D3.1 - e-Security Context and Threat Analysis

Deliverable ID	**D3.1**
Deliverable Title	**e-Security Context and Threat Analysis**
Work Package	**WP3.1**
Dissemination Level	**PU**
Version	**1.0**
Date	**2018-10-31**
Status	**Submitted**
Lead Editor	**ISMB**
Main Contributors	**BQ GAPES CNET**

Published by the Consortium

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776261.

Document History

Version	Date	Author(s)	Description
0.1	2018-08-03	ISMB	TOC
1.0	2018-10-31	ISMB	Ready for submission

Table of Contents

1 Introduction

This document represents a preliminary analysis of the proposed scenario in terms of security risks, while D3.5 will be an updated version with the focus on the countermeasures needed to protect the sensitive features as well as data exchanged using the platform respecting the privacy of its users.

1.1 Scope

The entire end-to-end communication chain is considered: this includes the hardware bus between the Galileo receivers and the chipset of the mobile systems, the information management within the mobile Operating Systems (including internal interaction within kernel space, mobile OS framework and applications), and the communication channel between the mobile and cloud services.
Concerning the Galileo-related threats, a deep analysis has been carried out considering both intentional and unintentional threats as jamming, meaconing and spoofing. From the analysis point of view, also the Galileo Commercial Service is considered.
A deep analysis has been done in term of security risks, in alignment with on-going e-security developments undergoing in the EU. It includes:

- auditing, assessment, and evaluation of the most critical security (and privacy) sensitive aspects in the scenario envisioned for the project;
- threat modelling and analysis;
- analysis of security countermeasures, also looking at user-related aspects whereas relevant.

The security context of each use case is mapped to the security risks it presents and thus with the associated required e-security features and countermeasures.

D3.1 only focuses on Threat Analysis and misuse case modelling.
D3.5 will also extend the analysis to countermeasures and definition of adaptive e-security features – as well as mapping the security level required by each use case with the appropriate set of e-security solutions.

1.2 Related documents

ID	Title	Reference	Version	Date
[RD.1]	S. Pullen, G.X. Gao, 'GNSS Jamming in the name of Privacy,'	Inside GNSS, Vol. 7, No. 2,.		March/April 2012
[RD.2]	F. Dovis, GNSS Interference Threats and Countermeasures	Artech House, Norwood, MA		2015
[RD.3]	D. Margaria, B. Motella, M. Anghileri, J. J. Floch, I. Fernandez-Hernandez and M. Paonni, "Signal Structure-Based Authentication for Civil GNSSs: Recent Solutions and Perspectives,"	IEEE Signal Processing Magazine, vol. 34, no. 5, pp. 27-37, Sept. 2017. doi: 10.1109/MSP.2017.2715898.		September 2017
[RD.4]	M. L. Psiaki and T. E. Humphreys, "GNSS spoofing and detection"	Proc. IEEE, vol. 104, no. 6, pp. 1258–1270		June 2016

2 Threats Analysis

The threats analysis refers to three relevant blocks:

- The trusted devices
- The trusted platform
- The applications

2.1 Trusted devices

This section deals with the threats analysis for the trusted devices, but it is extended to the "GNSS infrastructure and services" in order to take into account threats affecting the GNSS Signal In Space (SIS), namely spoofing, meaconing and jamming.

2.1.1 GNSS SIS Threats

This section provides an accurate analysis of the possible types of GNSS spoofing attacks, with the aim of selecting those attacks that have **a significant level of associated risk, in relation to applications**.

This analysis has three steps:

1. performing a *critical review* of different types of radio frequency (RF) attacks, as published in recent scientific literature (section 2.1.1.1);
2. *assessing the likelihood* associated to single attacks, taking into consideration both the *cost* at attacker side and the relation to LBS *applications* (section 2.1.1.2);
3. *Identifying the "most likely" spoofing attacks* for the applications of interest: these kinds of attacks will be those worth to be implemented in the test phase (section 2.1.1.3).

2.1.1.1 *Critical review of the spoofing attacks*

In general, attacks of intentional nature are classified in three different forms [RD.1][RD.2][RD.3].

