Project number: 653454
H2020-DS-2014-1

Project start: October 1, 2015
Project duration: 3 years

CREDENTIAL

Secure Cloud Identity Wallet

D2.4

Vulnerability Catalogue

Document Identification	
Due date	September 30, 2016
Submission date	September 30, 2016
Revision	1.0

Related WP	WP2	**Dissemination Level**	PU
Lead Participant	AIT	**Lead Author**	Aleksandar Hudic
Contributing Beneficiaries	AIT, ATOS, FOKUS, OTE, TUG	**Related Deliverables**	D2.2, D2.3, D2.5, D2.6

Abstract: Identifying vulnerabilities is essential for risk assessment. This document presents a comprehensive list of vulnerabilities for identity and access management systems and data sharing platforms in general, and in particular for privacy-preserving instantiations. The catalogue can be seen as extension of existing (generic) cloud vulnerability catalogues.

This document is issued within the CREDENTIAL project. This project has received funding from the European Union's Horizon 2020 Programme under grant agreement no. 653454.

Executive Summary

On a high level, the central goal of the CREDENTIAL project is to develop a privacy-preserving data sharing platform (wallet) with integrated identity provider (IdP), which can be used to share authenticated data without the wallet learning any of the user's personal information. The functionality and added value of these services will be showcased by concrete pilots from the domains of eGovernment, eHealth, and eBusiness.

A central task in the development of security- and privacy-sensitive software applications is a continuous and comprehensive threat analysis and evaluation. This in turn requires a precise analysis of potential vulnerabilities, as only discovering and understanding potential weaknesses allows one to assess their potential impact and to take counter-measures.

In this document we therefore perform such a vulnerability analysis for IAM and data sharing applications. The catalogue at hand is complementary to existing generic cloud vulnerability catalogues as published, e.g., by ENISA, OWASP, or SECCRIT. Namely, it puts its main focus on identity and access management systems as well as data sharing platforms, and only contains generic vulnerabilities if they are of outstanding relevance for the considered applications. Special attention will be paid to all aspects concerning the users' privacy.

Following CREDENTIAL's methodology, all identified vulnerabilities are mapped to the threat categories of STRIDE and LINDDUN, which will then be evaluated using STRIDE in subsequent deliverables.

Document information

Contributors

Name	Partner
Andreas Abraham	TUG
Andreas Happe	AIT
Felix Hörandner	TUG
Aleksandar Hudic	AIT
Agi Karyda	AIT
Stephan Krenn	AIT
Thomas Lorünser	AIT
Nicolás Notario McDonnell	ATOS
Stefan Schiebeck	AIT
Evangelos Sfakianakis	OTE
Florian Thiemer	FOKUS

Reviewers

Name	Partner
Welderufael Tesfay	GUF
Juan Carlos Perez Baun	ATOS
Andrea Migliavacca	LISPA
Florian Wohner	AIT

History

0.1	20.07.2016	Aleksandar Hudic	Initial commit and first toc draft
0.2	02.08.2016	Aleksandar Hudic	Drafting the vulnerability catalog
0.3	08.08.2016	Andreas Abraham	Introduction, vulnerability elicitation methodology
0.4	17.08.2016	Nicolás Notario McDonnell	Risk assessment
0.5	23.08.2016	Aleksandar Hudic	Related work
0.6	24.08.2016	Aleksandar Hudic Florian Thiemer	Vulnerability classes
0.7	29.08.2016	Stephan Krenn	Added auto-compiled tables
0.7	31.08.2016	Aleksandar Hudic	Vulnerability mapping with corresponding threats
0.8	02.09.2016	Stephan Krenn	Added overview table in appendix
0.9	17.09.2016	Aleksandar Hudic	Addressing reviewers comments
0.10	19.09.2016	Stephan Krenn	Added abstract and executive summary

0.11	22.09.2016	Andreas Happe Stephan Krenn Stefan Schiebeck	Reformatted document to focus more on IAM and data sharing
0.12	23.09.2016	Andreas Happe Stephan Krenn	Added new requirements to several categories
0.13	28.09.2016	Stephan Krenn Thomas Lorünser	Introductions to classes, added vulnerabilities
0.14	29.09.2016	Stephan Krenn Thomas Lorünser Stefan Schiebeck	Proof-reading, final threat mapping
1.0	30.09.2016	Agi Karyda Stephan Krenn	Final copy-editing before submission

Table of Contents

List of Figures

List of Tables

List of Acronyms

ACL	Access Control List
BFT	Byzantine Fault Tolerance
CP	Cloud Provider
CPU	Central Processing Unit
CSRF	Cross-Site Request Forgery
DFD	Data Flow Diagram
DoS	Denial of Service
GPG	GNU Privacy Guard
HSM	Hardware Security Module
IAM	Identity and Access Management
ICT	Information and Communication Technologies
IOI	Items of Interest
IdP	Identity Provider
IP	Internet Protocol (address)
HTTP	Hypertext Transfer Protocol
OWASP	Open Web Application Security Project
SECCRIT	SEcure Cloud computing for CRitical infrastructure IT
SIEM	Security Information and Event Management
SSL	Secure Sockets Layer
TLS	Transport Layer Security
TOTP	Time-based One-Time Password

XSS	Cross-Site Scripting

Glossary of Terms

DREAD
: Approach developed by Microsoft to prioritize risks, depending on **D**amage potential, **R**eliability, **E**xploitability, **A**ffected users, and **D**iscoverability.

LINDDUN
: Approach to prioritize privacy risks, depending on **L**inkability, **I**dentifiability, **N**on-repudiation, **D**etectability, Information **D**isclosure, Content **U**nawareness, and **N**on-compliance.

STRIDE
: Threat modeling technique developed by Microsoft; considers the categories **S**poofing, **T**ampering, **R**epudiation, **I**nformation disclosure, **D**enial of service, and **E**levation of privilege.

1 Introduction

A vulnerability is a flaw or weakness in a (computer) system which can be exploited to harm or violate the system's security, whereas a threat describes a potential for violation of security that might exploit a vulnerability by an action or event [Shi13].

The secure cloud identity wallet developed within the CREDENTIAL project is designed to deal with sensitive data, especially related to eGovernment, eHealth, and eBusiness scenarios (for a description of the considered scenarios and use-cases, cf. [ACF+16]). Therefore, CREDENTIAL needs to aim at high security standards, and needs to take into account various privacy preserving aspects. To reach these goals, this vulnerability catalogue will be utilized in different project phases and development stages. This is because knowing and understanding potential weaknesses of a system is of upmost importance in order to be able to develop mitigation strageties and to correct them.

The main objective of this deliverable is to identify potential vulnerabilities related to certain threat types, in particular related to identity and access management, identity provisioning, and data sharing applications.

1.1 CREDENTIAL

The CREDENTIAL wallet is a data-sharing and identity management platform in the cloud. Moreover, CREDENTIAL aims at highest security and privacy standards compared to already existing solutions. This will be achieved by deploying cryptographic primitives such as proxy-re-encryption [BBS98] and redactable signatures [JMSW02, SBZ01] as core elements of the system.

Figure 1 depicts the generic idea of the CREDENTIAL wallet, in particular being an IAM system used to access several services and offering a data sharing platform. The main features offered by CREDENTIAL wallet are as follows:

- Securely store personal data in the wallet (cloud).
- Share personal data with other participants in a secure way (using proxy re-encryption mechanisms).
- Use as an IAM system to access various online services in a privacy-preserving way.
- Malleable signatures allow to hide information in documents when sharing it with other participants while still maintaining the authenticity of the revealed data.

A benefit using the CREDENTIAL system is that the trust assumptions towards the cloud provider can be reduced by employing modern cryptographic methods and techniques. Additionally, increasing flexibility together with high security and strong authentication guarantees are further advantages.

The design of such a system requires several methodologies and approaches in various tasks to fulfill CREDENTIAL's challenges. This also results in an overall CREDENTIAL methodology

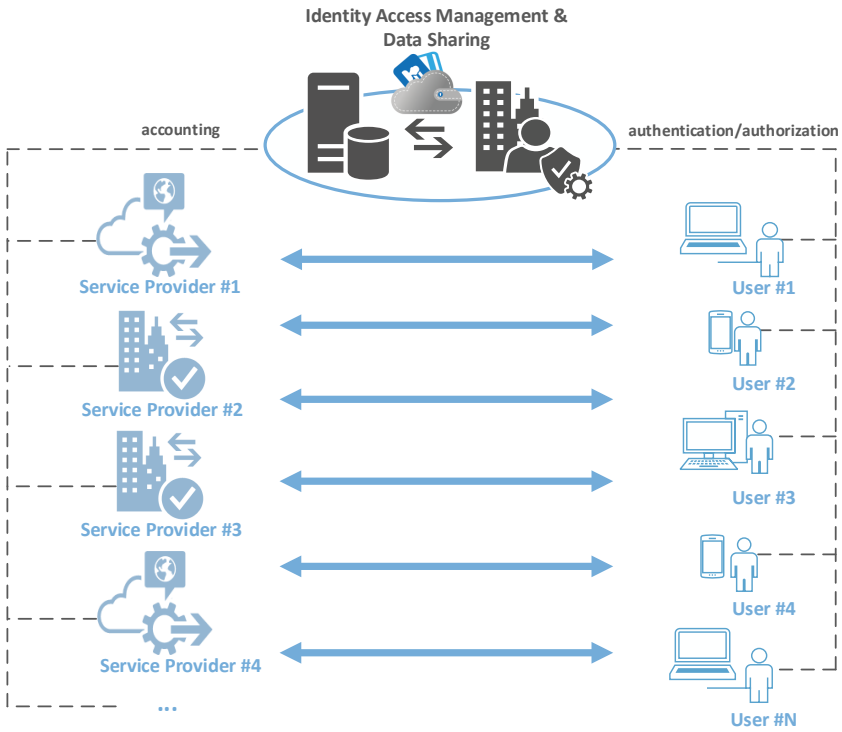

Figure 1: Generic IAM and data sharing setup within CREDENTIAL

regarding the identification of threats and process the risk assessment. This overall methodology makes use of STRIDE, DREAD and LINDDUN. STRIDE and LINDDUN are being used to identify threats whereas DREAD is being utilized for the risk assessment.

1.2 Motivation

Identification of vulnerabilities in an early stage is important and necessary to be able to take the identified weaknesses into consideration already in the design phase of a system. Moreover, the knowledge of a system's vulnerabilities allow one mitigate or correct these weaknesses, resulting a more secure system.

The main goal of the vulnerability catalogue is to have a central collection of identified vulnerabilities. This list of vulnerabilities will have influence in different development stages of the system. In an earlier stage, the design phase will be influenced by utilizing this vulnerability catalogue for example when choosing the identity protocol being used. The vulnerability catalogue is also needed during the implementation phase in order to follow security implementation guidelines with the aim of avoid code vulnerabilities. Finally, this catalogue will have influence in various further tasks and decisions in the CREDENTIAL project, e.g., risk assessment, secure design, and development of CREDENTIAL use cases.

