# D5.2 Description of the "Common Denominator" Elements

*Joerg Abendroth, Souheil Bcheri, Norbert Götze, Vasiliki Liagkou, Monika Orski, Robert Seidl, Fatbardh Veseli*


| | |
|---|---|
| *Editors:* | *Joerg Abendroth (Nokia Siemens Networks)* |
| *Reviewers:* | *Ahmad Sabouri (Goethe University Frankfurt), Gert Læssøe Mikkelsen (Alexandra Institute A/S)* |
| *Identifier:* | *D5.2* |
| *Type:* | *Deliverable* |
| *Version:* | *1.0(formatted)* |
| *Date:* | *04/09/2012* |
| *Status:* | *Final* |
| *Class:* | *Public* |



Abstract

ABC4Trust integrates different privacy-ABC Technologies to provide privacy protection in Internet communications. To evaluate the privacy-ABC Engine API, two pilots are implemented. The greater aim of ABC4Trust is to make privacy-ABC Technology useable for wide use, beyond the two pilots. This deliverable contributes to the greater aim by identifying and describing the common denominators of those pilots. The intended reader, a prospective company planning to utilize privacy-ABC Technology, gets an introduction how the two pilots use privacy-ABC technology, along with pointers to information. It also documents the overall hardware and network requirements for deploying the ABC technologies, using the pilots as examples. Additional remarks hopefully prove beneficial to define the roles and business models of prospective privacy-ABC technology users.


*The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 257782 for the project Attribute-based Credentials for Trust (ABC4Trust) as part of the "ICT Trust and Security Research" theme.*

# Members of the ABC4Trust Consortium

| | | | |
|---|---|---|---|
| 1. | Alexandra Institute AS | ALX | Denmark |
| 2. | CryptoExperts SAS | CRX | France |
| 3. | Eurodocs AB | EDOC | Sweden |
| 4. | IBM Research – Zurich | IBM | Switzerland |
| 5. | Johann Wolfgang Goethe – Universität Frankfurt | GUF | Germany |
| 6. | Microsoft Research and Development | MS | France |
| 7. | Miracle A/S | MCL | Denmark |
| 8. | Nokia-Siemens Networks GmbH & Co. KG | NSN | Germany |
| 9. | Research Academic Computer Technology Institute | CTI | Greece |
| 10. | Söderhamn Kommun | SK | Sweden |
| 11. | Technische Universität Darmstadt | TUD | Germany |
| 12. | Unabhängiges Landeszentrum für Datenschutz | ULD | Germany |

# List of Contributors

| Chapter | Author(s) |
|---|---|
| Executive Summary | Joerg Abendroth (NSN) |
| 1. Introduction | Joerg Abendroth (NSN), Robert Seidl (NSN) |
| 2. Overview Pilots | Joerg Abendroth (NSN), Souheil Bcheri (EDOC), Maxim Moneta, (EDOC), Monika Orski (EDOC) , Fatbardh Veseli (GUF), Vasiliki Liagkou (CTI) |
| 3.Common Denominator Elements | Joerg Abendroth (NSN), Fatbardh Veseli (GUF) |
| 4. Optional Generic Elements | Joerg Abendroth (NSN), Norbert Götze (NSN) |
| 5. Hardware Requirements | Norbert Götze (NSN) |
| 6. Outlook | Joerg Abendroth (NSN), Robert Seidl (NSN) |

# Executive Summary

*The ABC4Trust project's main objective is twofold: (i) the definition of a unified reference architecture for the systems deploying Privacy-enhancing Attribute-Based Credentials (Privacy-ABCs) and (ii) the development of an open reference implementation of a full system deploying Privacy-ABCs that will be integrated into two complete real pilot applications, providing feedback to the reference architecture and implementation results. These will be the first pilots of Privacy-ABC deployments in real application environments for collecting feedback.*

*This deliverable provides information for prospective users (e.g. application developers) of the Privacy-ABC technology. It references to the other deliverables in this project, to include less technical details, but explains the information to the readers that are in the process of deciding how to deploy Privacy-ABC technology within their scope of business.*

*Along with a brief introduction of the architecture for the two pilots and references to the relevant other deliverables, common denominator components are identified. The common denominators are reviewed with their relation to the existing state-of-art technologies and wider ecosystems. Prospective users can find remarks on how to successfully implement the role of a certain common denominator component as well as necessary network and hardware requirements. .*

# Table of Contents

# Index of Figures

# Index of Tables

# 1 Introduction

The objective of this project, ABC4Trust, is (1) to define a common, unified architecture for Privacy-ABC systems to allow comparing their respective features and combining them on common platforms, and (2) to deliver open reference implementations of selected Privacy-ABC systems and deploy them in actual pilots, for example allowing provable accredited members of restricted communities, or to provide anonymous feedback.

This deliverable reviews the components of the two production pilots with respect to the architecture and scenarios, and identifies common denominator components along with additional beneficial generic components and recommendations for future users of the ABC4Trust technology.

## 1.1 Ecosystem of ABC4Trust versus state of the art

The simplest online identification scenario is if a service provider does not rely on any external party (e.g. Identity Provider) and keeps all the data about the users in their own database. This has the disadvantage of requiring each user to create an account with this service provider, or not being able to customize their services to the users that have not registered with them. Hence the Identity Management system (IDM) has the role of storing common user attributes and facilitating sign on to different service providers (in the case of single-sign-on). Service Providers, Users, Infrastructure Providers and Regulators that currently do not use ABC4Trust technology will likely deploy a standard Identity Management (IDM) Ecosystem, which will be introduced below.

