Paweł Rajba
pawel@ii.uni.wroc.pl
http://www.itcourses.eu/

Security basics

Agenda

- Introduction
- Authentication, authorization, access control
- Identity & Access Management
- Authentication: common schemes
  - HTTP Basic
  - NTLM
  - Kerberos
  - Forms-based
- How to ensure good security?

Introduction

- Application Security
  - From Wikipedia
    Application security encompasses measures taken throughout the code's life-cycle to prevent gaps in the security policy of an application or the underlying system (vulnerabilities) through flaws in the design, development, deployment, upgrade, or maintenance of the application.
  - Part of non-functional requirements
  - Important to be taken into account from begining
  - Common problem: usually security is underestimated

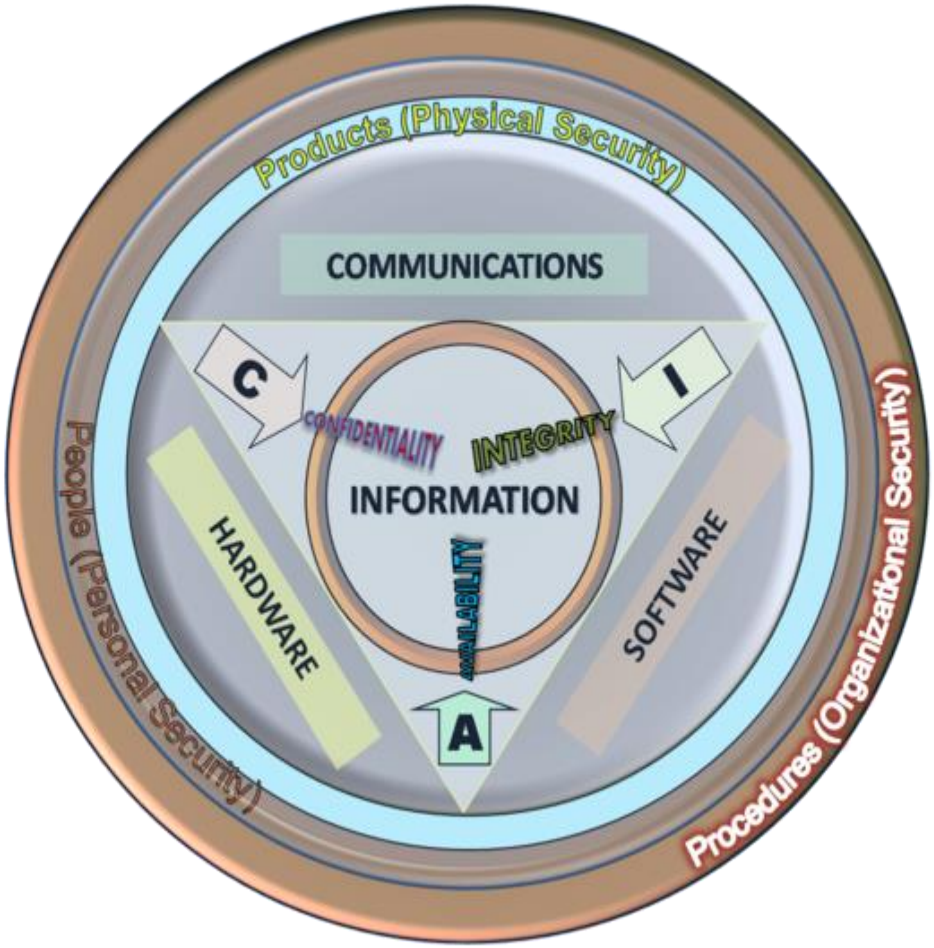
Introduction

- Information security
  - From Wikipedia:
    Information security, *sometimes shortened to InfoSec, is the practice of defending information from unauthorized access, use, disclosure, disruption, modification, perusal, inspection, recording or destruction. It is a general term that can be used regardless of the form the data may take (electronic, physical, etc…)*