Besides a list of vulnerabilities, this document provides further information such as the descrip-

tion of the used methodology to identify vulnerabilities as well as a description of the vulnerability categories. Moreover, the vulnerabilities are mapped to the related threats, potential countermeasures are suggested, and the targets of the vulnerabilities are identified.

1.3 Definitions

This subsection introduces the definition of the terms threat and vulnerability. This should help the reader to better understand the meaning and how they differ from each other.

Threat: The Internet security glossary[1] describes a threat in computer systems as *"Any circumstance or event with the potential to adversely affect a system through unauthorized access, destruction, disclosure, or modification of data, or denial of service." [Shi13]* A threat might exploit a vulnerability to harm or violate the security of a system.

Vulnerability: In computer security, a vulnerability describes any weakness or flaw in a system, which can be exploited by a threat to attack or for a harmful event. Flaws and weaknesses are possibly in the system's design, implementation, or operation and management. This event or attack will violate the system's security [Shi13].

1.4 Scope and Relation to Other CREDENTIAL Deliverables

This section identifies both, the scope of this deliverable and also the relation to other CREDENTIAL deliverables. Multiple deliverables are concerned with threats and vulnerabilities. Therefore, this section should help the reader to understand the differences between deliverables and to find the appropriate document.

Deliverable **"D2.4 Vulnerability Catalogue"** provides a central repository where all identified vulnerabilties related to CREDENTIAL are listed. Vulnerabilities in this list contain additional information namely such as a unique ID, a mapping to the related threats, or the affected stakeholders. Besides this list, the document provides a description of the used methodology to identify vulnerabilities, the vulnerability management and a description of the vulnerability classes. This deliverable is being used as input for other deliverables, especially in "D2.5 System Security Requirements, Risk and Threat Analysis – 2nd Iteration". Moreover, this deliverable is consulted in the decision making process for instance when making system design decisions or technology selection decisions.

The following list presents a brief description of each deliverable and shows which relation each specific deliverable has compare to this one.

D2.2 System Security Requirements, Risk and Threat Analysis – 1st Iteration: This deliverable focuses on defining security requirements and analyzing security threats. The

[1] https://tools.ietf.org/html/rfc4949

identified threats as well as the process of identifying and analyzing dangers were used as input for the document at hand, in particular in order to find the appropriate vulnerability categories.

D2.3 Cloud Identity Wallet Requirements: D2.3 focuses on CREDENTIAL related requirements. Moreover, it provides a central collection point for all requirements. Additional requirements will be infered from the vulnerability catalogue and will be integrated into D2.3.

D2.5 System Security Requirements, Risk and Threat Analysis – 2nd Iteration: Deliverable D2.5 is the second iteration of D2.2. While D2.2 was very generic and mostly independent of the concrete use cases, D2.5 will take into accout the concrete pilot specifications (from D2.1 and D6.1, repectively). The security requirements will be based on the performed risk and threat analysis, which in turn will be based on the vulnerabilities identified in the document at hand.

D2.6 User Centric and Usability Requirements: D2.6 complements the requirements collection with user centric and usability requirements. Similar to D2.3, the vulnerabilities catalogue (and the threats identification from D2.5) will be used in the requirements elicitation process.

D4.1 Assessment Report on Cryptographic Technologies, Protocols and Mechanisms: The purpose of this document is twofold. First, technologies are assessed according to CREDENTIAL's needs. Second, a technology selection process is being performed based on CREDENTIAL-related evaluation criteria. D2.4 is used as one input to define the best-possible evaluation criteria.

D5.1 Functional Design: In D5.1, the functional design of the architecture is described in detail and later used as basis for the technical design and implementation. Some functional design decisions might be influenced by D2.4 when taking vulnerabilities into consideration.

1.5 Outline

This document is structured as follows: Section 2 highlights the most relevant state of the art related work with regards to risk assessment and vulnerability management processes. Next, in Section 3, we outline the methodology for vulnerability and risk assessment that identifies CREDENTIAL-relevant vulnerability and threat classes. Afterwards, in Section 4, we present the enumerated vulnerabilities with regards to our generic CREDENTIAL use case model. Finally, we conclude the deliverable in Section 5. Additionally, in Appendix A, gives an overview over all vulnerabilities.

2 Related Work

Within this section we outline the current state of the art in vulnerability elicitation, vulnerability management, and risk assessment by addressing the context of cloud environments. We provide an overview of the current approaches that address vulnerability catalogs and identify their shortcomings with regards to addressing cloud computing principles, aspects and deployment models. The risk assessment subsection will describe the link with the risk assessment & management framework described in [AHH+16] and the vulnerability catalog compiled within CREDENTIAL and presented in Section 4.

2.1 Risk Assessment

As presented in D2.2 [AHH+16], the main steps of any risk management process are:

1. **Identify what is to be protected:** which is implemented within CREDENTIAL by developing a Data Flow Diagram of the system, depicting its entities, processes, data stores and data flows (that are the assets to be protected)
2. **Identify threats:** systematically iterate through the identified assets and STRIDE threat categories and identify relevant threats
3. **Prioritize threats:** assign numerical values or categories to the threats according to their potential impact and likelihood
4. **Address threats:** by establishing measures or including security controls. The threats should be addressed according to the established priority scale
5. **Monitor assets and the efficiency of the measures**

While CREDENTIAL has decided to fulfill the prescribed risk management process by using a combination of the STRIDE and DREAD [MMD+03] approaches (see [AHH+16]), STRIDE and other risk management methodologies do not always prescribe in detail how this steps can or must be implemented. One of the critical steps is to identify the threats. While Microsoft's Threat Modeling Tool ([Mic16]) already provides great support for semi-automatic identification of most common threats, a vulnerability catalogue will complement it by aiding in the identification of non-covered threats. This catalogue then also supports the threat prioritization-, threat addressing-, and monitoring steps detailed below.

Identifying threats through vulnerabilities: As already mentioned, Microsoft's Threat Modeling Tool [Mic16] includes a list of 47 threats classified using the STRIDE threat categorization taxonomy (Spoofing, Tampering, Information Disclosure, Denial of Service, Elevation of Privilege). While this is a quite large list, it is not a comprehensive list and there are more threats than the ones present in the list. By having a vulnerability catalog that can be jointly assessed with the to-be-protected assets it is expected to identify many threats not actually

present in Microsoft's tool and or to have more concrete threats that can be addressed in a more specific way.

A "Lack of fail-over strategy" vulnerability when contextualized in CREDENTIAL's DFD, may lead to the identification of a new threat "Third parties unable to grant access to users" that would have a deep impact in CREDENTIAL's reputation or even have economical consequences.

Prioritizing threats: In CREDENTIAL, DREAD has been chosen as the preferred approach to evaluate risk priorities. This method is based on the idea that risk is a combination of the potential damage, how easy is to exercises the threat, how many people will be impacted, and how easy it is to discover the threat. These measures are not easy to provide if the analyst is not aware of the different paths (vulnerabilities) that may lead to the threats realization. However, a list of vulnerabilities linked to threats is a mean that can be used to ease this exercise, as it is easier to determine how exploitable a vulnerability is than to determine the same for a threat.

Address threats: Neither STRIDE nor DREAD help to directly address the identified threats. These threats have to be addressed by redesigning the system, by applying or inventing technical or organizational mitigation measures, or, in some cases, risks simply have to be accepted. CREDENTIAL's vulnerability catalog may serve as a guide to address the identified threats. Instead of trying to tackle a threat at a general level, the mitigation measures may be applied to the related vulnerabilities, which in the end will result in the minimization of the threat's risk, that is the main objective.

Monitoring assets: Finally, the developed vulnerability catalogue may play an important role in the test and operational phases. The list of vulnerabilities may serve as a kind of test suite where threats may be tried to be exercised before deployment or once the system is operational to ensure that the mitigation measures are operative and provide the desired results.

2.2 Vulnerability Management and Elicitation in Cloud-Based Environments

In the following we detail on some of the most prominent existing vulnerabilities catalogues for cloud applications.

2.2.1 ENISA Vulnerability List

The European Network and Information Security Agency (ENISA) in its technical report [Dan09] outlines the assessment of the security risks and benefits for using cloud computing, and provides security and legal guidance for potential and existing cloud users. The security assessment is based on three use-case scenarios: Small-medium enterprise migration to cloud computing services, the impact of cloud computing on service resilience, cloud computing in eGovernment

(e.g., eHealth). The report from ENISA puts the emphasize on the most relevant technical, legal, policy and organizational risks within the context of cloud computing. Furthermore, the core focus of the technical report of ENISA is on the opposed risks rather then vulnerabilities. Therefore, each individual risk to determine appropriate risk level is associated with corresponding set of vulnerabilities and affected assets first.

The listed vulnerabilities from ENISA refer to generic vulnerability issues. However, CREDENTIAL focuses on identity provisioning and data sharing, which are very special applications for which additional specific vulnerabilities need to be taken into account.

2.2.2 OWASP Vulnerability List

The Open Web Application Security Project (OWASP)[2] provides a comprehensive and technical list of vulnerabilities demonstrated through various technologies and corresponding attacks. The comprehensive vulnerability list is the converged output of several initiatives from OWASP that implement best practices in development, testing, code review and application security verification as guidelines. The enumerated vulnerabilities that are very technical and detailed are also used as input for threat modeling and risk assessment processes, i.e., STRIDE and DREAD.

The vulnerability catalogue of OWASP is specifically targeting web technologies. However, when enumerating our CREDENTIAL related vulnerabilities we tried to maintain a higher level of abstraction that is focused more on application specific problems rather than actual implementation flaws. Yet, because of its practical importance, Figure 2 contains a summary of the Top-10 vulnerabilities according to OWASP.