As it is shown if figure 1, the state-of-the art IDM Ecosystem consists of a minimum of three parties:

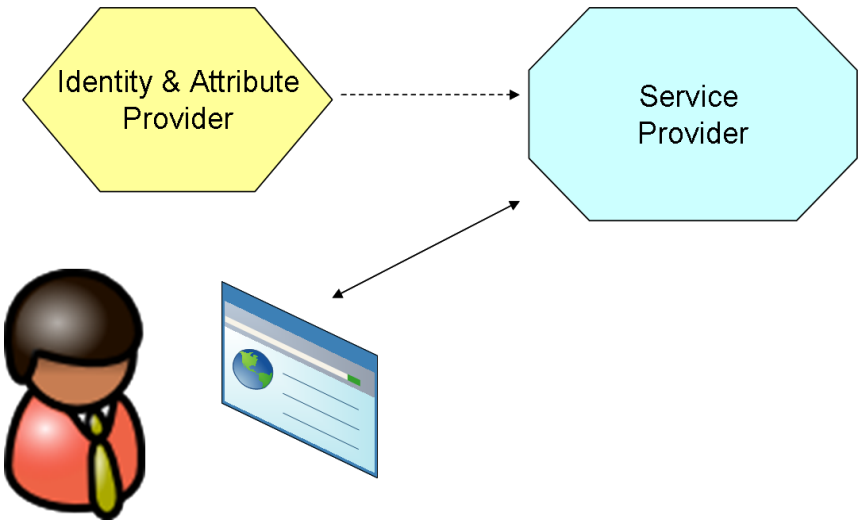


**Figure 1- Basic IDM System**

- The Identity provider, which knows the user.
- The service provider, which needs to authenticate the user (or verify an attribute).
- The User and the user client.

Typical scenarios, such as login to a webshop, include the user having a profile at the IDM and wanting to use the service of the service provider (e.g. shop in the webshop). She then would need to authenticate to the service provider, either for billing or to make data of her profile available (e.g. delivery address) to the service provider. The service provider is required to trust the IDM, which provides an attribute assertion about the user. The user needs to cooperate by linking his IDM account with the Service Provider, or it is also possible that the accounts at the service provider are created on the fly without a lengthy registration process.

Another scenario is targeted advertisement. In that scenario no strong authentication is done, identifying the user is sufficient. Often this process of identification is performed without requesting the users' help, but based on selected attributes (e.g. IP address, contact email, birthday, delivery address).

Due to the strong incentive of businesses to identify the users for service customisation, protection of their privacy is necessary. Also in scenarios where users participate in online voting or sensitive discussions it is mandatory to employ a system that protects their identity. Different privacy enhancing technologies that support minimal disclosure exist. The contribution of ABC4Trust project is to present a system that provides one API interface to the different underlying technology implementations.

The ABC4Trust system allows users to prove some facts about their attributes even without revealing the attribute values or their real identity. For this the IDM takes on the role of Issuer. Similar to X509 systems the Issuer provides the user with a credential, which can be used to prove possession of an attribute. New is, that the credential can also be partially shown, that is not all the attributes of the credential have to be revealed by the user – this is called selective disclosure. The process in showing the attributes to a service provider is called verification. It is possible that the user shows an attribute to two different service providers and they cannot infer that it was the same user, not even the Issuer can derive this information from a verification run.

On the business ecosystem there are only small changes between state-of-the art IDM systems and ABC4Trust systems, but on the technology side the differences are considerable. The Privacy-ABC Technology uses credentials (and not certificates or assertions). A credential is a container of certified values of attributes about an identity plus some credential-specific attributes, signed by the Issuer of the credential(s). By utilizing credentials the linking power of the IDM and the service providers will be limited, which is called unlinkability. The user has also the choice to reveal only selected attributes, which is called partial or selective disclosure. Finally and most important during the verification process the Issuer in the ABC4Trust system is not involved, thus does not know how often a specific attribute is being revealed (and to whom). This is a main advantage over the state-of-the-art IDM.

The standard role of IDM maps to the role of Issuer. The service providers can choose to employ a relying party secure token service as proxy or implement the verifier themselves. Revocation exists also in both systems. The ABC4Trust technology introduces a new powerful role of Inspector, which can lift anonymity if asked to. Inspection can only be done if it was stated in the verification policy. Thus the users will know in advance whether a verification process can end up in inspection of attribute values or not.

For more Information on state-of-the-art IDM Ecosystem please refers to Radhakrishnan [RaRa2007]. A good introduction to Standardization and Regulatory issues can be found in Bramhall et al. [BHRR07] or the EU primer for policy makers [DSTICICC09].

## 1.2 Dependencies with contributing Tasks

As described above this deliverable reviews the components of the two production pilots with respect to the architecture and scenarios and identifies common denominator components along with additional beneficial generic components.

This deliverable is mainly dependent on the following activities within the ABC4Trust project:

- Task 5.4 – Definition of hardware requirements
- Task 5.5 – Description of the common denominator of the scenarios
- Task 5.6 – Definition of the required information storage
- WP6 Söderhamn (Sweden) pilot specification and development
- WP7 Patras (Greece) pilot specification and development

This deliverable reflects the actual status of the project. However, it cannot cover the holistic view due to the ongoing specification and development of the pilots. Therefore, D5.2 presents a snapshot of the current activities.

For the Patras pilot, the initial user friendliness test is ongoing and based on this test D5.2 will provide a quite complete "Description of the 'Common Denominator' Elements". Whereas for the Söderhamn pilot the definition phase is still ongoing (in line with the time line in the DoW) so that this deliverable can only reflect as much as possible at the actual progress (e.g. for the hardware requirements) in WP6 (Swedish pilot).

## 1.3 Focus of this Deliverable

This deliverable will present a valuable input to ABC4Trust external parties that wish to utilize ABC4Trust technology. Similar to our two pilots, they will have their business (use) case in mind, but no thorough understanding of the Privacy-ABC technology. Yet the split of different functions, such as business logic, credential issuing and verification, and different administrative tasks is important for success. Additional questions, like which generic components or computing platform are needed are also of importance.