1. **Jamming**. Blocking of the reception of GNSS signals by deliberately emitting electromagnetic radiations (i.e., radio-frequency interference) to disrupt user receivers by reducing the signal-to-noise level;
2. **Meaconing**. Rebroadcasting of delayed GNSS signals without any distinction between SIS from different satellites;
3. **Spoofing**. Transmission of counterfeit GNSS-like signals, with the intent to produce a false position/time within the victim receiver, without disrupting GNSS operations.

The current analysis focuses on meaconing and spoofing, also known in the literature as *structured interference,* since they are based on the intentional transmission of delayed or counterfeit GNSS-like signals. We also consider modified versions of conventional attacks and hybrid/combined strategies mentioned in the literature.

The definition of the main forms of attack is summarized in the following list, entirely based on [RD.3] and references therein:

- **Meaconing**: reception and rebroadcasting of an entire block of RF spectrum containing an ensemble of received GNSS signals, without distinction between different satellite signals.
- **Meaconing with variable delay**: it is a modified version of the classical meaconing. It has the scope of controlling the delay introduced by the meaconer and fooling potential countermeasures based on the monitoring of the clock drift.
- **Meaconing with modem** (relaying attack or worm-hole attack): this is a type of meaconing in which the receiver is connected to a remote antenna (via a real-time radio link) located at the pretended position.
- **Simplistic spoofing**: it is able to generate counterfeit GNSS signals, not necessarily reflecting any information on the current broadcast signals. It can be put in practice by using:
  - low-cost hardware (HW) for receiving and replaying the GNSS signals + customized open-source signal simulators/synthesizers to control/modify the signals parameters;

  - commercial HW simulators, normally expensive and moderately complex to use.
- **Intermediate spoofing**: the spoofer synchronously generates counterfeit signals, trying to attack simultaneously each tracking channel of the target receiver, by first performing the code-phase alignment between false and genuine received signals.
- **Intermediate self-spoofing** (cooperative or limpet spoofing): refers to the case in which a complicit victim user directly performs an intermediate attack.
- ***SCER - security code estimation and replay***: the spoofer receives the genuine signal, estimates some information on it, and uses that to generate a spoofing signal, generally with a delay. Such an attack:
  - can applied to a signal with cryptographic defences including unpredictable security codes;
  - attempts to estimate (and not only to predict) each signal's unpredictable security code chips (or navigation data bits) on the fly.
- ***Meaconing/spoofing with high gain antennas***: based on the use of antennas with enough gain to directly separate single GNSS signal components from the noise, including, for example, unknown or encrypted code chips of restricted-access signals.
- ***Nulling attack***: This is an advanced spoofing technique. The spoofer also transmits the negative of the true signals (i.e., with same power but opposite carrier phase) that, in this way, results canceled at the victim receiver side.
- ***Sophisticated spoofing***: This needs a set of coordinated and synchronized spoofers, able to attack the victim receiver in an organized way. Such coordinated spoofers are able to generate and transmit counterfeit signals as in the case of intermediate spoofing. Different forms of sophisticated attacks are described in [RD.3].

Details on how the mentioned forms of attack are conducted can be found in [RD.3] and references therein.

2.1.1.2 *Assessment of the likelihood and impact on LBS applications*

Based on the list of attacks presented above, this section analyses the likelihood of each threat in relation to commercial LBS applications. Such an analysis is mandatory to filter out those attacks whose realization can be considered unlikely, leading to a low level of associated risk.

The approach allows for the selection of the attacks to be included in the lab-tests.

The **selection of the attacks** is based on two criteria, as conceptually represented in Figure 1:

1. The **relevance for the target applications**. By defining the technical requirements of the target application [ref], it is possible to assess whether or not the conditions needed for the realization of each attack are compatible with the expected application scenarios. In other words, if the specific hypotheses required to implement the attack are difficult to be satisfied in the specific application, then the attack will be considered unlikely (e.g.: complexity of the setup, practical constraints, etc.).
   For example, in the case of GNSS-based LBS applications, it might be difficult for the spoofer to reach physically the victim receiver antenna. Consequently, an attack based on a wired connection between the spoofing device and the victim receiver is considered impractical. On the other hand, it is considered generally feasible to implement the attack via radio-frequency channel.
   Attacks not relevant for the target applications will be filtered out.