Introduction

- Information security basic concepts
  - Confidentiality
    - Preventing discloure information
  - Integrity
    - Consistency of data over its entire life-cycle
  - Availability
    - Information must be available when needed
  - Authenticity
    - Ensure that the data, transactions or documents are genuine
  - Non-repudiation
    - Ensure involved party can’t deny his or her participation in activity

Introduction



Introduction

- Application security basic terms
  - Asset
  - Threat
  - Vulnerability
  - Attack
  - Countermeasure

Introduction

- Other concepts overview
  - Risk management
  - Controls (or countermeasures)
  - Security classification for information
  - Access control
  - Business Continuity

Introduction

- Most important security organizations
  - OWASP
    - https://www.owasp.org/
  - WASC
    - http://www.webappsec.org/
  - SANS Institute
    - http://www.sans.org/
  - ISACA
    - https://www.isaca.org/
  - ISC2
    - https://www.isc2.org/
  - NIST
    - http://csrc.nist.gov/

Authentication & authorization

- Identification
- Authentication
  - Identification + proof
  - Examples
    - Username + password
    - Security token
    - Smart card + PIN
    - Biometrics
- Authorization
  - Examples
    - File permissions
    - Encryption (only privileged get the key)
    - Boarding pass
    - Driver licence

Authentication

- Types of proofs
  - Something..
    - you know
    - you have
    - you are
- Type of authentication
  - Single factor
  - Dual-, multi-factor
    - E.g. smartcard + PIN

Authentication

- Centralized vs. Decentralized (federations)
  - SAML2
  - OpenID Connect
- SSO
  - Concept
  - Protocols supporting SSO
    - XTACACS, TACACS+, Kerberos, SAML2, WS-Trust, WS-Federation, OpenID Connect

Authentication: threats

- Smart cards (sth you have)
  - Steal card
  - Hack an issuer of cards
- One-time passwords (sth you have)
  - We consider both
    - Synchronic (generators on both sides)
    - Asynchronic (challenge-response protocol)
  - Again, steal device, hack device
  - Find a initial value for generator
    - Through hacking an issuer server

Authentication: threats

- Biometrics (sth you are)
  - Retina scan, finger print, voice recognition, signature recognition
  - Main problem: biometrics accuracy
    - False Rejection Rate (FRR) – false negative
    - False Acceptance Rate (FAR) – false positive
  - Accuracy problem implies that one may pretend by getting e.g. victims fingerprints
  - Accuracy ranking
    - retina > fingerprint > signature > voice

Authentication: threats

- Password (sth you know)
  - Attacker may see or record when one is typing
  - Keyloggers
  - Sniffing (e.g. local network)
  - Phishing
  - Dictionary and brute force attack
  - Social attack
  - Re-use attack
    - E.g. the same password in different places

Access Control System

- Combining AuthN & AuthZ with additional rules
- Examples
  - Rules on passwords (complexity, regular changes, history)
  - Object owner is able to determine object perms
  - Object owner is able to define object perms
  - Access denied by default
  - User ID can't be transferred
    - E.g. give to a new employee a login a someone fired

Access Control Models

- Discretionary Access Control
  - Owner of an object is able to decide who is allowed to access it
  - Common example: file system ACL
- Mandatory Access Control
  - Access rules defined centrally
  - Hard to manage, but offers higher security
  - Usually based on hierarchical sensitive labels
  - Two methods for applying MAC usually
    - Rule-based
    - Lattice-based (for more complicated scenarios)

Access Control Models

- Role-based access control
  - Based on roles/groups
  - Roles are usually organized in a hierarchy
  - Roles are controlled centrally
    - MAC model is intended for only read and write
    - Roles are considered as set of permissions and give more flexibility
  - A lot of systems implement RBAC
- Attribute-based access control
  - Not based on rights assigned to subject
  - Based on attributes which are used to prove the truth of statements (i.e. claims)
  - Example:
    - Claim: „older than 18"
    - Anyone, who can prove that statement, has granted access