2.2.3 SECCRIT Vulnerability Catalogue

The European project *SEcure Cloud computing for CRitical infrastructure IT (SECCRIT)*[3] conducted a thorough analysis and evaluation of cloud computing technologies with regards to security risks in highly sensitive environments. Part of the effort was devoted to analyze risks by identifying and enumerating cloud specific vulnerabilities. SECCRIT provides an understanding of vulnerabilities and associated risks, by outlining a cloud-specific threat and vulnerability catalogue [BLSS11] which can be used to support the implementation of a risk assessment. The vulnerabilities were primarily arranged in line with the five NIST essential cloud characteristics: on-demand self-service, broad network access, resource pooling, rapid elasticity and measured service. Additionally, they were extended by aspects regarding, e.g., virtualization as a key enabling technology, important organizational issues, the underlying physical cloud infrastructure, or issues associated with contemporary cloud offerings. The SECCRIT threat and vulnerability

[2] Open Web Application Security Project - https://www.owasp.org/index.php/Category:Vulnerability

[3] SECCRIT was a multidisciplinary research project with the mission to analyse and evaluate cloud computing technologies with respect to security risks in sensitive environments, and to develop methodologies, technologies, and best practices for creating a secure, trustworthy, and high assurance cloud computing environment for critical infrastructure IT, cf. https://www.seccrit.eu/

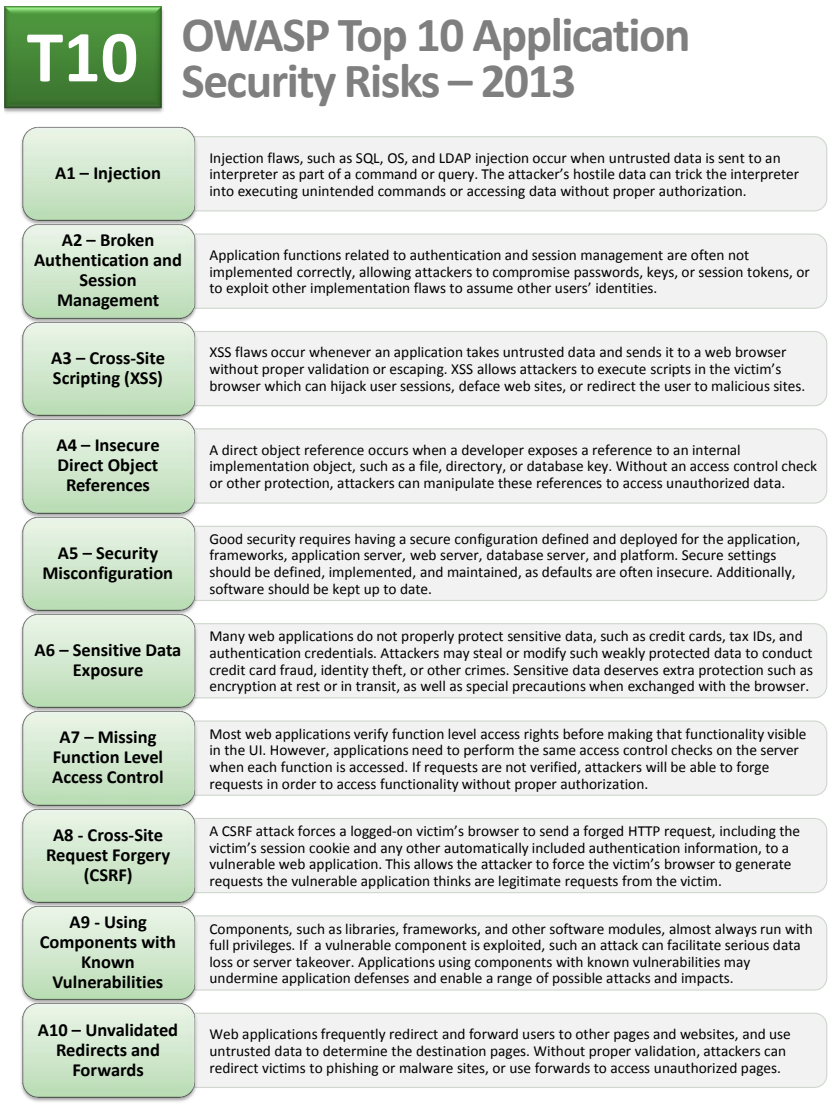

Figure 2: OWASP Top 10 Web Application Security Risks

catalogue [BLSS11] outlines generic threats and vulnerabilities on a very high level to guide detailed vulnerability assessments for specific applications.

Unfortunately the threat and vulnerability catalogue from SECCRIT does not specifically take into accout privacy, data sharing, or identity management or IAM aspects. Furthermore, it does not address currently prominent hybrid and federated cloud models. This is especially interesting when considering the challenges within the context of various cloud principles (e.g., interoperability, portability, transparency, scalability, auditability, availability, concurrency, data governance, and resilience) that are additionally confronted with cross administrative and geographical challenges.

2.2.4 Further Catalogues

Besides the above vulnerabilities, there exists a broad range of other cloud-specific catalogues. Specifically, the privacy-related vulnerabilities listed in this document were partially inspired by the challenges identified by Ziegeldorf et al. [ZMW14]. Furthermore, the German Federal Office for Information Security maintains an abstract and high-level list of vulnerabilities and attack vectors for (non privacy-preserving) identity and access management systems [BSI16].

3 Vulnerability Management

In the following we first describe CREDENTIAL's general approach to assess risks and threats. In Section 3.2 we then explain the vulnerability elicitation methodology and define the vulnerability categories that are later used to cluster weaknesses.

3.1 Overall Risk Assessment Methodology

This section describes the overall methodology to identify threats and perform a risk assessment. In CREDENTIAL, different methodologies and approaches are combined together to an overall CREDENTIAL methodology. This overall methodology is used to identify threats, vulnerabilities and later create the risk assessment. STRIDE and LINDDUN are being used to identify security threats whereas DREAD is being used for the risk assessment.

3.1.1 STRIDE

CREDENTIAL uses STRIDE, originally developed by Microsoft, as a threat modeling technique. STRIDE classifies threats according exploitation classes and each threat class of STRIDE can be matched with a corresponding security property, cf. Table 1.

Threat Class	*Security Property*
Spoofing	Authentication
Tampering	Integrity
Repudiation	Confirmation
Information Disclosure	Confidentiality
Denial of Service	Availability
Elevation of Privilege	Authorization

Table 1: STRIDE Threat Classes with corresponding Security Properties

STRIDE is an iterative process consisting of four steps, namely diagramming, identifying threats, address threats, and validation, which are described as follows:

1. **Diagramming:** First, STRIDE starts with the diagramming step. This step is a visual description of the system containing its entities, processes, data stores, data flows and trust boundaries. Data Flow Diagram (DFD) is the diagram type recommended by STRIDE.
2. **Identifying Threats:** After finishing the system description, all elements in the diagram are used to discover relevant threats. This task can be supported by Microsoft's Threat Modeling tool, which already contains a large repository of threats associated with each system element.

3. **Address Threats:** Next, the identified threats have to be addressed by performing various actions such as redesigning the system, applying additional controls for mitigation or inventing mitigations. The goal of this step is to reduce the residual risk to an acceptable level.

4. **Validation:** Finally, the threat model which has been developed has to be validated to ensure that the DFD matches its implementation, all threats are clearly identified to specific actions and all threats including mitigations have to be well described.

3.1.2 LINDDUN

LINDDUN is the privacy counterpart of the STRIDE methodology and follows a similar approach. It first defines a list of threat categories that match desirable privacy properties (see Table 2) and then continues by describing a simple iterative process supported by a series of tools.

Threat Class	*Privacy Property*
Linkability	Unlinkability
Identifiability	Anonymity & Pseudonymity
Non-Repudiation	Plausible deniability
Detectability	Undetectability & unobservability
Information **D**isclosure	Confidentiality
Content **U**nawareness	Content awareness
Non compliance	Policy and consent compliance

Table 2: LINDDUN Threat Classes with corresponding privacy Properties

As mentioned, it also follows a very similar process as the one described for STRIDE:

1. **Diagramming:** LINDDUN threat modeling methodology is tightly bound to Data Flow Diagrams. One of the main benefits of mixing LINDDUN and STRIDE is that this first step is common and the same DFD may be used for both analysis.

2. **Identifying Threats:** Mimicking STRIDE, LINDDUN methodology already identifies what are the potential threat categories that are applicable to each of the DFD elements (e.g. Entities, in a DFD context, are only vulnerable to threats related to Linkability, Identifiability and Unawareness). LINDDUN (as STRIDE) does not prescribe a unique methodology to identify threats but the methodology is complemented with useful privacy threat tree patterns that can be used to identify relevant threats. Vulnerability catalogues such as the one presented in this deliverable may also support this task.

3. **Address Threats:** While the prioritization and categorization of threats are out of LIND-

DUN's scope, it supports[4] the selection of the most adequate mitigation strategy which in the end eases the establishment or privacy requirements and or the selection of the most adequate Privacy Enhancing Technologies.

Neither STRIDE nor LINDDUN provide explicit risk analysis support. Hence, the prioritization of threats is out of scope and has to be addressed with an additional tool/methodology. As presented in D2.2 [AHH+16], DREAD, which is described in the following subsection 3.1.3, will provide support in this particular task.

3.1.3 DREAD

In addition to STRIDE and LINDDUN, DREAD is utilized in CREDENTIAL to identify and manage risks, which were discovered by STRIDE and LINDDUN. DREAD was also developed by Microsoft in order to assess risks and prioritize them. In DREAD, five different threat aspects are used to compute a risk value. Each threat is evaluated against these five categories with a number between 0 and 10 depending on the level. Equation 3.1.3 displays the DREAD algorithm consisting the five categories of DREAD used to compute risks.

$$\text{Risk}_{\text{DREAD}} = \frac{\textbf{D}\text{amage} + \textbf{R}\text{eproducibility} + \textbf{E}\text{xploitability} + \textbf{A}\text{ffected Users} + \textbf{D}\text{iscoverability}}{5}$$

3.1.4 Threat Classes

With regards to the above mentioned methodologies, STRIDE and LINDDUN, we propose an initial set of threat classes, i.e., these classes are cross mapped with the identified vulnerabilities outlined in Section 4.

Method	Threat Class	Description
STRIDE	**Spoofing Identity**	The concept of identity spoofing enables unprivileged code to use someone else's identity, and hence, their security credentials. For example, a driver that uses some form of a password mechanism is subject to this type of attack. Not all such drivers have security flaws, although, they are vulnerable to security flaws based on spoofing identity. The designers and implementers of the driver need to evaluate the level of vulnerability.
	Tampering with Data	Data tampering involves the malicious modification of data. Examples include unauthorized changes made to persistent data, such as that held in a database, and the alteration of data as it flows between two computers over an open network, such as the Internet.

