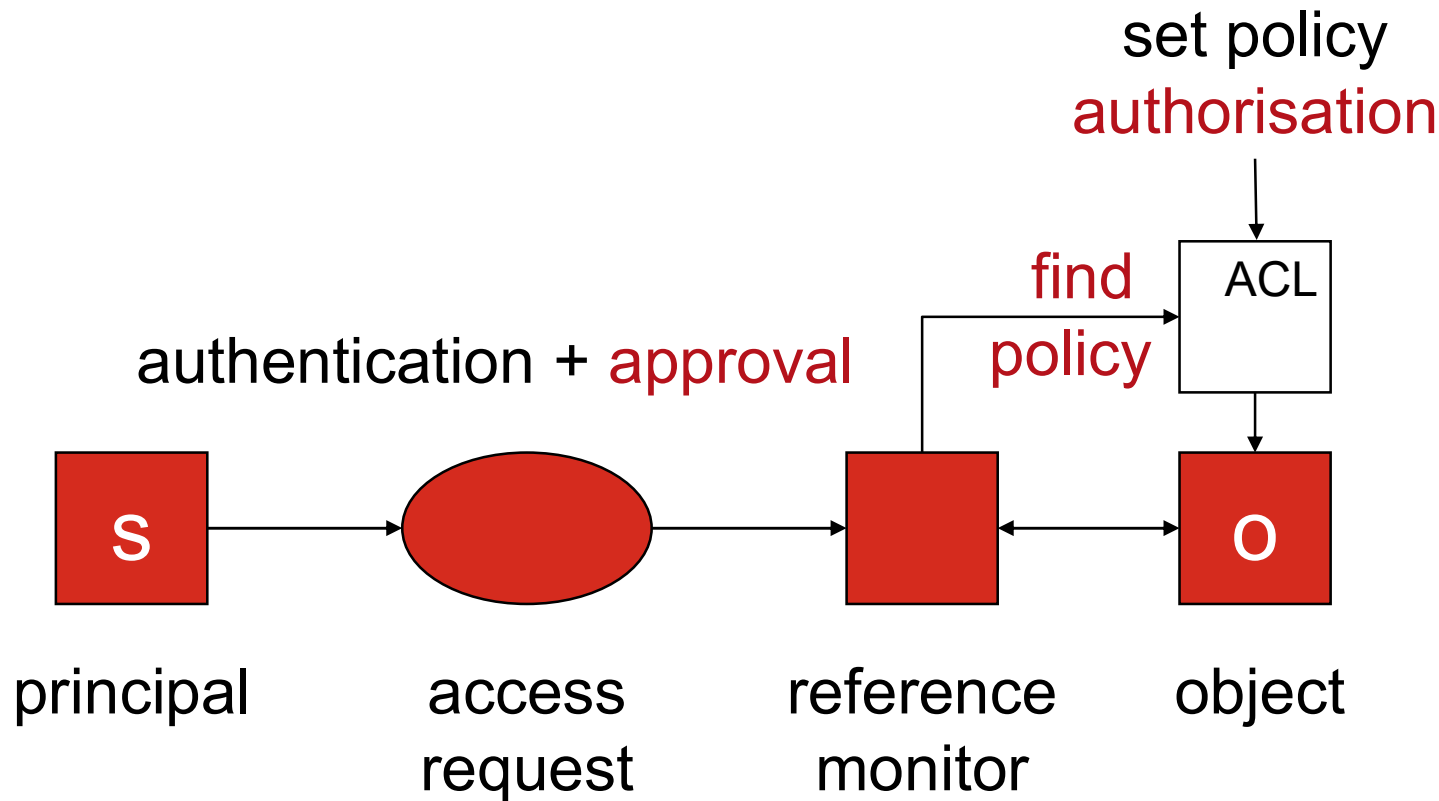
B. Lampson, M. Abadi, M. Burrows, E. Wobber: Authentication in Distributed Systems: Theory and Practice, ACM Transactions on Computer Systems, 10(4), pages 265-310, 1992