2. The **complexity/cost at the attacker side**. Depending on the complexity required to implement the attack, a cost can be associated to its realization. The term "cost" must be interpreted at the attacker side and refers to the definition given by authors of [RD.3] and [RD.4]. It includes three cost items, i.e.:
   a. the cost of developing or buying the hardware;
   b. the expertise required to set it up and run it;
   c. the complexity of operating it.

   Table 1 offers an estimate of the costs components associated to the attacks listed in section 2.1.1.1, as assessed in [RD.3].
   Attacks with a high associated cost are considered unlikely and will be filtered out.

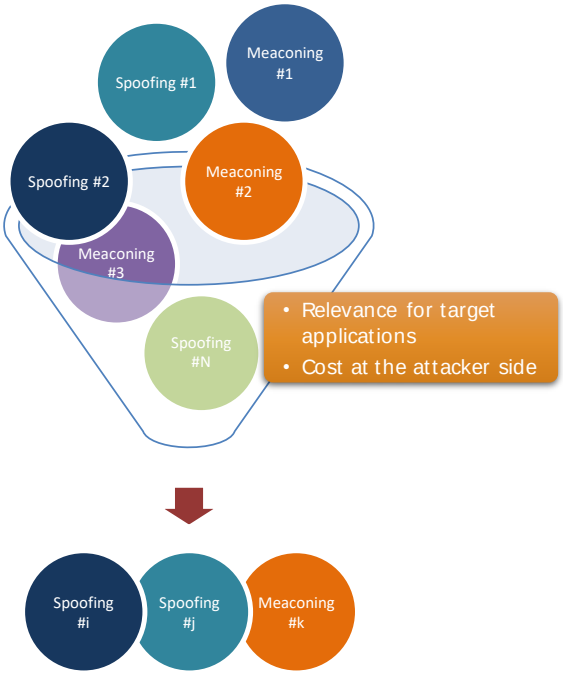

Figure 1 – Conceptual scheme of the selection of spoofing attacks, performed based on the associated cost and relevance for the target applications.

	Cost at the attacker side		
Attacks	**Developing or buying the HW**	**Required expertise**	**Complexity of operation**
Jamming	VL	L	VL
Meaconing	VL	M	VL
Meaconing with variable delay	L	M	L
Meaconing with modem	L	M	M
Simplistic spoofing (custom low-cost HW)	L	H	M
Simplistic spoofing (HW simulator)	H	M	M
Intermediate self-spoofing	M	H	M
Intermediate spoofing	M	H	H
SCER – security code estimation and replay	M	VH	H
Meaconing/spoofing with high gain antennas	VH	H	H
Nulling attack	M	VH	VH
Sophisticated spoofing	VH	VH	VH

VH: very high – H: high – M: medium – L: low – VL: very low

Table 1 – List of attacks and associated costs.

2.1.1.3 *Identification of the most likely attacks*

Because of the two criteria, the selection of attacks is presented and motivated in Table 2. **Attacks not relevant for the target applications and/or with a non-affordable associated cost[1] are filtered out.** On the contrary, **the selected attacks are characterized by affordable costs and good relevance for the target applications**.

[1] A cost is *qualitatively* considered "non-affordable" here when the gross price of the necessary technical equipment and skills appears to overcome the expected economic revenue of the spoofing operation.

Attacks	Selected/ filtered out	Motivations
Jamming	✓	
Meaconing	✓	Victim receivers under a meaconing attack will output manipulated position and velocity and a time in arrears of true time. Generally, it is used to attack the position.
Meaconing with variable delay	✓	Different from simple meaconing, in the case of meaconing with variable delay the meaconer can increase the delay at a rate that is consistent with the clock drift of the target receiver and then gradually impose a significant timing delay. This attack is in fact considered a specific menace for applications based on the time information, fooling possible implemented countermeasures based for example on the monitoring of the clock drift.
Meaconing with modem	✘	This is a modified version of the classical meaconing. In addition to the motivations reported above (meaconing attack), there is also the fact that it may be logistically complex to implement and then not.
Simplistic spoofing (custom low-cost HW)	✘	In the case of simplistic spoofing, the counterfeit GNSS signals are not necessarily consistent with the current broadcast signals, thus allowing the detection with simple countermeasures. This type of attacks can be put in practice by using a commercial HW simulator, which is normally expensive and moderately complex to use. In addition, it requires to have access to the victim receiver antenna, which might be not feasible (or very difficult) in the case of timing applications.
Simplistic spoofing (HW simulator)	✘	
Intermediate self-spoofing	✓	Self-spoofing is considered likely for the target LBS applications. This form of attack is very relevant, where users are motivated to falsify their positions to get economic advantages (e.g.: road tolling, pay-as-you drive, fishing in restricted/prohibited areas, etc.).
Intermediate spoofing	✓	
SCER – security code estimation and replay	✓	
Meaconing/ spoofing with high gain antennas	✘	The elevated costs affect the attack likelihood and, consequently, the associated risk.
Nulling attack	✘	
Sophisticated spoofing	✘	

Table 2 – List of selected and not-selected attacks.