[4] https://distrinet.cs.kuleuven.be/software/linddun/solutions.php

Method	Threat Class	Description
	Repudiation	Repudiation threats are associated with users who deny performing an action without other parties having any way to prove otherwise. For example, a user performs an illegal operation in a system that lacks the ability to trace the prohibited operations
	Information Disclosure	Information disclosure threats involve the exposure of information to individuals who are not supposed to have access to them. 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.
	Denial of Service	Denial of service (DoS) attacks deny service to valid users e.g., by making a Web server temporarily unavailable or unusable. We must protect against certain types of DoS threats simply to improve system availability and reliability.
	Elevation of Privileges	In this type of threat, an unprivileged user gains privileged access and thereby has sufficient access 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 become part of the trusted system itself, which results in a dangerous situation indeed.
LINDDUN	**Linkability**	Threats related to linkability are those that can allow attackers to be able to sufficiently distinguish whether two IOI (items of interest) are linked or not, even *without* knowing the actual identity of the subject of the linkable IOI.
	Identifiability	In this type of threat, an attacker can sufficiently identify the subject within a set of subjects (i.e. the anonymity set). The attacker is able to link one data subject with a piece of data.
	Non-Repudiation	Non-repudiation threats are those that enable an attacker to link (in a undeniable way) an action with a data subject. This kind of threats are key in systems where anonymity is expected (e.g. eVoting system). For further information, also see Repudiation.
	Detectability	Detectability attacks result in the ability of an attacker to distinguish if a particular IOI exists or not, even if it is not able to determine the content of such IOI.
	Unawareness	Unawareness threats are those that stem from the lack of information regarding the consequences of sharing information.

Method	Threat Class	Description
	Non-Compliance	All those threats that are related to the non-complaint with legislation, regulation or corporate policies.

Table 3: List of identified threat classes with regards to STRIDE and LINDDUN.

3.2 Vulnerability Identification in CREDENTIAL

We next describe our vulnerability identification method and introduce the vulnerability categories used in the remainder of this document.

3.2.1 Vulnerability Elicitation Methodology

CREDENTIAL's vulnerability elicitation methodology is based on Microsoft's threat modeling process [Mic05] shown in Figure 3. This process has been adopted in order to fulfill CREDENTIAL's needs.

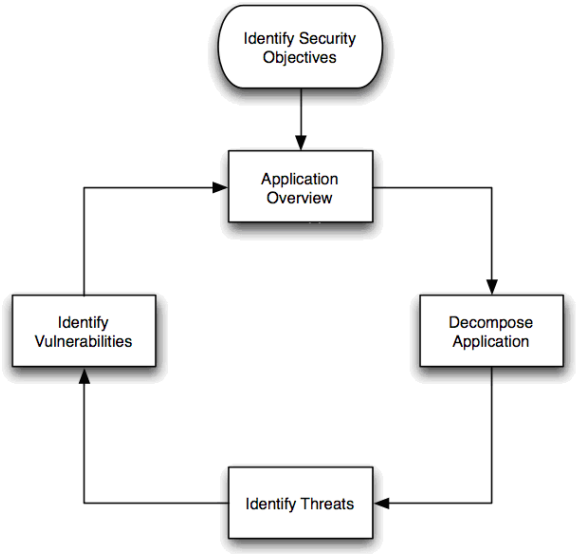

Figure 3: Microsoft's Threat Modeling Process [Mic05]

The approach defined by Microsoft [Mic05] has five major steps to identify vulnerabilities, which are described as follows:

1. **Identify Security Objectives.** First, the security objectives have to be identified in order to determine the following steps.

2. **Survey the Application.** Next, the important characteristics and actors needs to be detected.
3. **Decompose the Application.** Decomposing the application will support the detailed understanding of the mechanisms, which will help to uncover relevant threats.
4. **Identify Threats.** The results from steps 2 and 3 are used to identify threats.
5. **Identify Vulnerabilities.** Finally, reviewing the threats and the different layers of the application to identify vulnerabilities in the application. Vulnerability classes are used to categorize them into different areas.

The CREDENTIAL approach uses requirements and actors to identify threats and vulnerabilities. Identified threats collected in various deliverables are being used to detect weaknesses of the system.

In addition to Microsoft's approach, existing vulnerability repositories are used to identify weaknesses in CREDENTIAL. The utilized repositories and catalogs are the SECCRIT Vulnerability Catalogue [BLSS11], the ENISA Vulnerability List [Dan09], and the OWASP Vulnerability List [OWA12]. A description of these vulnerability collection can be found in Section 2.2.

3.2.2 Vulnerability Elicitation Outcomes

Before presenting the identified vulnerability classes, we give a brief overview of the outcome of the presented methodology.

The security objectives and threats coincide with those identified in Tables 1 and 2 and will not be discussed here any further.

The main actors of the generic IAM and data sharing applications are:

- Issuers
- Identity (management) providers
- Storage providers
- Service providers
- Users

Finally, the decomposition of the application lead to the data flow diagram presented in Figure 4 which was already used in D2.1 [ACF+16] to perform the initial security requirements elicitation.

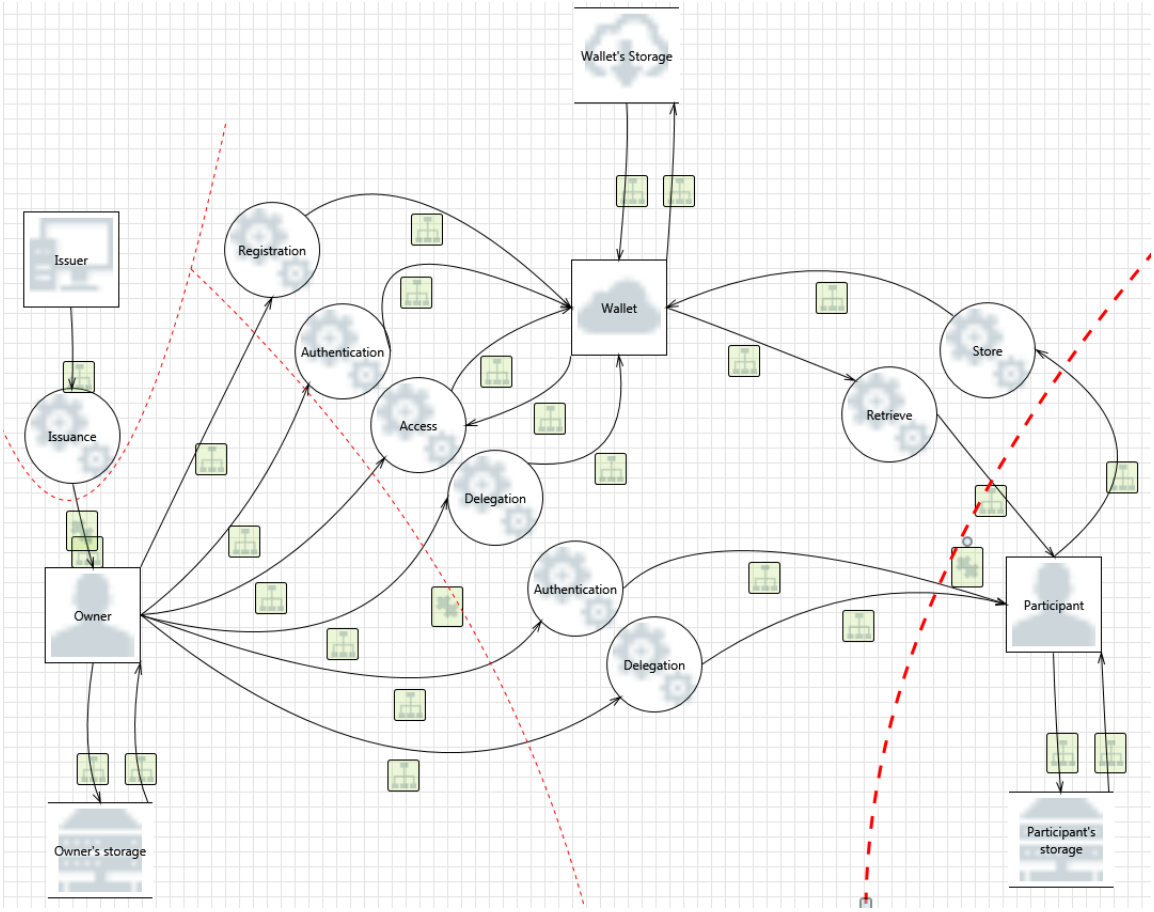

Figure 4: CREDENTIAL's Wallet DFD - Level 1

3.2.3 Vulnerability Classes

In Table 4 we enumerate the core set of the vulnerability classes that we are using to cluster vulnerabilities.

It should again be mentioned that this catalogue—and thus also the considered categories—should be seen as extension of generic catalogues and aims at minimizing the redundancy by mainly discussing complementary issues for the specific scope of identity provisioning and data sharing. In particular, generic vulnerability classes related to scalability (e.g., insufficient resources), interoperability (e.g., loss of connectivity between services due to misconfiguration or hardware malfunction), portability (e.g., vendor lock-in or insecure code distribution), or availability (e.g., lack of communication link redundancy) are therefore out of scope of this document, but can be found in the related (generic) literature. Furthermore, technology- and implementation-specific weaknesses (see, e.g., Figure 2) are not covered in this document, as those are typically also not specifically related to certain applications. However, if a vulnerability has a particularly high potential impact on IAM or data sharing applications, we might still mention it in our catalogue.

ID	Vulnerability Class	Description
IMP	**Impersonation**	This class contains vulnerabilities that might enable an attacker to take over a user's identity.
BEH	**Behavior Analysis**	This class contains vulnerabilities that would allow an adversary to break the user's privacy by analyzing, e.g., access patterns.
ISO	**Service Isolation**	This class contains vulnerabilities that might leak sensitive information to an adversarial program running, e.g., on the same physical machine.
AUD	**Audit Trails**	This class contains vulnerabilities related to integrity and privacy that would arise from insecure logging and audit trails.
AUT	**Authentication**	This class contains vulnerabilities that might arise from weak mutual certification of users and servers.
ACL	**Access Control**	This class contains vulnerabilities in particular related to user privacy that could arise, e.g., from improper usage of access control lists.
INT	**Integrity and Consistency**	This class contains vulnerabilities related to data correctness, in particular considering parallel accesses by different users.
CON	**Confidentiality**	This class contains vulnerabilities that might undermine the secrecy of stored data.

Table 4: Vulnerability Classes

4 Vulnerability Catalogue

This section addresses the most relevant vulnerabilities with respect to identity management and data sharing services. Because user privacy is a major goal of the CREDENTIAL system, specific categories of privacy related vulnerabilities have also been introduced. In summary, the vulnerabilities are clustered in eight vulnerability classes and presented in the following format:

Vulnerability name	
Reference ID	A unique identifier for this vulnerability, where the first three letters indicate the category, e.g., AUD01
Description	A detailed description of the vulnerability
Countermeasures	Potential mitigation strategies for the vulnerability are described here
Related categories	Certain vulnerabilities might affect different categories. For instance, audit-trail-related vulnerabilities might be related to (and potentially contradict) vulnerabilities centered around user privacy and behavior analysis
Responsible	Here, the stakeholders responsible for taking countermeasures are identified.
Threats	We here map the vulnerabilities to the threat categories from STRIDE and LINDDUN, cf. Sections 3.1.1 and 3.1.2
Specific to	We here indicate whether the given vulnerability is specific to IAM or data sharing, and whether it has a direct relation to the user's privacy

Appendix A contains an overview table over all vulnerabilities identified in this section.