Authorisation

- **Authorisation** checks whether an access request for an object originating from a principal should be granted or denied
 - The **reference monitor** is the **guard** enforcing the policy
 - The policy is given in an **Access Control List (ACL)**
 - The ACL is attached to the object
 - Setting the policy is out of scope for this model

 - A more general view on access control has two more phases
 - Setting the policy, also called **authorisation**
 - Finding the rules applicable for a given access request
- 

Access Control = Authorisation + Authentication + Approval




Access Control – Case Studies


- “Military” security policies
- “Commercial” security policies – RBAC
- Digital rights management
- Mobile apps
- Origin-based policies in the web
- UCON & ABAB



“Military” Security Policies


- Security policies for classified data (1970s)
 - Labels unclassified – confidential – secret – top secret
 - Users have clearance levels, objects have security labels
 - “No write-down” and “no read-up” rules on data flow
 - **Discretionary access control (DAC)** policies captured by an access control matrix, **set by the owner of an object**
 - **Mandatory access control (MAC)** policies based on labels, **set by the system administrator**
 - **BellLaPadula** model for confidentiality, **Biba** model for integrity
- 

“Commercial” Security Policies


- Based on **roles, well-formed transactions, separation of duties**
 - Often a stronger emphasis on availability and integrity than on confidentiality
 - **Role-based Access Control (RBAC)**: users are assigned roles, are authorised to execute the operations linked to their active role
 - Roles are the principals in access rules
 - Operations can be well-formed transactions with built-in integrity checks that mediate the access to objects
 - **Separation-of-duties** policies stop single users from becoming too powerful
 - The NIST RBAC model defines four flavours of RBAC
- 

Digital Rights Management (DRM)

CyBOK

- Has its origin in the entertainment sector
 - Objects of access control: games, videos, music, ... (“content”)
 - Uncontrolled copying of digital content seriously impairs the business models of content producers and distributors
 - These parties must be able to control how their content can be accessed and used on their customers’ devices
 - Change in perspective: DRM imposes the security policy of an external party on the system owner rather than protecting the system owner from external parties
 - DRM needs tamper resistant [roots of trust](#)
- 

Roots of Trust

- Level of tamper resistance required depends on threat model
 - **Trusted Platform Modules** are a hardware solution giving a high degree of assurance.
 - **Trusted Execution Environments** (TEEs) such as enclaves in Intel SGX are a solution in system software
 - **Attestation** provides trustworthy information about a platform configuration
 - **Direct anonymous attestation** protects user privacy
 - Remote attestation supports security policies that are predicated on the software running on a remote machine
- 

Mobile Apps (Android)

- Smartphones typically have a single owner, hold private user data, offer various communication functions, can observe their surroundings (camera, microphone), know their location (GPS)
- Apps are the **principals** for access control
- The **objects of access control** are the sensitive data and device functions on a phone
- Access control protects the privacy of the owner and the integrity of the platform




Android Permissions


- **Normal permissions** do not raise privacy or platform integrity concerns; granted by app developer in the app manifest
- **Dangerous permissions** can impact privacy and must be granted by user
 - Since Android 6.0, users are asked to authorise a permission when it is first needed
- **Signature permissions** have an impact on platform integrity; must be granted by platform provider
 - Permission is granted when app and permission are signed by the same private key




Origin-Based Policies

- In web applications, security policies specify which resources a script in a web page is allowed to access, or the hosts a script may send requests to
 - Web applications are here the principals in access control
 - By convention, principals are named by the domain names of the server hosting an application;
 - The [Same Origin Policy](#) (SOP) states that a script may only connect back to the origin it came from or may only read cookies from its own origin
 - [Origin](#) is defined by scheme (protocol), host name and port number
- 


Cross-site Scripting

- Cross-site scripting attacks affect browsers that let all scripts that arrive in a web page speak for the origin of that page
 - Attacker injects a script into a page of some other server; the script may then read cookies that server had set
 - With [Content Security Policy](#) (CSP) the web server informs the browser about the sources of authorized scripts, e.g., a directory path on the server
 - In [strict CSP](#) policies, the server declares a nonce in the CSP policy as the authorized 'source' and labels all scripts sent to that client with that nonce
- 