This deliverable will neither be a detailed explanation of the two pilots (for this please refer to D6.1 and D7.1), neither presents, nor will it present every feature that the ABC Engine[1] is capable of performing. For readers that require advanced ABC Engine knowledge D2.1 is recommended.

Chapter 2 provides an overview of the two Pilots, so to allow understanding the architecture and functions. Chapter 3 introduces the Common Denominator Elements that are to be implemented by all the use cases. Additional Generic elements that will touch a large number of use cases are described in Chapter 4. Chapter 5 describes technical requirements to utilize ABC4Trust technology and finally Chapter 6 gives an outlook.

[1] ABC Engine (or ABCE in its short form) is a core component of ABC4Trust architecture that provides technology-agnostic interface to the applications for Privacy-ABC operation.

# 2 Overview of the two Pilots

The two pilots are independent systems that do not share any actual component, except the ABC4Trust Engine, which is being used in both pilots without modifications. Yet both their use cases require privacy.

- The Patras pilot is about students collecting attendance credentials while attending lectures and then evaluating the course at the end of semester.
- The Söderhamn pilot is about privacy protected online communication, where in some chat rooms only students of a certain age or group can participate.

In the following the two pilot systems are described. Then we will focus on the common denominator components. The common denominators are the components that any Ecosystem utilizing the ABC4Trust technology will need to implement.

Additional generic components that may be beneficial, but did not turn out to be common denominators in our two pilots, will also be described.

The purpose of this description is to provide an illustrating example and establish the architecture commonly used with the Privacy-ABC technology. Please refer to D6.1 and D7.1 for an in depth technical descriptions of the pilots.

## 2.1 Patras Pilot (Course Rating by Certified Students)

Course evaluations have become standard practice in most universities around the world. However they are typically conducted on paper to protect the students' privacy. In cases where they are conducted through computers, the computers are operated by a neutral trusted organization independent from the school doing the evaluation; otherwise the students need to put a lot of trust in the fairness and privacy practices of their university.

The Patras pilot addresses this special challenge: For important and influential results of electronic course evaluation to be correct and credible, the privacy of the people expressing their opinion must be preserved. Therefore, Privacy-ABC technology is employed to guarantee that an eligible student participates in the course evaluation anonymously without revealing his private and personal information. At the same time, the system must guarantee that only eligible students can have access to the evaluation of a course. That is, the system must first verify that a student (1) is enrolled in the university, (2) has registered to the course and (3) has attended a sufficient number of the lectures of that course.

To satisfy the above requirements, each student obtains a smart card, which is used to receive Privacy-ABCs credentials, issued by the university. The students will use these credentials at the end of the semester to prove the desired properties, e.g. verify their enrolment in the university and the course they have registered for, without revealing their identity. The students utilize the same smart card to anonymously collect evidence for their class attendance during the semester by waving the card in front of a NFC device installed in the lecture room. At the end of semester, they anonymously authenticate from their PCs to the online evaluation page of the corresponding course, by combining the credentials they have collected. The technology behind the scene does not allow the card owners to exchange their obtained credentials or submit more than one final evaluation for the same course.

As a result of this technology, universities will be able to run their own online course evaluation systems that increase the trustworthiness between the students and the university. The goal of the pilot

is to gather information on the reactions of a typically critical group of users. Most importantly, it will provide concrete feedback on the user acceptance and usability of the technology, something that has not been done for this technology before.

### 2.1.1 Patras Pilot Components

The Patras pilot consists of three different systems (University Registration System, Course Evaluation System and User Client), which all include a Privacy-ABC System component. Figure 2 gives an overview of the components and the subsystems they belong to.

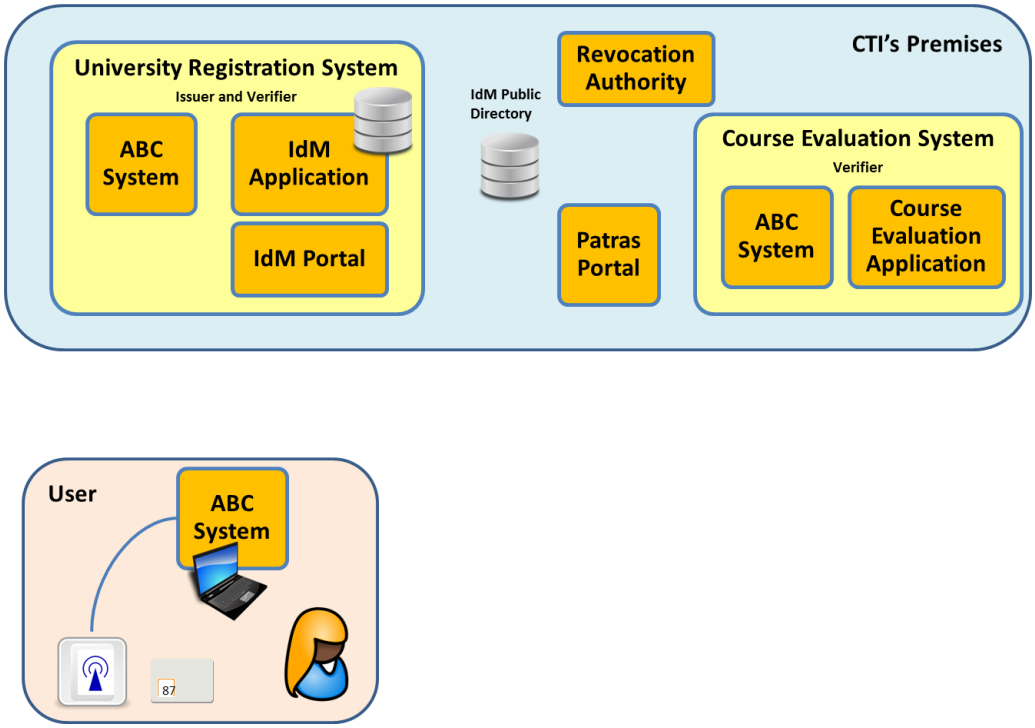


**Figure 2: Patras Pilot Components**

More details about each subsystem can be found in the following sections describing the details.