2.1.2 Device Related Treats

Most of the previously described threats have an impact on the devices performance, both at HW and OS level. In addition to the potential attacks “on the air” described in the previous section, there are several applications in Android, that are very well-know among the users community. These allow using fake positions and simulating the presence in unreal places, what means that malicious users could get access to some benefits that they should not have.

These applications are quite widespread and are available free in the main apps markets, both for Android and iOS devices. As an example, some of the most used apps in Android are:

- Fake GPS Location Spoofer Free → LINK
- Fake GPS → LINK

The position selected in any of these apps is going to replace the original signal calculated by the smartphone through the GNSS system. That means that any other app installed is going to consider the new position as valid and therefore the user could easily cheat the GOEASY system if no countermeasures are introduced to mitigate this.

As a first approach, a couple of measures have been described in order to mitigate the impact of users sending fake positions to the GOEASY apps.

The first one includes some checks at OS level that are related to the denial of some permissions that allow the smartphone to authenticate fake positions as real. These permissions are:

- Root or superuser permissions
- Bootloader unlocked
- "Allow fake positions" option in the Android developers options menu

The second one is related to a potential double check of the position between the GNSS system and the network, considering as trusted positions only those for which GNSS position is a subset of the network position (that is less accurate than the GNSS position). This means that if the user uses any app to fake the position, this position will not match with the location given by the network and therefore will be rejected by the GOEASY platform. Please refer to D2.2 Figures 13 and 14 for further reference about the proposed countermeasures.

There is also the possibility of using a VPN that could fake the network location and is being investigated how to mitigate the impact of both GNSS + network locations being modified by users.

2.2 Trusted platform

This section deals with the threats analysis for the trusted including the segment between the mobile and cloud services.

2.2.1 Stride Threat Analysis

This section provides a detailed threat analysis for each service based on the STRIDE Threat Model[2].

The STRIDE Threat Model provides the means to better analyze the potential threats for a system by grouping threats into 6 categories:

- **Spoofing**. An example of identity spoofing is unauthorized accessing and then using another user's authentication information, such as username and password.
- **Tampering**. Data tampering involves the malicious modification of data. Examples include unauthorized changes made to persistent data as well as the alteration of data as it flows between two computers over an open network, such as the Internet.
- **Repudiation**. Repudiation threats are associated with users who deny having performed an action without other parties being able to prove the contrary. For example, a user performs an illegal operation in a system that lacks the ability to trace the prohibited operations. Nonrepudiation refers to the ability of a system to counter repudiation threats. For example, a user who purchases an item might have to sign for the item upon receipt. The vendor can then use the signed receipt as evidence that the user did receive the package.
- **Information disclosure**. Information disclosure threats involve the exposure of information to individuals who are not supposed to have access to it - for example, the ability of users to read a file that they were not granted access to, or the ability of an intruder to read data in transit between two computers.

[2] https://msdn.microsoft.com/en-us/library/ee823878%28v=cs.20%29.aspx

- **Denial of service**. Denial of service (DoS) threat is the ability of denying service to valid users, e.g. by making a Web server temporarily unavailable or unusable.
- **Elevation of privileges**. In this type of threat, an unprivileged user gains privileged access and thereby has enough access rights to compromise or destroy the entire system. Elevation of privilege threats include those situations in which an attacker has effectively penetrated all system defenses and becomes part of the trusted system itself.

2.2.2 STRIDE threat analysis of GOEASY

An initial STRIDE analysis has been performed for the main GOEASY components that will be presented in the next paragraphs, providing deeper details on the analysis outcomes for each considered service.