Usage Control (UCON)

- Framework for authorisations based on the attributes of subject and object, obligations and conditions
 - **Obligations** are additional actions a user has to perform to be granted access, e.g., click on a link to agree to terms of use, or actions the system must perform, e.g., log an access request
 - **Conditions** are aspects independent of subject and object, e.g., time of day when a policy permits access only during office hours or the location of the machine access is requested from
 - Many UCON concepts have been adopted in XACML 3.0
- 

Attribute-based Access Control (ABAC)

- *A logical access control methodology where authorisation to perform a set of operations is determined by evaluating attributes associated with the subject, object, requested operations, and, in some cases, environment conditions against policy, rules, or relationships that describe the allowable operations for a given set of attributes.*
 - This is a generic definition of access control that no gives no special place to the user or to the user's role, reflecting how the use of IT systems has changed over time
 - If 'access control' is reserved for the operating system (and other infrastructures) and operations, ABAC can be viewed as a synonym for application-level access control
- 

Granting & Delegation

- **Delegation** and **granting** of access rights both capture that a principal, or a subject, gets an access right from someone else
 - The research literature and the trade literature do not have firm definitions for those terms
 - **Granting** tends to be used in a generic sense; granted access rights often refer to the current access rights of a subject
 - **Delegation** sometimes stands more narrowly for short-lived access rights
 - XACML distinguishes between policy administration and dynamic delegation that permits some users to create policies of limited duration to delegate certain capabilities to others
- 

Delegation & Revocation

- A second distinction applies **delegation** to the granting of access rights held by the delegator, while granting access also includes situations where a managing principal assigns access rights to others but is not permitted to exercise those rights itself
- Rights may not be granted in perpetuity
 - The grantor may set an **expiry data** on the delegation
 - A right may be valid only for the current session
 - There may be a **revocation** mechanism such as the **Online Certificate Status Protocol (OCSP)** for X.509 certificates
- OCSP is supported by all major browsers




Authentication & Authorisation in Distributed Systems


- SAML 2.0 (2005 ≈ web applications)
 - **Single-sign** on meta-protocol; an **asserting party** authenticates the user and supplies a **relying party** with assertions about that user
- OAuth 2.0 (2012 ≈ apps)
 - An http-based authorisation protocol where a user (resource owner) authorizes a client (an app) to access a resource
 - User issues an **authorization grant** to the client, presented to an **authorisation server** to obtain an **access token**
 - Client presents token to a **resource server** to access the resource
 - Use case: personal data in a social network
- OpenID Connect (2014)
 - Adds user authentication to OAuth 2.0



Accountability

- *Security goal that generates the requirement for actions of an entity to be traced uniquely to that entity. This supports non-repudiation, deterrence, fault isolation, intrusion detection and prevention, and after-action recovery and legal action*
 - Supports processes that are launched after events had occurred
 - Regular audit that checks whether an organisation complies with existing regulations
 - Technical audit that scans logs in search for signs of a cyber attack
 - Investigation triggered by an incident that tries to identify the vulnerabilities exploited
 - Investigation that tries to identify the parties responsible
- 

Accountability

- These processes make use of **event logs**, kept by the operating system, by networking devices, or by applications
 - **Audit policies** define which events will be logged
 - Attackers may hide their traces by deleting incriminating log entries once they have acquired sufficient privileges but should not be able to tamper with the evidence already logged
 - **Tamper resistance of a log** could rely on physical measures like writing the log to WORM (write-once read-many) memory or on cryptography (**hash chain**)
 - Privacy rules can have an impact on the events that may be logged; logging may have unintended privacy impacts
- 

Conclusions

- **Authentication** is always done for a purpose
 - The purpose can be access control or **accountability**
 - **Authorisation** is an overloaded term
 - Setting the rules in a policy and applying those rules to an access request are both called ‘authorisation’
 - Authentication, authorisation and accountability come together in **access control**
 - Access control has grown from its roots in the **multi-user systems of the 1960s** to a **multi-faceted** field as usage of IT has diversified
 - It is tempting to call access control in a new field by a new name, but the fundamental principles have remained the same
- 