#### 2.1.1.1 University Registration System

It is a rational expectation from the students' perspective that the Privacy-ABC enabled course evaluation system appears seamlessly incorporated into the existing university IT facilities. This is due to the fact that the pilot is not separated from the university. More specifically, the corresponding course is part of the required curriculum and the pilot participants do not differentiate it from the other courses that they attend during the semester. As a result, they will need less effort to adopt the pilot application if it is aligned with the current university systems.

For the above to be true, the University Registration system of the pilot has to integrate with the greater university system, at least in such a way that the students can utilize their student number (or matriculation number) for the initial login, find the one evaluation courses and be able to register for it.

Main function of the university registration system is not to provide an anonymous administration of the students' identity, which is known as soon as the student number is known, but to bootstrap the anonymous system, that is to issue credentials.

#### 2.1.1.1.1 Application Architecture and Interfaces

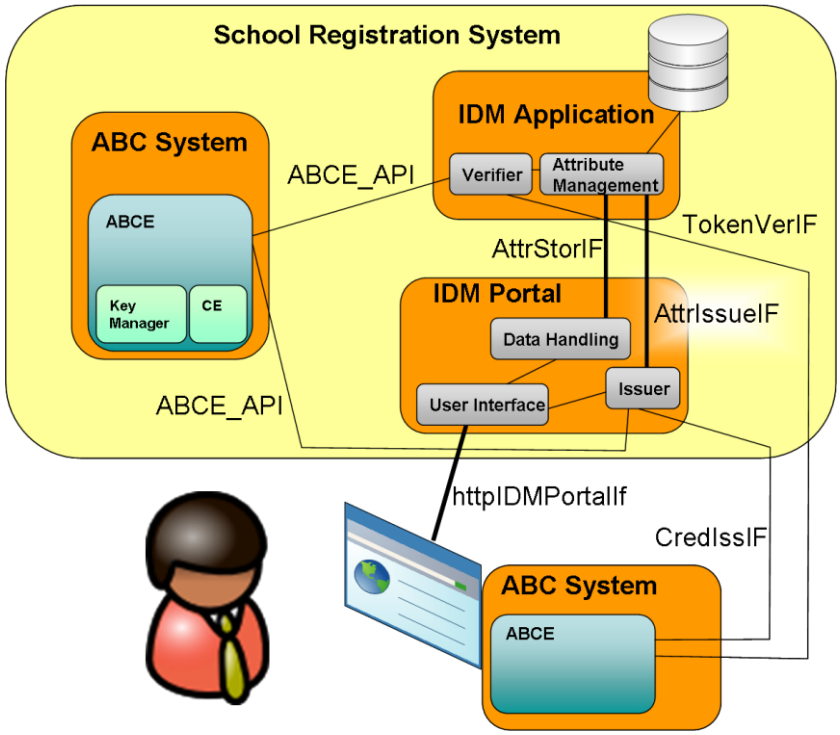


**Figure 3: University Registration System**

The functional components of IDM application and Privacy-ABC Systems are cleanly separated, as shown in Figure 3. The interface between both components has been described in Deliverable 2.1.

The IDM Portal is the self-administration portal of the registration system. It is also the place where the user gets new credentials issued. Authentication toward the portal can be done by student number along with a one-time password, or other means. Later authentication with ABC4Trust credentials is also possible. However, as the IDM Portal supports use cases such as "Broken Smart Card", there needs to be other authentication methods than ABC4Trust credentials. The IDM Portal is the frontend, which does not store any data, but relies on the IDM Application to provide attributes and verify attributes.

The IDM Application has two purposes: (1) to store and provide attributes, which is not done based on Privacy-ABC mechanisms; for example storing and retrieving students' addresses or classes. And (2) is to provide Privacy-ABC operations for verifying tokens or attributes, which is equivalent to authentication in a standard IDM Ecosystem.

The interface to fulfil the first purpose may utilize a custom protocol since it is University Registration system internal. But for larger or multi-vendor installations standard protocols are needed. In this regard, SAML, OAuth or in certain cases OpenID are good candidates.

Concerning the second purpose, verifying attributes and tokens requires two separate interfaces; one for receiving the Privacy-ABCs token and another for providing the result of the evaluation to an external service. However, in this pilot the IDM portal is the only recipient of the verification results.

However, in other scenarios external recipients are conceivable, e.g. when the IDM application acts as a Relying Party Token Proxy Service for some external parties.

Below there is a table listing the interfaces, the protocol used and the two endpoints of the interface. The Interface names refer to Figure 3. The interfaces can be internal and external. Interfaces that touch the ABC Engine, are implemented by ABC4Trust WP4 (Reference Implementation) and do not concern the developers.

**Table 1 University Registration System IDM Interfaces**

| Name | Applications that share this interface | Protocols used | Purpose |
|---|---|---|---|
| **AttrIssueIf** | **IDM Application – IDM Portal** | | **Storage of attributes, e.g. changed home address.** |
| **AttrStoreIf** | **IDM Application – IDM Portal** | | **Retrieval of Value for issuing a credential** |
| **CredIssIf** | **IDM Portal– User Client** | | **Providing the user with a credential.** |
| **TokenVerIf** | **IDM Application – User Client** | | **User client providing the token to be verified** |
| **httpIDMPortalIf** | **IDM Portal – Web Browser** | **http** | **To view and administrate the user's data using standard web browser.** |
| **ABCE_API** | **ABC System – Verifier of IDM Application** | **API calls** | **To verify tokens** |
| **ABCE_API** | **ABC System – Issuer of IDM Portal** | **API calls** | **To issue tokens** |

#### 2.1.1.2 Revocation Authority

Besides issuing credentials, a proper authentication system should also support revocation of issued credentials. Revocation is the process which ends the validity of a credential. In the ABC4Trust architecture, revocation is the responsibility of an entity known as a Revocation Authority (RevAuth), which may be the Issuer or the Verifier itself, or a separate entity dedicated for revoking credentials.