Source: http://en.wikipedia.org/wiki/Computer_access_control

OAuth2 & delegated authorization

- Let's imagine the following scenario
  - You have an account on Google
  - You found a very fancy calendar application on your phone market
  - You want to use it, but don't want to give the application permission to all Google account data (e.g. mails, contacts, etc. – only calendar entries)
- In this scenario we consider 3rd party application which is considered as untrusted
  - And this is the place when the OAuth2 helps

OAuth2 & delegated authorization

- Actors
  - Resource server
    - Service which is protected and understands tokens
  - Resource owner
    - User
  - Client
    - 3rd party application
  - Authorization server
    - The one who issues tokens

OAuth2 & delegated authorization

- Client types and profiles
  - Protocol emphasizes 3 types of clients
    - Server-side web application
    - Client-side application running in a web browser
    - Native application

OAuth2 & delegated authorization

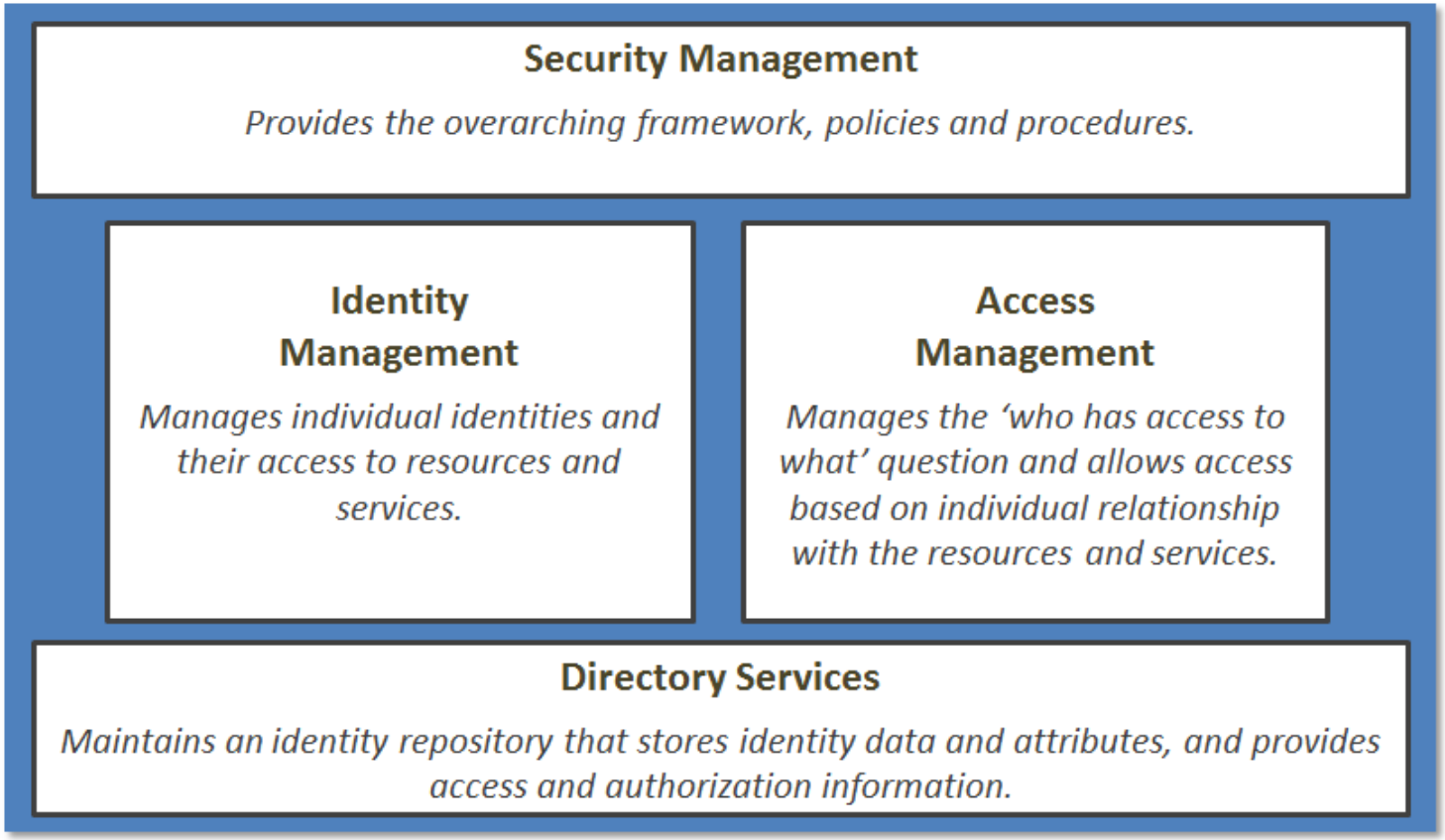
- Authorization flows
  - Authorization Code Flow
  - Implicit Flow
  - Resource Owner Credential Flow
  - Client Credential Flow