We emphasize once more that this catalogue should be seen as an extension to existing catalogues such as OWASP (cf. Figure 2). Namely, we do not cover generic vulnerabilities that also affect many other cloud applications, but we only list vulnerabilities that are specific to IAM and data sharing. However, certain generic vulnerabilities might still be mentioned because of their high importance and influence to such systems.

4.1 Impersonation

Impersonation attacks are central vulnerabilities in any IAM system. In such attacks, an adversary tries to fake a user's identity in order to get access to non-public resources. Such attacks can be performed on different layers, using different vulnerabilities that are discussed in the following. Impersonation attacks are particulary relevant in identity management settings, as successfully mimicing the user's identity towards the identity provider allows the adversary to impersonate the user towards all linked service providers.

Man-in-the-middle attack on network level	
REFERENCE ID	IMP01
DESCRIPTION	A common attack vector is identity theft, esp. when it can be performed through attacks on the transport level, i.e., the adversary does not have to attack the IdP or services themselves but can attack the (commonly less monitored) communication between components. This allows the attacker to assume the identity of another user, potentially gaining access the other other user's capabilities. Common enablers for this kind of attack are protocol vulnerabilities or lack of integrity protection and confidentiality of transported messages.
COUNTERMEASURES	Enhancing the used protocol's security is the commonly used countermeasure. In case of HTTP, enforcement of TLS with confidenciality and integrity guarantees in combination with verification of the peer's TLS certificate (Certificate Pinning, Revocation Lists) is commonly given as solution. As the attacker needs access to the transported network communication, a clean service isolation on the network level can improve security.
RELATED CATEGORIES	Service Isolation
RESPONSIBLE	User, Issuer, Service Provider, Identity Provider, Storage Provider
THREATS	Spoofing identity, tampering with data, repudiation
SPECIFIC TO	☑ IAM ☑ Data Sharing ☑ Privacy

Unintended session re-use	
REFERENCE ID	IMP02
DESCRIPTION	The user identity is mapped to an unique session identifier; if the attacker gains direct or indirect access to this session identifier he can perform operaitons as that given user. In contrast to IMP01, only a subset of operations is accessible for the attacker, the whole identity is not compromised. On a network/transport layer protocol replay attacks can be performed. Here a simple operation is captured by the attacker and can be replayed against the target. On a higher level, i.e. HTTP, CSRF-Attacks are common. In this case the attacker can utilize additional flaws within an HTML interface (e.g., XSS attacks) to execute serverside operations. As the identity is automatically added by the target's browser, the user's current server-side identity is used as identity for those operations.
COUNTERMEASURES	Issued operations must be made unique. If the server can verify that an operation has already be performed or an incoming operation is out-of-sequence, the operation can be discarded. On a network level, usage of timestamps or nonces is often employed to solve this problem. A similar approach, called Synchronizer-Pattern is employed to prevent CSRF attacks.
RESPONSIBLE	User, Issuer, Service Provider, Identity Provider, Storage Provider
THREATS	Spoofing identity, tampering with data
SPECIFIC TO	☑ IAM ☑ Data Sharing ☑ Privacy

Compromise of identity information	
REFERENCE ID	IMP03
DESCRIPTION	Similar to online impersonation attacks, attacks against stored/cached credentials or key material can be performed. An notorious example of those are so called "pass-the-hash/token" attacks within Microsoft Windows networks. During those, locally stored hashes or tokens are extracted from Windows Workstations and reused against Windows Servers to gain access with the same access level as the token's original owner.
COUNTERMEASURES	Tokens must be stored in a manner that prevents access by unauthorized persons. In addition, if tokens are generated, they should have a clear scope and expiration date.
RESPONSIBLE	User, Issuer, Service Provider, Identity Provider, Storage Provider
THREATS	Spoofing identity, tampering with data
SPECIFIC TO	☑ IAM ☑ Data Sharing ☑ Privacy

Insecure session Ids	
REFERENCE ID	IMP04
DESCRIPTION	This is a very common vulnerability, for detailed information see OWASP Testing Guide v4. User identities are represented through numeric session identifieres. If the latter can be guessed by an attacker, he can assume the target's identity. This is a common occurrence with badly written web applications.
COUNTERMEASURES	Session identifiers must be generated using cryptographically secure random number generators. In addition their length must be secure.
RESPONSIBLE	User, Issuer, Service Provider, Identity Provider, Storage Provider
THREATS	Spoofing identity
SPECIFIC TO	☑ IAM ☑ Data Sharing ☑ Privacy

Identity assurance	
REFERENCE ID	IMP05
DESCRIPTION	The user's digital identity is generated by the issuer. It is part of the issuer's task, to verify that the user's electronic and physical identity are consistent. If not, an malicious actor could create a digital identity for another person and thus access that person's digital data.
COUNTERMEASURES	The issuer must verify an user's identity during intial digital identity. If regulation is available for this process, these rules must be conformed to.
RELATED CATEGORIES	Authentication
RESPONSIBLE	Issuer
THREATS	Spoofing identity, repudiation, tampering with data, information disclosure
SPECIFIC TO	☑ IAM ☐ Data Sharing ☑ Privacy

Delayed or missing revocation	
REFERENCE ID	IMP06
DESCRIPTION	This is a refinement of IMP03. If tokens are generated, that allow for offline identification of users against services, token revocation operations must be performed in a secure distributed manner. Otherwise an attacker might be able to use a local token until its expiry date is hit, even if the token has already been revoked (e.g., after an user contract is cancelled).
COUNTERMEASURES	Usage of revocation mechanisms such as revocation lists, usage of short expiry dates.
RELATED CATEGORIES	Authentication
RESPONSIBLE	Service Provider, Issuer, User
THREATS	Spoofing identity, tampering with data, repudiation
SPECIFIC TO	☑ IAM ☐ Data Sharing ☑ Privacy

Open re-directs within protocols	
REFERENCE ID	IMP07
DESCRIPTION	A common security vulnerability with ties to user's digital identities are open redirection attack. During those, weaknesses within protocols or services are utilized to generate a new network requests, origination from the vulnerable server. For example, an attacker can issue a network request originating from a compromised server including the victim's user identity. This is problematic as external security systems might utilize this information as part of their access control system. In addition, companies prefer if their servers do not access illegal data (which can be traced back to the company).
COUNTERMEASURES	Operations providing open proxy/redirection capabilities must be removed from the system. If this is not feasible, the allowed targets of this operation must be defined within a whitelist.
RESPONSIBLE	Service Provider, Storage Provider, Identity Provider
THREATS	Spoofing identity, tampering with data, repudiation
SPECIFIC TO	☑ IAM ☑ Data Sharing ☐ Privacy

Malicious group administrator	
REFERENCE ID	IMP08
DESCRIPTION	In certain scenarios, e.g., large companies, it might be the situation that there is a privileged user (administrator) that is coordinating the subscriptions, deletions, etc. of users to the IAM system. Such a super-user might misuse his power to impersonate other users. This might also lead to privacy problems, e.g., if he accesses state-keeping services linked to a user's account.
COUNTERMEASURES	Introducing a four-eye principle for resetting the user's authentication information towards the IAM system can reduce the risk of a single corrupt administrator.
RESPONSIBLE	Identity Provider
THREATS	Spoofing identity, repudiation, tampering with data, information disclosure, denial of service, non-compliance
SPECIFIC TO	☑ IAM ☐ Data Sharing ☑ Privacy

4.2 Behavioral Analysis

This category is dedicated to vulnerabilities that could lead to de-anonymization of user's, violation of data confidentiality, and other privacy problems that can arise by analiyzing behavior patterns of users. These problems are very specific to privacy-related applications, and are usually not covered by generic cloud vulnearbility catalogues.

User behavior analysis by the Identity Provider	
REFERENCE ID	BEH01
DESCRIPTION	Having an omnipresent identity provider which is used for accessing numerous different services might leak significant information to the identity provider, e.g., when does the user connect to which service. The IdP could then analyze the user's behavior based on this time-line, thereby breaking the user's privacy.
COUNTERMEASURES	On a cryptographic level, this problem could be solved by splitting the IdP into two entities, one communicating with the users and the other one with the service providers. On a policy level, accessing the user's logs could be prohobited. Note that using different accounts with the IdP on the user's side would render the idea of a central identity provider meaningless.
RESPONSIBLE	Identity Provider
THREATS	Information disclosure, Linkability, Identifiability
SPECIFIC TO	☑ IAM ☐ Data Sharing ☑ Privacy

Service provider can link user actions	
REFERENCE ID	BEH02
DESCRIPTION	This vulnerability is similar to BEH01, but considers the linkability of a user's actions by a single or multiple colluding service providers. By being able to trace recurring users, service providers could do excessive profiling of users. This vulnerability exists on a network level as well as on an applciation level.
COUNTERMEASURES	On a network level, IP addresses can be randomized using anonymity networks such as Tor. On an application level, appropriate privacy-enhancing cryptographic technologies can be deployed. In particular, no uniquely identifying information (IDs, etc.) must be disclosed to the service provider.
RESPONSIBLE	Identity Provider
THREATS	Information disclosure, Linkability, Identifiability, detectability
SPECIFIC TO	☑ IAM ☐ Data Sharing ☑ Privacy

Network traffic can be linked by outsiders	
REFERENCE ID	BEH03
DESCRIPTION	In the case of an identity provider, an external adversary tapping the network might be able to link incoming requests (from the user to the identity provider) and the outgoing network traffic (from the identity provider to the service provider), e.g., by statistical analysis. Even though the attacker might not be able to learn the sent information, it may be able to infer which user is authenticating itself to which service provider. This attack vector also exists, e.g., in anonymous network or anonymous mailing services.
COUNTERMEASURES	A common mitigation strategy is batching, i.e., the identity provider could always accumulate a certain number of authentication requests before sending them out in a random order. Also, outbound network traffic from the IdP could be routed through an anonymization network like Tor.
RELATED CATEGORIES	Service Isolation
RESPONSIBLE	Identity Provider, Storage Provider
THREATS	Information disclosure, Linkability, Identifiability, detectability
SPECIFIC TO	☑ IAM ☐ Data Sharing ☑ Privacy

Confidentiality compromise through metadata analysis	
REFERENCE ID	BEH04
DESCRIPTION	Even though the precise content of a document might be encrypted, the revealed meta-information might often be sufficient to break confidentiality. For instance, the order and time-line of downloads (e.g., user, diabetologist, insurance) might already reveal valuable information about the content of the document. Similarly, the set of authorized users specified in access control lists (ACL) might reveal similar information. Other sensitive metadata might contain size, generation date, access date, etc.
COUNTERMEASURES	In the literature, solutions for encrypting ACLs can be found. For accessing data can be done in an anonymous way, only proving the possession of the required privileges. Access patterns can disguised by using private information retrieval, where the precise accessed data location is not revealed to the storage provider.
RELATED CATEGORIES	Access Control, Confidentiality
RESPONSIBLE	Identity Provider, Storage Provider
THREATS	Information disclosure, Linkability, Identifiability, detectability
SPECIFIC TO	☐ IAM ☑ Data Sharing ☑ Privacy

Excessive logging on different layers	
REFERENCE ID	BEH05
DESCRIPTION	The user's actions can be logged extensively on different layers (e.g., network or application). This can then be used as a basis for BEH01, BEH02, BEH03, and BEH04.
COUNTERMEASURES	Every interaction should only contain only the absolute necessary information.
RELATED CATEGORIES	Audit trails
RESPONSIBLE	Issuer, Service Provider, Identity Provider, Storage Provider
THREATS	Information disclosure, Linkability, Identifiability, detectability, non-compliance
SPECIFIC TO	☑ IAM ☑ Data Sharing ☑ Privacy

4.3 Service Isolation

In particular when processing sensitive information such as cryptographic key material, insufficient means for service isolation can pose severe vulnerabilities to a system. In the following we thus discuss some of those vulnerabilities, which partially also affect other services, but might have a significant impact specifically to IAM and data sharing applications. Furthermore, some of subtle vulnerabilities (e.g., ISO05) might not be detected and covered by standard techniques.