The Revocation Authority is responsible for revoking credentials (Privacy-ABCs). In the Patras pilot, this role is performed by the University Registration Office, which is also the Issuer of such credentials (see deliverable *"D7.1 - Application Description for Students"*). The administration office of the university can accept revocation requests from "Revocation Requesters" or act as a Revocation Requester on its own, and process them adequately. Finally, it should make the latest revocation information available to stakeholders.

The identified reasons for revoking students' Privacy-ABCs in the Patras pilot include cases when:

- A student leaves the university (e.g. not turning up).

- A student leaves the pilot, withdrawing the previously given consent to take part in it.
- A student loses possession of her smart card, i.e. if the smart card has been stolen or lost.
- A student gets a new credential replacing the old one.

The pilot will take care to properly maintain not only the process of revocation, but also comply with the legal requirements concerning the storage and deletion of personal data. After revocation, a specified data retention period of time will be allocated, during which data related to the revoked credentials will be stored and after which they will be deleted. If revocation is not followed by re-issuance of the credential, e.g. as the student leaves the university or has withdrawn consent, the data in the IDM database must be deleted within due time (without remainders of data copies, e.g. from backups). If protocol and log files are necessary for legal/audit purposes this information may be processed only for purposes of ensuring IT security.

The pilot will put the necessary efforts to perform a timely revocation of credentials and timely dissemination of revocation information to the stakeholders. This measure is necessary to avoid impersonation attacks or other types of credential misuse, in case a smart card containing the credential(s) is reported stolen and needs immediate revocation.

Finally the principal (user) whose credential is revoked will be informed of the revocation, and if necessary, the grounds for such an action.

*At the current state of the project the interfaces are not defined yet.*

#### 2.1.1.3 Patras Portal

Patras Portal is an information web portal. All User groups can access it through a standard browser. The main role of Patras Portal is to inform the users about the Course Evaluation for certified students. Moreover every time a user wants to interact with the Patras pilot, her first action is to visit this portal, and by following the instructions she can perform various pilot operations (e.g. register to a course, evaluate a course). Thus, this web portal will provide to the users the following links:

1. University Registration link: When a student wants to register at the University and obtain a valid student credential, she navigates to the Patras portal and selects the menu ‘Get University ABC Credential’. Then the Patras Portal will redirect her to the course University Registration system.
2. Booking Course link: When a student wants to book a course and obtain a valid course credential, she has to browse the Patras Portal and select the menu ‘Obtain Course Credential’. Then the Patras Portal will redirect her to the University Registration system where she will get a valid course credential in her smart card.
3. Course Evaluation link: Whenever a student wants to evaluate a course, she has to select the menu item ‘take part in the course evaluation’. Then the Patras Portal will redirect her to the course evaluation system where the credentials on the smart card will be verified to ensure that only users satisfying the specified policies will be able to access.

##### 2.1.1.3.1 Application Architecture and Interfaces

Figure 4 depicts the Patras Portal architecture and its subcomponents. There is only one Interface, explained in the table below. The three http connections are merely Links that links the browser can follow.

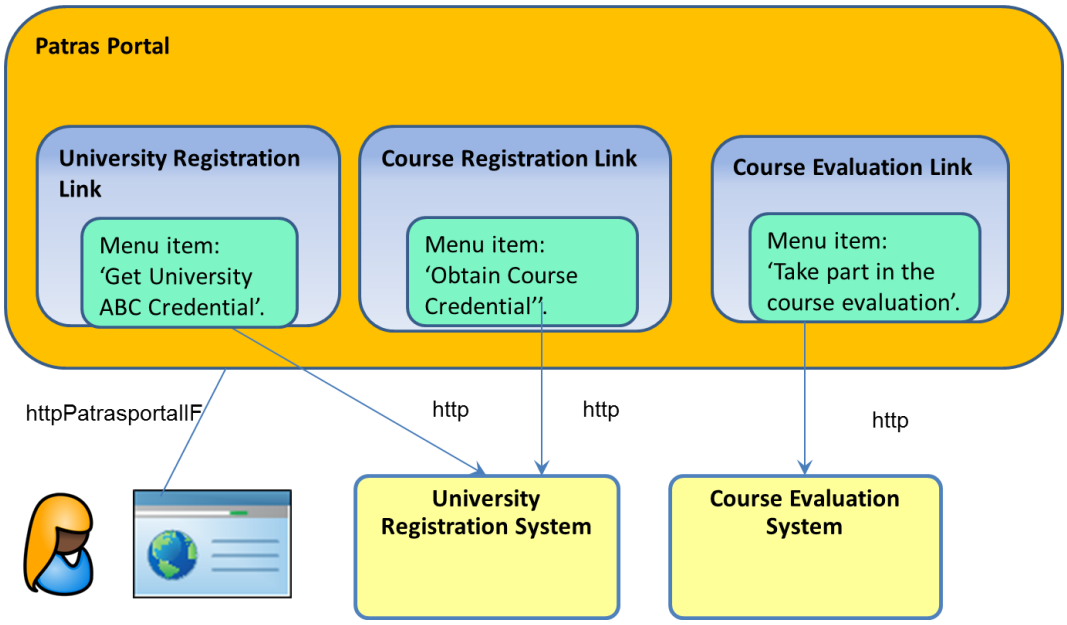


**Figure 4 Patras Portal**

**Table 2 Patras Portal Interfaces**

| Name | Applications that share this interface | Protocols used | Purpose |
|---|---|---|---|
| **httpPatrasportalIF** | **Patras Portal-User Browser** | **http** | **To provide instructions for pilot operations** |

### 2.1.1.4 Course Evaluation System

The main function of Course Evaluation System is to realize the anonymous course evaluation process. Potential users of this system are students, professors, Hellenic Quality Assurance Agency (HQAA) members and system administrators.