Identity and Access Management

- Access Management
  - The process of granting authorized users the right to use a service, while preventing access to non-authorized users.
- Identity Management
  - The process of managing identities in the organization.
  - Usually is supported by tools like
    - Directory Services
    - Federation Service
    - …
- Very often there is IAM unit in organization which combine both areas
- Good articles:
  - http://www.cio.gov.bc.ca/local/cio/informationsecurity/policy/summaries/26_access_cntrl_mngmnt.pdf
  - http://www.cio.gov.bc.ca/local/cio/informationsecurity/policy/summaries/26_access_cntrl_mngmnt.pdf

Identity and Access Management

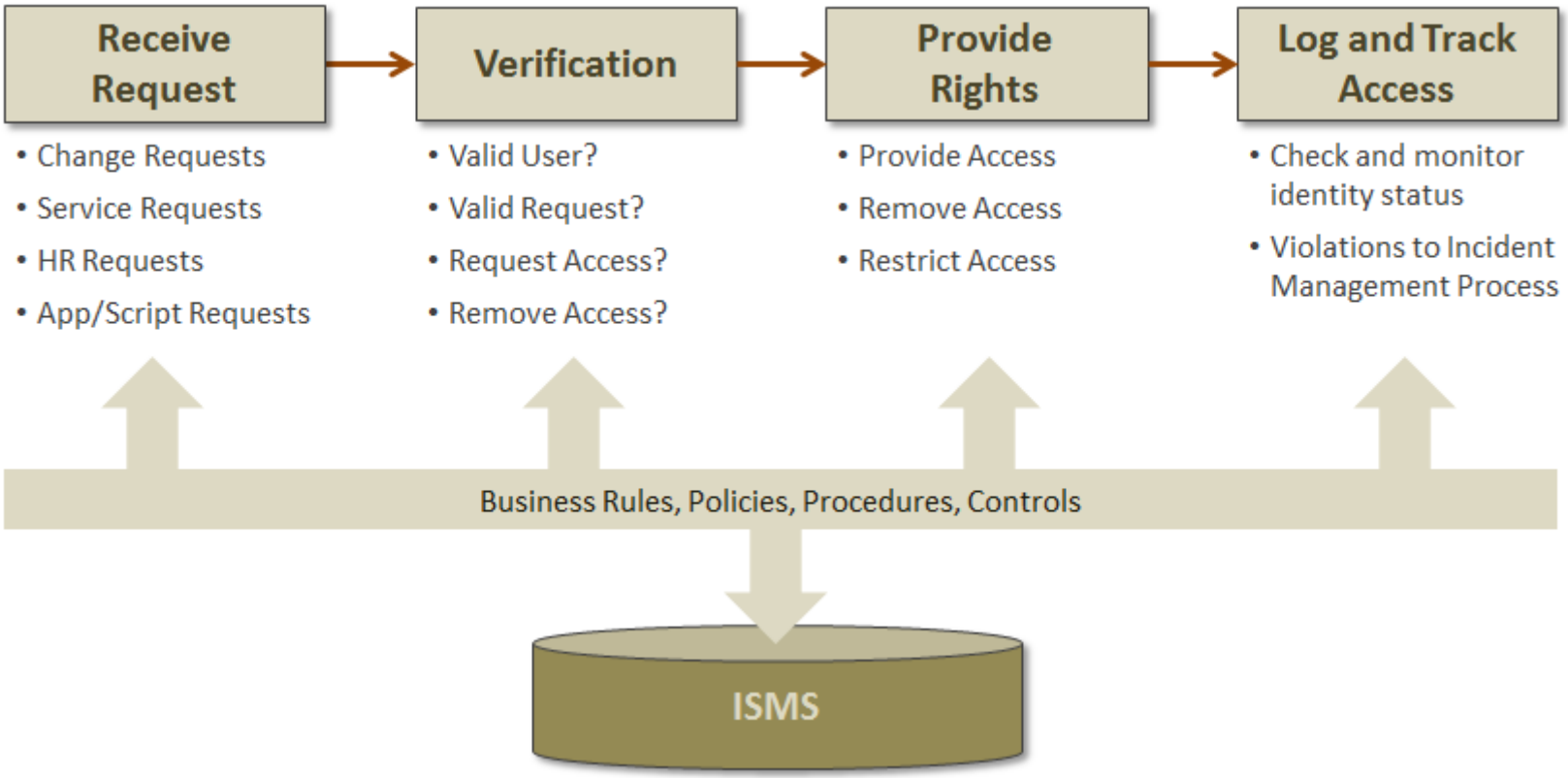
- Pieces together



Source: http://blogs.interfacett.com/access-management-control-it-just-tools

Identity and Access Management

- As a process



Source: http://blogs.interfacett.com/access-management-control-it-just-tools

Authentication: common schemes

- HTTP Basic
- NTLM
- Kerberos
- Forms-based

HTTP Basic

- A client sends a request to a protected resource
- A server answers with 401 HTTP status
  - Additionaly a Realm (area description) is attached
- In the client's browser usually a prompt for a login and password pops up
  - With every subsequent request a new header is attached