Measurement attacks between services	
REFERENCE ID	ISO01
DESCRIPTION	Side-Channel attacks allow reasoning about executed operations, e.g., analysis of the service's response time or CPU utilization can be used to infer information about performed operations. In critical cases this can be used to extract used key material, thus reducing privacy guarantees. Side-Channel attacks can be performed between a service and its clients or between multiple co-located services (as they share the same hardware).
COUNTERMEASURES	Algorithms can be made side-channel free—in that case special attention must be given to compiler optimizations that can reintroduce side-channels into the code (side-channel resilience is contrary to performance).
RESPONSIBLE	Service Provider, Identity Provider, Storage Provider, Issuer
THREATS	Information disclosure

Data extraction between services	
REFERENCE ID	ISO02
DESCRIPTION	When services are co-located on the same computational host, unintended interactions between those services can occur. A simple example of this, would be a service that utilized large quantities of CPU time to DoS another co-located service. Recent attacks utilized hardware issues within the memory subsystem to allow an attacker within a virtual machine to access the memory of another colocated virtual machine (this attack was named Rowhammer). This attacks can typically extract small amounts of data and thus identity information (SSL keys, GPG keys) are prime targets.
COUNTERMEASURES	Service Isolation must be performed, hardware selection must prevent attacks between virtual machines (e.g., rowhammer-attacks can be thawed by careful memory selection)
RESPONSIBLE	Service Provider, Identity Provider, Storage Provider, Issuer
THREATS	Information disclosure

Shared data between services	
REFERENCE ID	ISO03
DESCRIPTION	If two virtual machines have access to the same (potential read-only) data, an attacker can pivot through attacking one of the two machines. In addition,j concurrent write-access to the same data introduces the opportunity of race-conditions and thus potential attacks.
COUNTERMEASURES	No data must be shared between virtual machines.
RESPONSIBLE	Service Provider, Identity Provider, Storage Provider, Issuer
THREATS	Information disclosure

Lack of network traffic isolation	
REFERENCE ID	ISO04
DESCRIPTION	Many attacks (e.g., IMP01, BEH04) are based upon network traffic analysis. If an attacker gains access to comprehensive and multi-service network traces, detailed information about user behaviour can be gathered. Also open-redirection attacks and impersonation attacks are made more potent by the opportunity to contact unintended communication partners.
COUNTERMEASURES	With the rise of software-defined-networking (and prior to that with firewall rules sets) communication between services, virtual machines, and users can be restricted. This complicates data gathering by malicious actors and thus reduces the chance of successful attacks.
RELATED CATEGORIES	Behavior Analysis
RESPONSIBLE	Service Provider, Identity Provider, Storage Provider, Issuer
THREATS	Information disclosure, Linkability, Identifiability, detectability

DoS on crypto co-processor, HSM, etc.	
REFERENCE ID	ISO05
DESCRIPTION	While DoS attacks against common resources, i.e., network bandwith, CPU time, available memory and storage, are well known, cryptographic challenging use-cases introduce addiitonal attack vectors. Special hardware as HSM, crypto co-processors or random number generates are commonly not virtualized and thus are directly "forwarded" into virtual machines. If multiple virtual machines access the same hardware resource, race-conditions, timing- or side-channel attacks and DoS vectors can occur.
COUNTERMEASURES	If possible, cryptographic hardware must be bound for usage by only a single virtual machine. If this is not feasible, access control, auditing and rate-limits must be employed to limit an attacker's capabilities.
RELATED CATEGORIES	Behavior Analysis
RESPONSIBLE	Service Provider, Identity Provider, Storage Provider, Issuer
THREATS	Denial of service

4.4 Audit Trails

Auditing is a mandatory step in every ICT environment that poses high security requirements. Unfortunately, the extensive and reliable auditing and logging are often directly contradicting the users' need for privacy. On the one hand, the following vulnerabilities are therefore related to this trade-off. On the other hand, we discuss vulnerabilities that would lead to semantically incorrect audit trails, i.e., audit trails not reflecting the actual history of actions.

Logging contains privacy sensitive data	
REFERENCE ID	AUD01
DESCRIPTION	Logging information can contain sensitive information, e.g., which user did access which service and when. They are a prime target for attackers that need to gain input for behavioural analysis. Recent develoments, i.e., SIEM systems, centralize and protect logging/auditing information.
COUNTERMEASURES	Logging information should be scrubbed of sensitive information.
RELATED CATEGORIES	Behavior Analysis
RESPONSIBLE	Service Provider, Identity Provider, Storage Provider, Issuer
THREATS	Information disclosure, Linkability, Identifiability, detectability, non-compliance, non-repudiation
SPECIFIC TO	☑ IAM ☑ Data Sharing ☑ Privacy

Lack of integrity of audit logs	
REFERENCE ID	AUD02
DESCRIPTION	Audit trails are part of an user's provenance, thus can be used as input for a court of law. If so, the integrity of audit logs is paramount, i.e., no attacker should be able to alter logs in an undetected manner.
COUNTERMEASURES	Scrubbing of data before logging, providing confidentiality and integrity guarantees for audit data.
RELATED CATEGORIES	Behavior Analysis
RESPONSIBLE	Service Provider, Identity Provider, Storage Provider, Issuer
THREATS	Repudiation, non-compliance
SPECIFIC TO	☑ IAM ☑ Data Sharing ☐ Privacy

Unsound audit logs	
REFERENCE ID	AUD03
DESCRIPTION	Audit information must correctly represent executed actions, i.e., an attacker must not be able to create fake log information that implicates an otherwise innocent user.
COUNTERMEASURES	Usage of cryptographic protocols that verify that actions (as written down in the log file) have actually happened.
RESPONSIBLE	Service Provider, Identity Provider, Storage Provider, Issuer
THREATS	Tampering with Data, repudiation
SPECIFIC TO	☑ IAM ☑ Data Sharing ☐ Privacy

Insufficient access control on log files and backups	
REFERENCE ID	AUD04
DESCRIPTION	While audit logs do not represent the dynamic of a live system they allow reconstruction of near-realtime system states. This states can include sensitive user data—this mandates that all access to audit information must itself be logged and be subject to authentication and authorization checks. Otherwise an attacker could gain vital information by bypassing the "live" system and attacking the logging system itself.
COUNTERMEASURES	Access to audit information must be as protected at least as well as the original data was protected, i.e., authentication-, authorization- and auditing systems must be in-place.
RELATED CATEGORIES	Behavior Analysis
RESPONSIBLE	Service Provider, Identity Provider, Storage Provider, Issuer
THREATS	Information disclosure, non-repudiation, linkability, identifiability, detectability
SPECIFIC TO	☑ IAM ☑ Data Sharing ☑ Privacy

Service provider or IdP can prove user's behavior to 3rd parties	
REFERENCE ID	AUD05
DESCRIPTION	A service-, storage-, or identity provider might be able to prove to a third party that a user performed a specific action. However, the mere fact that a user accessed some resource might already cause negative impact for the user.
COUNTERMEASURES	Interactions can be done in a deniable way such that forwarding proofs is cryptographically impossible. However, such mitigation strategies might contradict the need for legally binding logs.
RELATED CATEGORIES	Behavior Analysis
RESPONSIBLE	Service Provider, Identity Provider, Storage Provider, Issuer
THREATS	Non-repudiation
SPECIFIC TO	☑ IAM ☑ Data Sharing ☑ Privacy

4.5 Authentication

Authentication vulnerabilities affect virtually any service, and are therefore very generic vulnerabilities. However, in the case of IAM they might have even higher potential impact than in other scenarios. This is because an exploit of such a vulnerability would not only allow the adversary to access a single service, but use the IAM provider to affect *any* service that is linked to the user's account. We therefore discuss some of the most fundamental authentication vulnerabilities in the following.

Weak authentication towards IAM or data sharing provider	
REFERENCE ID	AUT01
DESCRIPTION	An attacker will always use the easiest attack vector. If the overall system is written in a secure manner but users choose insecure passwords, brute-force attacks against the login system will succeed and thus lead to total compromise of an user's data.
COUNTERMEASURES	At a bare minimum, a carefully-chosen minimal password entropy must be enforced. In addition to pure password-based weak authentication methods, strong authentication—preferably multi-factor authentication—should be enforced. Possibilities include biometric authentication or time-based one-time passwords (TOTP).
RELATED CATEGORIES	Impersonation
RESPONSIBLE	User, Identity Provider, Storage Provider
THREATS	Elevation of privilege, spoofing identity

Weak authentication towards user	
REFERENCE ID	AUT02
DESCRIPTION	The service must authenticate itself against the user, otherwise a rogue service can convince users to enter sensitive data.
COUNTERMEASURES	A common way of verifying the server-side identity is TLS. In addition, many websites allow users to add customized welcome messages. These offer a technically simply yet working authentication mechanism. If the website is spoofed/faked by an attacker, the attacker does not know the greeting phrase—if the user detects the lack of the greeting phase, she should not enter sensitive (or any) data.
RELATED CATEGORIES	Impersonation
RESPONSIBLE	Identity Provider, Storage Provider, User
THREATS	Spoofing identity