2.2.2.1 Authentication and authorization services

2.2.2.1.1 Access Manager

The Access Manager (AM) is an implementation of a XACML policy decision point. It will be mainly implemented exploiting available open-source implementations such as AuthzForce[3] from the FIWARE catalogue[4]. It stores the policies of its PI and provides an interface for querying them.

The AM typically receives queries from the Security Enforcement Points (SEP) of the components placed at the boundaries of the platform's trusted area.

The relevant assets are policies and the decisions made by the AM.

Spoofing

An attacker could try to spoof the system by acting as a Platform Administrator (PA) operator.

Tampering

An attacker could try to change the network traffic to influence the outcome of the policy query. This can be done either by changing the query or by changing the result of the query. Moreover, an attacker could try to insert, delete or change policies.

Repudiation

If the process of creating and removing policies is not monitored it might lead to misuse. For example, it might be possible for an authorized attacker (a malicious operator) to insert a policy that enables the query of confidential information and subsequently remove that policy.

Information disclosure

Since the policies are not confidential there is no information disclosure threat.

Denial of service

An attacker could try to delete all policies. Additionally, an attacker could send multiple complex queries to increase the load on the AM. Finally, the attacker could try to modify the policy load that could be handled by the AM.

Elevation of privilege

An operator can give himself the rights to access all components of the platform.

2.2.2.1.2 Identity Manager

The Identity Manager (IM) provides Single Sign On for platform and federation users and manages user identification inside a single Platform Instance. It is mainly implemented exploiting available

[3] https://github.com/authzforce
[4] https://www.fiware.org/developers/catalogue/

open-source implementations such as KeyRock[5]. Such a tool is clearly critical in the platform and is subject to the following threats, classified according to the STRIDE methodology.

Spoofing

Cross-site request forgery (CSRF): is a web-based attack whereby HTTP requests are transmitted from a user that the web site trusts or has authenticated (e.g., via HTTP redirects or HTML forms). Any site that uses cookie-based authentication is vulnerable for these types of attacks.

Clickjacking: a malicious site loads the target site in a transparent overlay on top of a set of dummy buttons that are carefully constructed to be placed directly under important buttons on the target site. When a user clicks a visible button, he is actually clicking a button (such as an "Authorize" button) on the hidden page. An attacker can steal a user's authentication credentials and access its own resources.

Compromised Access Tokens: an attacker might try to compromise the IM access tokens to get unauthorized access.

Open redirect: An attacker could use the end-user authorization endpoint and the redirect URI parameter to abuse the authorization server as an open redirector. An open redirector is an endpoint using a parameter to automatically redirect a user agent to the location specified by the parameter value without any validation. An attacker could utilize a user's trust in an authorization server to launch a phishing attack.

Brute Force Attack: A brute force attack happens when an attacker is trying to guess a user's password.

Registration spoofing, i.e., tricking an operator into registering an attacker

Tampering

Typical tampering attacks exploit SQL Injection on the identity (user) database.

Repudiation

Clickjacking: a malicious site loads the target site in a transparent overlay on top of a set of dummy buttons that are carefully constructed to be placed directly under important buttons on the target site. When a user clicks a visible button, they are actually clicking a button (such as an "Authorize" button) on the hidden page. An attacker can steal a user's authentication credentials and access their resources. In such a way any action carried on the system is deniable and can be "mapped" onto another, real user (no way to trace back the real attacker)

Compromised Access Tokens: an attacker might try to compromise IM access tokens to get unauthorized access.

Information disclosure

Password database disclosure: While a human could probably never crack a hashed password, it is very possible that a computer could. The security community suggests around 20,000 hashing iterations to be done to each password. This number grows every year due to increasing computing power (It was 1000 almost 12 years ago).

Denial of Service

An attacker might attempt a DoS by submitting multiple-repeating authentication requests with the wrong credentials. This causes the system to block the user for which attempts are done, thus generating a denial of service for the actual user.

Elevation of Privileges

An attacker might exploit broad or light security token scopes to gain higher access privileges.

[5] https://catalogue-server.fiware.org/enablers/identity-management-keyrock

2.3 Applications

This section presents the analysis of the e-security aspects of ApesMobility and AsthmaWatch.

2.3.1 ApesMobility

This section provides an analysis of potential threats for the ApesMobility application, following the STRIDE Threat Model and the definitions introduced in the previous paragraphs.