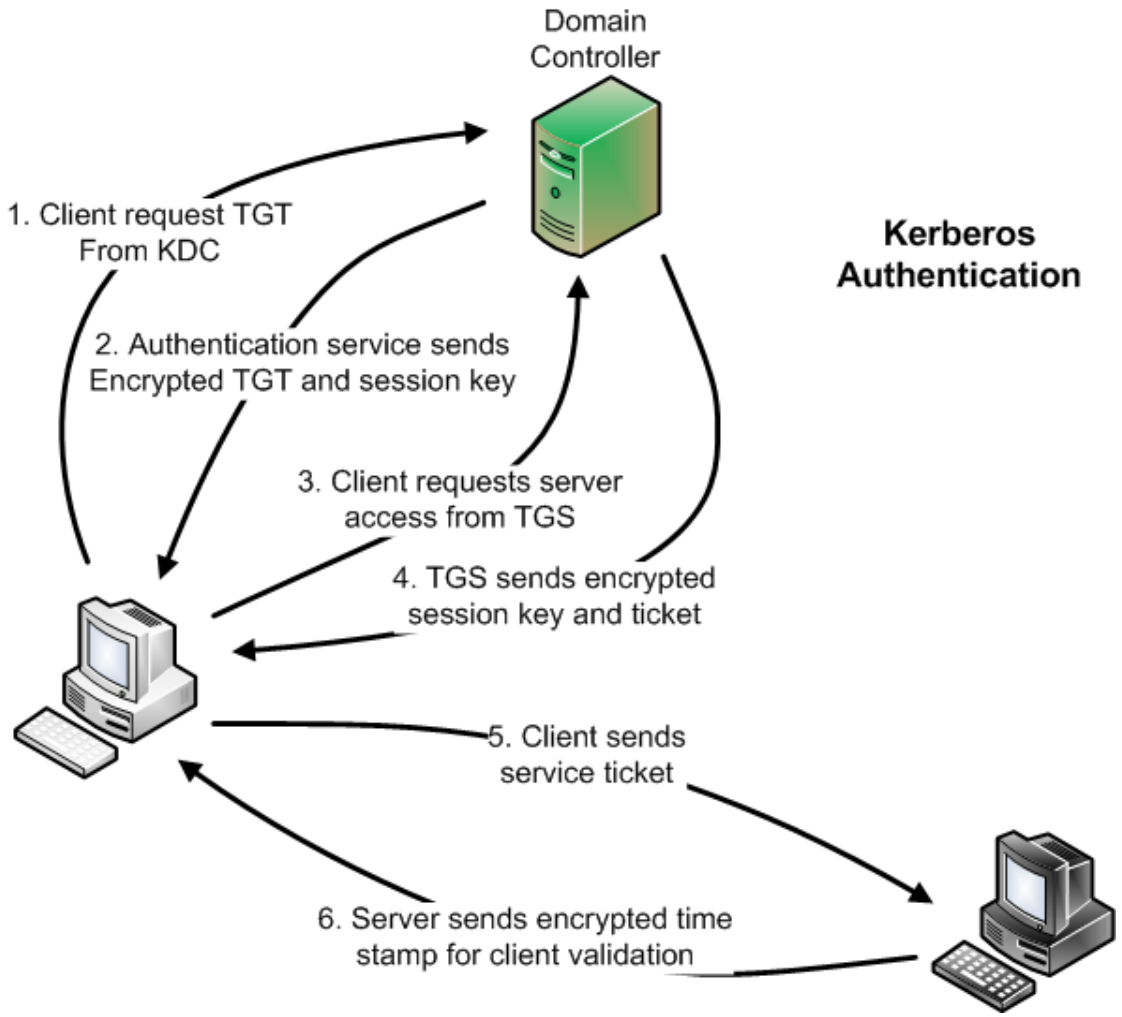
    `Authorization: Basic QWxhZGRpbjpvcGVuIHNlc2FtZQ==`
    - In data login:password sequence is encoded using Base64 algorithm
- After providing correct credentials the client is able access to the resource on the server

NTLM

- A challenge response protocol
- Handshake
  - 1: C → S
    - GET …
  - 2: C ← S
    - 401 Unauthorized
      WWW-Authenticate: NTLM
  - 3: C→ S
    - GET …
      Authorization: NTLM <base64-encoded type-1-message>
  - 4: C ← S
    - 401 Unauthorized
      WWW-Authenticate: NTLM <base64-encoded type-2-message>
  - 5: C → S
    - GET …
      Authorization: NTLM <base64-encoded type-3-message>
  - 6: C ← S
    - 200 Ok

Source: http://www.innovation.ch/personal/ronald/ntlm.html

Kerberos



Source: http://digital-forensics.sans.org/blog/2012/09/18/protecting-privileged-domain-accounts-network-authentication-in-depth

Forms authentication

- Forms authentication
  - Based on login form and authentication cookie
  - Authentication cookie has several parameters
    - Protection: None | All | Encryption | Validation
  - Meaning of
    - MembershipProvider
    - RoleProvider

How to ensure good security?

- What is good security?
  - Risk analysis, security is not a feature
- Security Development Lifecycle
  - Trainings of the team
  - Security Requirements
  - Threat modeling
  - Security Zones, Information flows, Software Security, Identity & Access Management, etc.
  - Secure coding
  - Verification including pentesting
  - Deployment and follow up

How to ensure good security?

- Security Architecture
  - Business Driven
  - Traceability of requirements
    - From the business to the technical and other way around
  - Defence in depth
    - Deterence, Prevention, Containment, Detection, Recovery
  - Other principles
    - The weakest link, Need-to-know, Separation of duties, Least privilege