Protocol features that allow DoS	
REFERENCE ID	AUT03
DESCRIPTION	Authentication protocols and implementations often come with mechanisms that protect against brute-force attacks. For example, an user account might be blocked after three invalid login attempts. An attacker can abuse those mechanisms to lock-out a valid user by entering wrong authentication data (for that user). While this vulnerability also affects virtually any other cloud-based service, its impact in IAM scenarions must not be underestimated. This is because if a user cannot authenticate towards the IdP anymore, it also cannot authenticate to any service provider any more.
COUNTERMEASURES	All mechanisms that prevent brute-force attacks must not be able to introduce DoS opportunities. For example, a blocked account must be able to be re-openend by the user. As there is an additional operation involved during re-opening this still thawes brute-force attacks. An alternative to account locks might be rate-limits or additional delays between log-in attempts.
RESPONSIBLE	Identity Provider, Storage Provider, Service Provider
THREATS	Denial of service
SPECIFIC TO	☑ IAM ☐ Data Sharing ☐ Privacy

Malicious identity issuer colluding with users	
REFERENCE ID	AUT04
DESCRIPTION	If a malicious (or compromised) identity issuer colludes with a user, the user might be able to impersonate other users. Furthermore, the user might be able to prove identity attributes about himself which are actually not true.
COUNTERMEASURES	This vulnerability cannot be tackled on a cryptographic level. A careful choice of identity issuers and continous auditing are recommanded.
RELATED CATEGORIES	Impersonation
RESPONSIBLE	Service Provider, Storage Provider
THREATS	Spoofing identity
SPECIFIC TO	☑ IAM ☐ Data Sharing ☐ Privacy

Provider accepting credentials issued by rogue issuer	
REFERENCE ID	AUT05
DESCRIPTION	If a service provider can be made to accept login requests based on credentials from corrupt issuers, a user might be able to access services without having a legitimate account.
COUNTERMEASURES	Issuer's public keys and certificates should be verified and checked for their revocation status on a regular basis.
RESPONSIBLE	Service Provider, Storage Provider
THREATS	Spoofing identity, information disclosure, tampering with data
SPECIFIC TO	☑ IAM ☑ Data Sharing ☐ Privacy

Weak authentication of service provider towards IdP	
REFERENCE ID	AUT06
DESCRIPTION	Not only it is important to have a mutual authentication between users and IAM or storage providers, but also the service provider must authenticate itself towards the identity providers. Otherwise, a malicious service provider could make the IdP forward user data by claiming another service provider's identity.
COUNTERMEASURES	TLS with strong mutual authentication can be used to secure the communication between service- and identity providers.
RESPONSIBLE	Identity Provider
THREATS	Information disclosure, Linkability, Identifiability, detectability, non-repudiation
SPECIFIC TO	☑ IAM ☐ Data Sharing ☑ Privacy

4.6 Access Control

Vulnerabilities due to a lack of appropriate access control mechanisms are inherent to every cloud-based application. We therefore omit such vulnerabilities here, and focus on somewhat subtle vulnerabilities, mainly related to privacy, that are important for data sharing scenarios. In particular, we discuss several vulnerabilities which arise when the storage provider has access to access control lists and can infer information from those.

Information only secured by ACL

REFERENCE ID	ACL01
DESCRIPTION	ACLs are commonly written to control access to (potentially sensitive) data. If the whole access concept consists of the server-side application controlling access, then side-channel attacks (e.g., by the infrastructure storage provider) on the underlying data are possible.
COUNTERMEASURES	ACLs must be implemented in a way, that the access control mechanism is inherently integrated into the stored data, i.e., that access to the data without also having access to the ACL "key" does not produce the original (plain-text) data.
RESPONSIBLE	Storage Provider
THREATS	Information disclosure
SPECIFIC TO	☐ IAM ☐ Data Sharing ☑ Privacy

ACLs are stored in plain text

REFERENCE ID	ACL02
DESCRIPTION	Access Control Lists detail which users may access which file. This mapping itself can be highly problematic, esp. when the files in question are of a senstive, e.g., medical, nature. An attacker can gain meaningful insights into the user by just just observing the access control lists and its content—this is traditional meta-data analysis.
COUNTERMEASURES	ACLs must utilize encryption in such a manner that the assignment of physical users to files must not be possible. In addition the ACL should give attackers no indication about controlled files and their contents.
RESPONSIBLE	Storage Provider
THREATS	Information disclosure, Linkability, Identifiability
SPECIFIC TO	☐ IAM ☐ Data Sharing ☑ Privacy

Unauthorized modification of ACLs

REFERENCE ID	ACL03
DESCRIPTION	If the attacker gains unauthorized write access to the ACL list, she can change the list to gain access to otherwise prohibited information or can lock-out other users from accessing their legitimate data.
COUNTERMEASURES	The ACL system itself must be subject of authentication, authorization and auditing. This implies that most vulnerabilities mentioned within this catalog should be also applied to the ACL data structure
RESPONSIBLE	Storage Provider
THREATS	Denial of service, information disclosure

Information disclosure when granting access to other users	
REFERENCE ID	ACL04
DESCRIPTION	If data-sharing is secured through ACLs or encryption is utilized to create such a system, the sharer and sharee's identities might be encoded into the shared data file. An attacker might extract this sensitive metadata from the shared data.
COUNTERMEASURES	Sharing schemes must make sure, that no sensitive metadata, e.g., sharer's and sharee's identity, are leaked during the sharing process or through generated artefacts.
RESPONSIBLE	Storage Provider
THREATS	Information disclosure, Linkability, Identifiability
SPECIFIC TO	☐ IAM ☐ Data Sharing ☑ Privacy

Unintended sharing of data or identity information	
REFERENCE ID	ACL05
DESCRIPTION	This vulnerability covers any unintentional sharing through the user. In particular, this covers sharing data with other users, or revealing unnecessary identity attributes to service providers.
COUNTERMEASURES	This vulnerability is hard to migrate. High usability is a key requirement to avoid unintentional data sharing.
RESPONSIBLE	User
THREATS	Information disclosure, unawareness, non-compliance
SPECIFIC TO	☐ IAM ☑ Data Sharing ☑ Privacy

Privacy can be abused to circumvent quota limitations	
REFERENCE ID	ACL06
DESCRIPTION	The idea of privacy and quote systems might be opposed to each other: the former tries to hide an file owner's identity while the latter tries to impose limits on the maximum amount of resources (e.g., storage capacity) a single owner can consume.
COUNTERMEASURES	Devise schemes that allow for quota enforcement while not disclosing a file's owner.
RESPONSIBLE	Storage Provider
THREATS	Denial of service
SPECIFIC TO	☐ IAM ☐ Data Sharing ☑ Privacy

4.7 Integrity and Consistency

Losing integrity is one of major concerns in every cloud-based system, and many relevant vulnerabilities can thus be found in generic cloud vulnerability catalogues, e.g., [Dan09, OWA12, BLSS11] We therefore only list those vulnerabilities that are of utmost importance (in particular) for data sharing scenarios.

Inconsistent data due to parallel accesses	
REFERENCE ID	INT01
DESCRIPTION	Parallel access to shared data must be synchronized, otherwise data consistency can be damaged through race conditions. An attacker can utilize those race conditions to introduce maliciously altered data into the system. In addition race conditions can not only occur between data operations but also between data- and metadata-parts of the same operation.
COUNTERMEASURES	An accepted operation synchronization directive needs to be employed for operation ordering. Possible candidates are paxos, raft, or one of the multiple BFT variants.
RESPONSIBLE	Storage Provider
THREATS	Tampering with Data
SPECIFIC TO	☐ IAM ☑ Data Sharing ☐ Privacy

Data loss due to merge conflicts	
REFERENCE ID	INT02
DESCRIPTION	If the data store utilized versioning, i.e., fork consistency, new file versions can hide concurrent changes performed by slightly older file versions. An attacker could utilize this to hide malicious changes to files.
COUNTERMEASURES	All merge conflicts must be presented to clients; the final decision about merged versions must be delegated to client ruling.
RESPONSIBLE	Storage Provider
THREATS	Tampering with Data
SPECIFIC TO	☐ IAM ☑ Data Sharing ☐ Privacy

4.8 Confidentiality

We address the loss of confidentiality as one of the core CREDENTIAL objectives to the protect the information from unauthorized disclosure. The impact of unauthorized disclosure of confidential information can range from the jeopardizing of nonpublic, personal private information to loss of competitive advantage or trade secret information.

Existence of crypto backdoors / master secret keys /...	
REFERENCE ID	CON01
DESCRIPTION	During the last decades, the governments of multiple countries tried to introduce backdoor functionality into various parts of vital cryptographic subsystems. Those attempts ranged from backdoored random number generators, existence of backup secret keys or mandatory backdoor integration. Those additions neither improve the security or privacy of the designed software component, system nor society but introduce potential attack vectors that can be exploited by rogue agents.
COUNTERMEASURES	Make sure that proper cryptographc algorithms with no (known) backdoor have been used. In case of mandated backdoors, inform users. If user notification is prohibited by law, employ canaries.
RESPONSIBLE	Storage Provider, Issuer, Identity Provider, Service Provider
THREATS	Information disclosure, Linkability, Identifiability, Non-repudiation, non-compliance

Insecure deletion	
REFERENCE ID	CON02
DESCRIPTION	In particular in the case of hardware decommissioning, insecurly deleted data might lead to excessive data leaks. This not only affects harddisks, but also every device with internal buffers such as printers, etc.
COUNTERMEASURES	Make sure that sensitive data is deleted before hardware decommissioning. Utilize cryptography, so all stored data is unusable without knowledge of an (non-persisted) key.
RESPONSIBLE	Identity Provider, Storage Provider, Service Provider, Issuer
THREATS	Information disclosure, Linkability, Identifiability, detectability, Non-repudiation, non-compliance

Insecure temporary files	
REFERENCE ID	CON03
DESCRIPTION	During normal operation a multitude of temporary files are created, e.g. during file-upload, printing, or high-memory utilization situations. These files can be extracted by malicious actors and searched for sensitive information. In addition, if temporary files are read back by the application, an attacker might be able to guess the next temporary filename and introduce malicious data through a custom created temporary file (with that name) into the application logic.
COUNTERMEASURES	All temporary files must not be accessible by the outside world. If this is not feasible due to the use-case, the ACL system must police all access to those temporary files. In addition all data within those files should be encrypted and their filename should be created using secure random filename generators.
RESPONSIBLE	Identity Provider, Storage Provider, Service Provider, Issuer
THREATS	Information disclosure, Linkability, Identifiability, detectability, non-repudiation

Long-term insecurity of cryptographic methods	
REFERENCE ID	CON04
DESCRIPTION	Data that has been leaked, published, or stored off-site can no longer be deleted by the legitimate data owner. It must therefore be assumed that any such data will be analyzed or target to attacks also in the far future. In particular, future advances in cryptanalysis (e.g., by the advent of large-scale quantum computers, development of improved algorithms, increase of an attacker's computing power) might be used to break current cryptographic schemes.
COUNTERMEASURES	The long-term confidentiality requirements of all files need to be clearly defined before storing them off-site. Depending on those, appropriate cryptographic algorithms (e.g., based on "quantum-safe" primitives) and key sizes should be chosen in a very conservative way.
RESPONSIBLE	Service Provider, Identity Provider, Storage Provider, Issuer, User
THREATS	Information disclosure, Linkability, Identifiability, Non-repudiation
SPECIFIC TO	☑ IAM ☑ Data Sharing ☐ Privacy