The Course Evaluation System can be accessed only by the users who own credentials that can satisfy certain policies. This system will only allow professors, certified students, HQAA employees and administrators to use it. It will support role-based access, and different actions will be allowed to each role. For instance, it will allow a professor to upload the questionnaires for his course, or it will allow certified students to fill in the available questionnaires. Each student is allowed to evaluate multiple times but only the last evaluation will be taken into account.

Course lecturers will access the Course Evaluation System in order to upload questionnaires regarding their course, and to setup the threshold number of attended lectures required for participating in the evaluation. Students will access the Course Evaluation System to evaluate the courses that they have

registered to and attended. When the evaluation procedure is completed, HQAA members will access the Course Evaluation System in order to view accumulated course evaluation results. HQAA have no role or participation at the evaluation procedure, they will only be able to browse the results (for more details please view D7.1).

The Course Evaluation System will manage:

- data related to course's information
- students' submitted data
- data for the evaluation of questionnaires and other evaluation data.

#### 2.1.1.4.1 Application Architecture and Interfaces

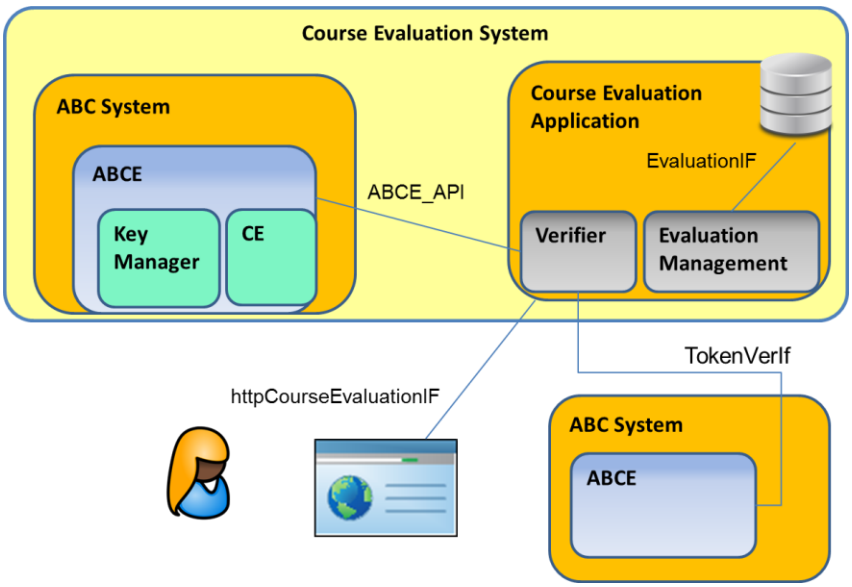


**Figure 5 Course Evaluation System**

The interfaces of Course Evaluation System and its application architecture are shown in Figure 5. The sub-components of the Course Evaluation System are a Privacy-ABC System and a Course Evaluation Application (implementing the business logic). Both components share an interface (ABCE_API).

While the Course Evaluation Application defines who can gain access using the presentation policy, the ABC System does the actual verification process by evaluating the presentation token and checking it against the policies. The detailed description of Privacy-ABC System functionalities and its interfaces has been described in Deliverable 2.1.

The Course Evaluation Application is the main component and implements the actual business logic. It has three functions: The first one (1) is to provide verification of tokens that fulfil the following properties:

a. The owner is registered with the university and enrolled in the course

b. The owner has attended a sufficient number of lectures

The other (2) is to store the course evaluation submitted data of the verified students that have the above properties. Finally (3) the Course Evaluation Application can provide accumulated results and various representations of them to HQAA members.

The interface to fulfill purpose (1), verifying tokens requires two separate interfaces one for receiving the (authentication) token and another for providing the result of the verification. These operations are provided by the verifier component of Course Evaluation Application, which makes calls to ABC Engine. The interface to full-fill purposes (2) and (3) can be a custom protocol that is executed by evaluation management interface.

**Table 3 Course Evaluation System Interfaces**

| Name | Applications that share this interface | Protocols used | Purpose |
|---|---|---|---|
| **httpCourseEvaluationIF** | **Course Evaluation Application-Web browser** | **http** | **To access and administrate the Curse Evaluation System** |
| **TokenVerIF** | **Course Evaluation Application – User Client** | | **User client provide the token to be verified** |
| **EvaluationIF** | **Course Evaluation Application- Data base** | | **To submit or retrieve or process course evaluation data** |
| **ABCE_API** | **ABC System-Verifier of Course Evaluation Application** | **API calls** | **To verify tokens** |

#### 2.1.1.5 User Client

The User Client refers to the hardware as well as the software that the Patras pilot users should have in order to be able to interact with the pilot platform.

First of all, each user should possess a hardware token (e.g. smart-card), where his credentials and private information (e.g. secret key) will be stored securely. Moreover, each user should have a smart card reader, enabling communication with his smart card, attached to his PC.

When a User wants to interact with the online systems, e.g. University Registration System and Course Evaluation System, he must use the browser with the ABC4Trust plugin installed. To enable the communication with the Privacy-ABC subsystems of the University Registration System and the Course Evaluation System, the software extension (browser plugin) utilizes a separate TCP connection. Apart from the basic token presentation functionality, the plugin provides a GUI that enables the User to browse his credentials, decide which credentials/pseudonyms to use when interacting with online systems. Finally, in order to be able to perform all ABC operations (e.g. create

an issuance token, create a presentation token) the User must have installed a Privacy-ABC System on her PC.