- **Spoofing**. ApesMobility does not handle any identity information of the user, thus no major threats are relevant when it comes to unauthorized access. Nevertheless, a malicious attack to the GOEASY platform could grant access to time+location data of an anonymous user (being sent by the ApesMobillity app). If this access intercepts data before the GOEASY components perform additional anonymization treatments (such as noise around the points of departure and arrival of a user), this information could provide a basis for the identification of a specific house/location/identity.
- **Tampering**. A malicious modification of location data could imply the “unfair” attribution of points to a user. A tampering attack (e.g. carried out via an emulator) could provide fake positions (over time) to the ApesMobility application. The ApesMobility App and the GOEASY platform could interpret this data as a “rewardable green behavior” (e.g. the fake date simulate a bike trip) and erroneously assign rewarding points to the user.
- **Repudiation**. There are no major threats, as the only controversial issue resulting from a Repudiation relates to the “loss” of rewarding points of a user. Users who install ApesMobility will nevertheless have to accept Terms and Conditions, which will specify that the application might occasionally fail in detecting some “sustainable mobility” actions (because of lack of signal, or simply technical problems) and also that this information will be transient on the client side (i.e. the information is saved on the app, which could be deleted or lost along with the personal device). This implies that the user will not be entitled to claim lost points because of accidental or malicious events.
- **Information disclosure**. As ApesMobility has no identity information attached to the user, the only Information disclosure threats relate to the possibility that someone gains illegitimate access to the phone or finds a way to “intercept” information about a phone via a spoofing attack. In these cases the attacker could retrieve via ApesMobility a log of positions of the user (e.g. recorded journeys and check-ins) that the user has not cleared from the app.
- **Denial of service**. DoS threats could prevent the user from certifying sustainable mobility behaviours, thus not allowing him/her to collect rewarding points. This could happen for temporary unavailability of the GOEASY platform because of technical problems or overload.
- **Elevation of privileges**. No elevation of privileges is foreseen as there is no “identity” concept on the app. Furthermore, the anonymous information collected on public databases is public by nature.

2.3.2 AsthmaWatch

The purpose of the AsthmaWatch use case is to demonstrate the use of Galileo in a mass-market application. The AsthmaWatch use case will develop an infrastructure for fine-grained collection of air quality measurements using Galileo enabled stationary and mobile sensor gateways. The collected measurements are transferred to the GoEasy cloud and processed. They will then serve as the basis for one or several end-user apps that will help people with different type of lung and airway related disease to avoid problems in their daily lives.

There are two different set of e-security aspects for AsthmaWatch.

- Sensor measurements and data collection
- Usage and consumption of air quality data by end-user apps and services

Sensor Measurements and Data Collection

- People collecting air quality measurements in the city could try to fake their position. For instance being reimbursed for collecting data in the field while in fact being at home just sending fake positions and fake sensor data.
- Another scenario is someone trying to fake the real sensor value so that position is correct but the level of pollution is lowered
- A third scenario is to do correct measurements in one position and delay the sending so that it appears to have been done in another position, to make that position look less polluted than it is.
- Someone could try to extract logged data from the sensor platform and deduct something about the user's behaviour.

Usage of Air Quality Data

- Someone might try to interfere with the position of the users smart phone, to give the wrong position data to the AsthmaWatch app so that it makes erroneous suggestions regarding how healthy the environment is.
- An intruder might try to capture some privacy and health related data either from users phone or when being transmitted to or from the GOEASY Cloud.

3 Conclusions

In the present Deliverable, the security risks have been analysed considering the three main perspectives: the devices, the GOEASY platform and the GOEASY testbed applications.

In D.3.5 the security context of each use case will be mapped with the security risks, it presents, serving to associate the risks themselves to the required e-security features and countermeasures.

Acronyms

Acronym	Explanation
AM	Access Manager
CSRF	Cross-site request forgery
DoS	Denial of Service
IM	Identity Manager
PA	Platform Administrator
RF	Radio Frequency
SCER	Security Code Estimation and Replay
SEP	Security Enforcement Points
SIS	Signal In Space
STRIDE	model of threats: Spoofing of user identity Tampering Repudiation Information disclosure (privacy breach or data leak) Denial of service Elevation of privilege
URI	Uniform Resource Identifier
XACML	eXtensible Access Control Markup Language

List of figures

List of tables