Server-side data leaks	
REFERENCE ID	CON05
DESCRIPTION	Data leaks caused by the storage provider, e.g., due to misconfiguration, hacks, human error, or bribary are one of the main risks to data confidentiality.
COUNTERMEASURES	The risk of data leaks can be reduced by securly encrypting data before storing it in the cloud. Furthermore, storage providers can introduce comprehensive assurance loops and internal logging to avoid unintended data leaks.
RESPONSIBLE	Storage Provider
THREATS	Information disclosure, Linkability, Identifiability, Non-repudiation
SPECIFIC TO	☑ IAM ☑ Data Sharing ☐ Privacy

Insecure (external) backups by storage provider	
REFERENCE ID	CON06
DESCRIPTION	Backup systems are employed to store copies of multiple versions of the "live" data. An attacker might gain access to the backup data store and extract sensitive data from this backup store.
COUNTERMEASURES	In addition to live and audit data, backup data must also be protected with the same vigilance as the former data and in particular be encrypted using strong cryptographic schemes.
RESPONSIBLE	Storage Provider
THREATS	Information disclosure, Linkability, Identifiability, Non-repudiation
SPECIFIC TO	☐ IAM ☑ Data Sharing ☐ Privacy

Colluding issuer and service or identity provider	
REFERENCE ID	CON07
DESCRIPTION	If the issuer colludes with the service- or identity provider, they could pool their data in order to violate the user's privacy. For instance, the service provider could hand to the issuer the revealed identity information, and the issuer could amend it by the information that was revealed upon issuance; on the other hand, the issuer would learn behavior patterns of the user.
COUNTERMEASURES	On the one hand, this vulnerability can be addressed by appropriate SLAs, and by verifying that issuer and service- or identity provider are independent legal entities. On the other hand, cryptographic mechanisms can be used to guarantee unlinkability of authentications towards a service provider and issuances (except via the information that was revealed to both parties).
RESPONSIBLE	Issuer, Service Provider, Identity Provider
THREATS	Information disclosure, Linkability, Identifiability, Non-repudiation
SPECIFIC TO	☑ IAM ☑ Data Sharing ☑ Privacy

Colluding service- and identity provider	
REFERENCE ID	CON08
DESCRIPTION	If the identity provider colludes with a service provider (or both are corrupted by the same adversary), he can learn significant amounts of information and identity attributes even if all data is encrypted. This is because the service provider needs to be able to access certain information about the user during the authentication process, and can then hand this information back to the identity provider. Furthermore, the identity provider could reveal more information to the service provider than intended by the user.
COUNTERMEASURES	This vulnerability cannot be tackled on a cryptographic level. However, the non-collusion of the identity provider with other entities can be enforced by service level agreements (SLAs), and by external audits that they are satisfied. A user should carefully choose the providers and make sure that the different providers are independent legal entities in order to reduce the risk.
RELATED CATEGORIES	Behavior Analysis
RESPONSIBLE	Service Provider, Identity Provider
THREATS	Information disclosure, Linkability, Identifiability, Non-repudiation
SPECIFIC TO	☑ IAM ☐ Data Sharing ☑ Privacy

Colluding storage provider and data receiver			
REFERENCE ID	CON09		
DESCRIPTION	This vulnerability is similar to CON08, except that the storage provider colludes with the (legitimate) receiver of the data.		
COUNTERMEASURES	Analogeous contermeasured to CON08 apply.		
RESPONSIBLE	Storage Provider		
THREATS	Information disclosure		
SPECIFIC TO	☐ IAM	☑ Data Sharing	☑ Privacy

5 Conclusion

The core objective of this deliverable is outlined as the methodology for identifying most relevant security concerns in from of vulnerabilities with regards to the CREDENTIAL wallet data-sharing and identity management platform. The deliverable provides a central repository of vulnerabilities that are specific to those applications. Besides describing the vulnerabilities in detail, we also map them to the corresponding related threat categories of DREAD and LINDDUN, cf. Section 3. This deliverable will be used as the foundation for further risk assessment processes and second iteration of system security requirements analysis that will be conducted within the scope of the CREDENTIAL project.

List of References

[ACF+16] Andreas Abraham, Pasquale Chiaro, Sara Faccinetti, Felix Hörandner, Stephan Krenn, Andrea Migliavacca, Nicolás Notario McDonnell, Anna Schmaus-Klughammer, Evangelos Sfakianakis, Florian Thiemer, Alberto Zanini, and Juan Carlos Pérez-Baún. D2.1 Scenarios and Use-Cases. Technical report, CREDENTIAL Project Deliverable, 2016.

[AHH+16] Andreas Abraham, Andreas Happe, Aleksandar Hudic, Stephan Krenn, Nicolás Notario McDonnell, Christoph Striecks, and Florian Thiemer. D2.2 System Security Requirements, Risk and Threat Analysis. Technical report, CREDENTIAL Project Deliverable, 2016.

[BBS98] Matt Blaze, Gerrit Bleumer, and Martin Strauss. Divertible Protocols and Atomic Proxy Cryptography. In Kaisa Nyberg, editor, *Advances in Cryptology - EUROCRYPT '98*, volume 1403 of *LNCS*, pages 127–144. Springer, 1998.

[BLSS11] Jerry Busby, Lucie Langer, Marcus Schöller, and Paul Smith. Deliverable 3.1-methodology for risk assessment and management. *Journal of Network and Computer Applications*, 34(4):1097–1107, 2011.

[BSI16] BSI. B 1.18 Identitäts- und Berechtigungsmanagement. https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/_content/baust/b01/b01018.html?nn=6604926, 2016. accessed: 2016-09-27.

[Dan09] Daniele Catteddu and Giles Hogben. Cloud Computing Benefits, risks and recommendations for information security. Technical Report, November 2009.

[JMSW02] Robert Johnson, David Molnar, Dawn Xiaodong Song, and David Wagner. Homomorphic Signature Schemes. In Bart Preneel, editor, *Topics in Cryptology - CT-RSA 2002*, volume 2271 of *LNCS*, pages 244–262. Springer, 2002.

[Mic05] Microsoft. Threat modeling web applications. https://msdn.microsoft.com/library/ms978516.aspx, May 2005. accessed August 2016.

[Mic16] Microsoft. Microsoft threat modeling tool 2016. https://www.microsoft.com/en-us/download/details.aspx?id=49168, 2016.

[MMD+03] J.D. Meier, A. Mackman, M. Dunner, S. Vasireddy, R. Escamilla, and A Murukan. *Improving web application security: threats and countermeasures roadmap*. Microsoft, 2003.

[OWA12] OWASP. Vulnerability Catalog. Technical Report, September 2012.

[SBZ01] Ron Steinfeld, Laurence Bull, and Yuliang Zheng. Content Extraction Signatures. In Kwangjo Kim, editor, *Information Security and Cryptology - ICISC 2001*, volume 2288 of *LNCS*, pages 285–304. Springer, 2001.

[Shi13] Robert W. Shirey. Internet Security Glossary, Version 2. RFC 4949, March 2013. https://rfc-editor.org/rfc/rfc4949, accessed August 2016.

[ZMW14] Jan Henrik Ziegeldorf, Oscar García Morchon, and Klaus Wehrle. Privacy in the internet of things: threats and challenges. *Security and Communication Networks*, 7(12):2728–2742, 2014.

A Appendix

The following table providers an overview of all vulnerabilities identified in this document, and shows the actors responsible to take mitigation strategies.

ID	Title	User	Service Provider	Identity Provider	Storage Provider	Issuer
IMP01	Man-in-the-middle attack on network level	x	x	x	x	x
IMP02	Unintended session re-use	x	x	x	x	x
IMP03	Compromise of identity information	x	x	x	x	x
IMP04	Insecure session Ids	x	x	x	x	x
IMP05	Identity assurance					x
IMP06	Delayed or missing revocation	x	x			x
IMP07	Open re-directs within protocols		x	x	x	
IMP08	Malicious group administrator			x		
BEH01	User behavior analysis by the Identity Provider			x		
BEH02	Service provider can link user actions			x		
BEH03	Network traffic can be linked by outsiders			x	x	
BEH04	Confidentiality compromise through metadata analysis			x	x	
BEH05	Excessive logging on different layers		x	x	x	x
ISO01	Measurement attacks between services		x	x	x	x
ISO02	Data extraction between services		x	x	x	x
ISO03	Shared data between services		x	x	x	x
ISO04	Lack of network traffic isolation		x	x	x	x
ISO05	DoS on crypto co-processor, HSM, etc.		x	x	x	x
AUD01	Logging contains privacy sensitive data		x	x	x	x
AUD02	Lack of integrity of audit logs		x	x	x	x
AUD03	Unsound audit logs		x	x	x	x
AUD04	Insufficient access control on log files and backups		x	x	x	x
AUD05	Service provider or IdP can prove user's behavior to 3rd parties		x	x	x	x
AUT01	Weak authentication towards IAM or data sharing provider	x		x	x	
AUT02	Weak authentication towards user	x		x	x	
AUT03	Protocol features that allow DoS		x	x	x	
AUT04	Malicious identity issuer colluding with users		x		x	
AUT05	Provider accepting credentials issued by rogue issuer		x		x	
AUT06	Weak authentication of service provider towards IdP			x		
ACL01	Information only secured by ACL				x	

ID	Title	User	Service Provider	Identity Provider	Storage Provider	Issuer
ACL02	ACLs are stored in plain text				x	
ACL03	Unauthorized modification of ACLs				x	
ACL04	Information disclosure when granting access to other users				x	
ACL05	Unintended sharing of data or identity information	x				
ACL06	Privacy can be abused to circumvent quota limitations				x	
INT01	Inconsistent data due to parallel accesses				x	
INT02	Data loss due to merge conflicts				x	
CON01	Existence of crypto backdoors / master secret keys /...		x	x	x	x
CON02	Insecure deletion		x	x	x	x
CON03	Insecure temporary files		x	x	x	x
CON04	Long-term insecurity of cryptographic methods	x	x	x	x	x
CON05	Server-side data leaks				x	
CON06	Insecure (external) backups by storage provider				x	
CON07	Colluding issuer and service or identity provider		x	x		x
CON08	Colluding service- and identity provider		x	x		
CON09	Colluding storage provider and data receiver				x	

Table 5: Overview of identified vulnerabilities and mapping to targets