#### 2.1.1.5.1 Application Architecture and Interfaces

Figure 6 presents the application architecture and interfaces of the User Client. The functional components of the Privacy-ABC System have been described in Deliverable 2.1. The User-Agent application is a part of the software package that is installed on the User's browser. Via the graphical user interface that this User-Agent offers, some actions on the credentials stored on the card (e.g. browse credential, delete credential) can be carried out. Furthermore, it helps ABC Engine to contact the user for some operations such as credential selection or smartcard unlocking. As it is shown in figure 6, a smart card reader is used in order to enable the ABC Engine to communicate with the smartcard. Finally, the smart card is the hardware token that securely stores the User's personal information e.g. credentials and cryptographic keys.

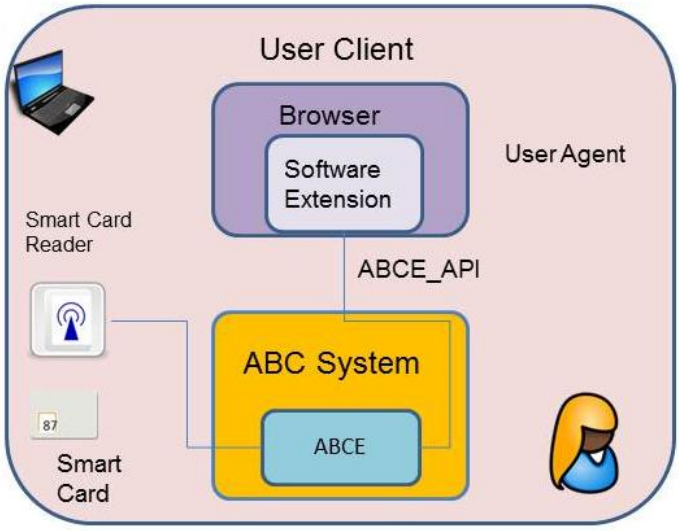


**Figure 6 User Client**

**Table 4 Patras User Client Interfaces**

| Name | Applications that share this interface | Protocols used | Purpose |
|---|---|---|---|
| **ABCE_API** | **Browser Application – ABC System** | **API calls** | **To facilitate ABC issuance and verification processes** |
| **User Agent** | **Browser Application – ABC System** | | **To provide a GUI to User for his credentials** |
| **Smart Card Reader** | **Browser Application – User Smart Card Application** | | **Enables communication with the smart card** |

### 2.1.2 Properties of the Pilot

This Pilot is a standard use case. The Course evaluation system has a high need for authentication, to stop unauthorized persons from influencing the evaluation (be that they haven't attended the lecture often enough for a proper opinion or are no student altogether). In this scenario it is also important to stop the lecturer pressing the student into positive responses. This scenario occurs frequently, therefore it allows to have two rounds of this pilot - insights from the first run will improve the Privacy-ABC technology and can be tested in the second run.

There are two distinct privacy features in this pilot:

- Collection of attendance credentials and conversion of them into a token, that allows participation in the evaluation
- The evaluation process itself

Additionally this pilot is prepared for standard processes, such as a broken & lost smart card, or revocation of credentials. With respect to revocation we choose to deploy a credential hierarchy, which facilitates de-central issuing and central revocation (e.g. if the student leaves university). This is implemented by the use of two credentials (University Credential + Course Credential) along with a verification policy of requiring both credentials.

## 2.2 Söderhamn Pilot (Community Interaction among Pupils)

Swedish schools are obliged, according to laws and regulations, to inform the guardians when a pupil is absent from a class. In addition, schools are obliged to create individual plans for each student. However, individual plans contain private data and very sensitive information about a child's ability to read, ability to write and other important skills, wishes and goals for the future. Today, Swedish schools use mainly the Internet as the means of communication. Thus, protection of privacy of pupils and their guardians is limited.. There could also be dangers in the possibility to identify all of a user's communications with the school, e.g. if the same username is used in sensitive and everyday use matters. The school pilot will use Privacy-ABC to enable secure identification in communications between staff, pupils and guardians.

The pilot application at a Swedish school, based on Privacy-ABC, will involve pseudonymous community access and social networking for pupils. This pilot addresses the specific challenges posed by the fact that Internet users get ever younger and often are minors. Nowadays, Swedish schools are mainly using the Internet for communication between teachers, pupils and parents. They are using different portals, operated by private companies with business interests that usually conflict privacy protection goals, for example profiling of pupils for higher advertisement returns. Sometimes private communities are setup to make communication possible. A big threat to privacy of the pupils is unauthorized access to personal information such as individual plans, presence reports, grades, exam results and other information and functions available in the school portal. Several applications, such as social networking or anonymous student counselling or medical advice will benefit from the ABC4Trust-project as it allows combining strong authentication and privacy protection into one solution. The proposed community will protect the pupil's identity against theft while protecting their anonymity and privacy. On one hand, pupils will be able to authenticate to access restricted chat rooms and restricted information. On the other hand they will be able to remain anonymous when asking private and sensitive questions from school personnel, while assuring the school personnel that they communicate with authorized pupils of the respective school, gender, age or class. The pilot will

help to gather information on the usability of the proposed system under specific challenging usability conditions imposed by category of the users.

### 2.2.1 Söderhamn Pilot Components

The Söderhamn pilot consists of four different systems: the Söderhamn Portal, the School Registration System, the Restricted Area System and the User Client. The Söderhamn Portal connects to the Restricted Area System. The School Registration System and the Restricted Area System are server side systems, both connecting to ABC Engine component. The User Client is a client side system, in the form of a web browser plug-in that also connects to a local ABC Engine component.

