TU Wien
Institut für Rechnergestützte Automation
Forschungsgruppe Industrial Software
Arbeitsgruppe ESSE
`http://security.inso.tuwien.ac.at/`

Report Lab1

IT Security in Large IT Infrastructures

183.633 - SS 2013

05.06.2013

Group 5

Name	**MatrNr.**	**Email address**
Michael Heil	0826358	e0826358@student.tuwien.ac.at
Lukas Puschmann	0825354	e0825354@student.tuwien.ac.at
Dominik Amon	1228536	e1228536@student.tuwien.ac.at

Contents

1 lab1a - Safety Objectives

Security is the core requirement to this software system. In this chapter we will list and explain the safety objectives of the system as a whole as well as its implications to its separate components. The order of the targets has no impact on their importance.

1.1 Authorization

Despite the fact that authorization represents the basic functionality of this system, this goal is also an important non-functional issue concerning the communication of the different components themselves. Consequently, all components, especially those outside of immediate control, but also those within system boundaries are subject to strict authorization.

A strategy to achieve this goal is to follow the *complete mediation* design principle which states that each and every request needs to be verified instead of keeping a cached authorization result. As such, the performance might suffer a bit but regarding security this is a must.

1.2 Authenticity

Similar to *authorization*, authenticity is important for intra- and inter-communication of all components. To ensure authenticity the *complete mediation* principle provides support to a comprehensive certificate-based authentication system. These certificates are managed and replaced at regular intervals.

The most vulnerable spot are the external interfaces of the *authorization and object management* as an intruder could get a) information about the access rights (confidentiality) and b) change any of these data (integrity). Therefore it is important to impose a strict authentication and autorization policy here.

Functional authentication will basically be realized using a RFID-chip each employee or person who wishes to gain access has to carry. In normal situations the person to be authorized has to put the chip onto or in front of a reader and provide his six digit pin (which has to be changed every six months). There will also be the possibility to lock a chip if lost similar to a credit card. For critical secured objects an additional iris-scan will be conducted and compared to the data stored on the chip. More information is to follow in a later chapter.

Generally no component in the system must ever trust any component from beyond the border. The possibility is always there that it was replaced or reconfigured to act in a malicious way. The same goes for *secured objects* as it is much easier to fake a terminal than this component.

1.3 Integrity

Data integrity is partially guaranteed by the rigid authentication and authorization mechanisms present within and around the system. Naturally this goal plays an important role wherever data is stored (which is the *auditing* and *authorization and object management* component in this system). Checksums allow the detection of illegitimate changes to these data. As the authorization storage is held redundantly (see later) changes to single instances can easily be detected and reported.

The most important card against alteration of communication data is the application of encrypted protocols everywhere, even within secured boundaries.

1.4 Confidentiality

As with integrity, the fact that all communication happens in an encrypted way ensures the important confidentiality goal. Additional measures to avoid unauthorized access are the already mentioned authentication mechanism of the componentes themselves, the *complete mediation* principle as well as the authentication rules for the users of the *authorization and object management* subsystem.

1.5 Non-Repudiation

The goal of non-repudiation is directly linked to data integrity of the auditing system. As this subsystem is responsible for tracking all authorization requests and the respective results it is clearly replicable who wanted to access what and when.

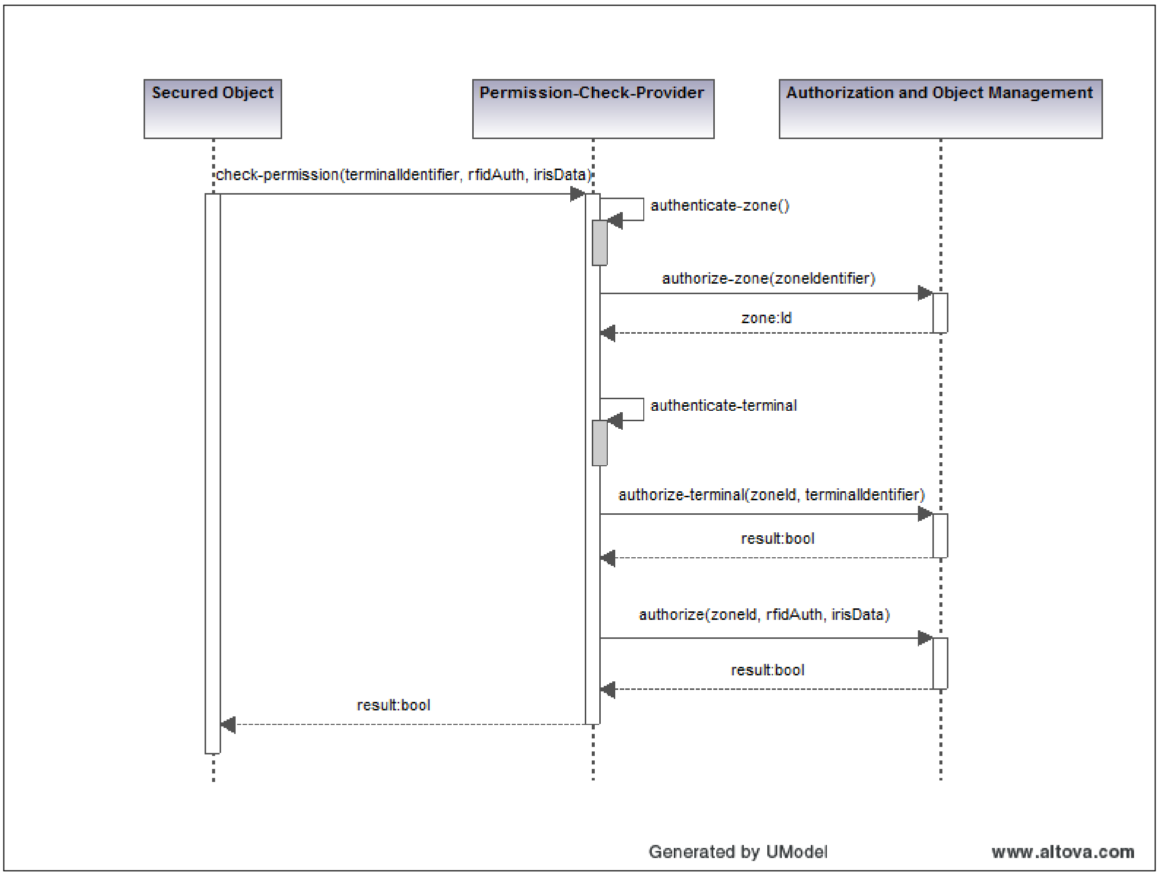
As concerns the *authorization and object management* it will be logged locally which changes were made by whom and when in order to ensure this goal. The log shall be written in a way that it is possible to visiualize the full change-history of a certain record.

1.6 Availability

Availability is an important safety objective as a system unavailability requires a fallback to another authorization system (if any at all) which will certainly mean safety and security detriments. For this sake the core components shall be operated redundantly.

The *Permission-Check-Provider* will be grouped in a cluster with one cluster manager and several slaves which carry out the load assigned by the manager. A similar configuration is intended for the *authorization and object management* which is additionally arranged in different shards (horizontal partitions) to increase performance.

With reduced effort but still to remain (practically) immune to loosing the non-repudation-property also the *auditing* subsystem will have a active/passive redundancy fallback mechanism in case of a failure within the primary unit. The similar principle will be applied to the *secured object* subsystem of each object and the *notification system* (as it is only there to forward information and thus error-prone to a lesser extend). Eventually there must be at least one terminal per zone. The same requirements apply to external *terminal* and *secured object* subsystems.


Secured Object
Permission-Check-Provider
Authorization and Object Management
check-permission(terminalIdentifier, rfidAuth, irisData)
authenticate-zone()
authorize-zone(zoneIdentifier)
zone:Id
authenticate-terminal
authorize-terminal(zoneId, terminalIdentifier)
result:bool
authorize(zoneId, rfidAuth, irisData)
result:bool
result:bool
Generated by UModel
www.altova.com

Overall availability will also be supported by an active supervisory system to detect failures, emergency backup plans to reduce downtime and regular health-checks conducted on all internal components to prevent failure at all, at least in certain cases.

2 lab1b - Interface Definitions

Tarpit sodass man nicht Brute-forcen kann

Authorization and Object Management hat die Zertifikate der Secured Objects und eventuell auch Terminals (zur Authentifizierung)

Was muss dem Gesetz nach gelogged werden?

3 lab1c - Security Analysis