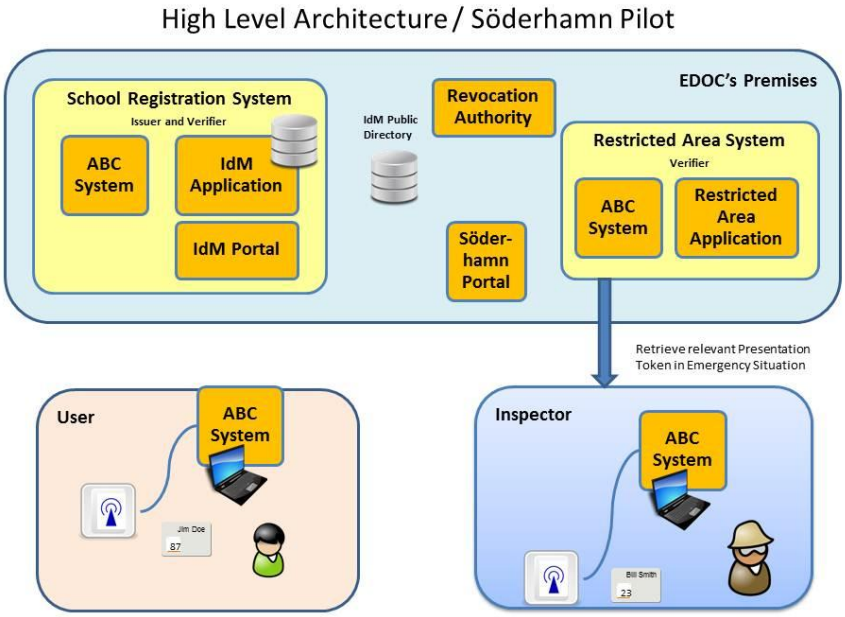


**Figure 7: High Level Architecture of Söderhamn Pilot**

The architecture of the application includes various interfaces to assure the integrity of the system. The architecture is modular and the entire system consists of servers, which are implemented using different technologies and programming languages. Most of the servers in the architecture can be delivered as virtual machines.

Interfaces are connecting different servers for requesting communication. The Restricted Areas application needs primarily an interface to request the ABC System to check access rights for a user according to her credentials. The rest of the interfaces serve to give the system the ability to perform this check and also functions as issuing credentials, revocation etc.

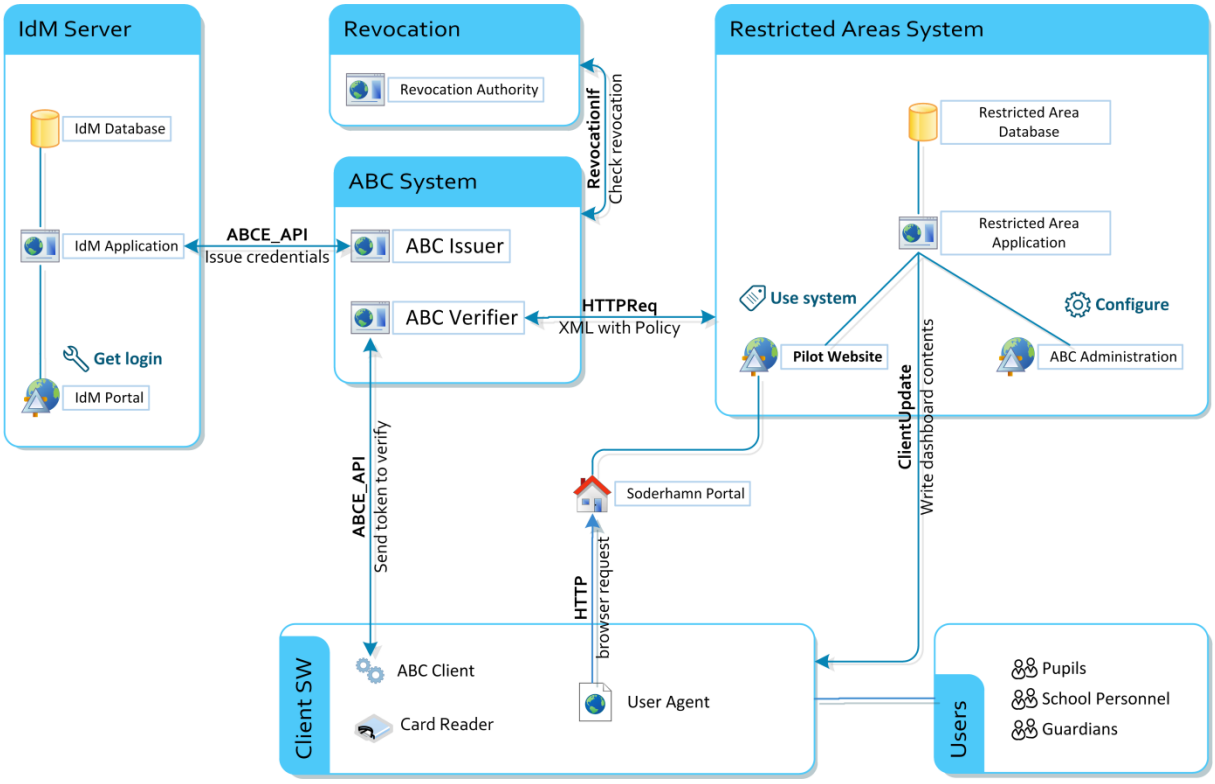


**Figure 8. School Pilot Interfaces**

### 2.2.1.1 School Registration System

The School Registration System is the component that binds the real world identities (name, civil ID number, age etc) to the corresponding credentials. This system needs to know the real world identity and allows management of attributes, for example registering that a pupil swapped classes, left the school or that new subgroups are formed.

In this pilot the School Registration System is used to bootstrap and administrate the pilot system. During the daily operation the School Registration System will not be involved in transactions. In other use cases (e.g. with often changing external attributes) it is conceivable that the user has to regularly log into the School Registration System and obtain new credentials. Within the School Registration System, the IDM system will be the connection to obtain the identities as credentials and include the issuer functionality. The verifier functionality is for convenience to allow students logging in via smart cards instead of remembering another password.

#### 2.2.1.1.1 Application Architecture and Interfaces

The internal architecture of the School Registration System is depicted in Figure 9: Architecture of the School Registration System

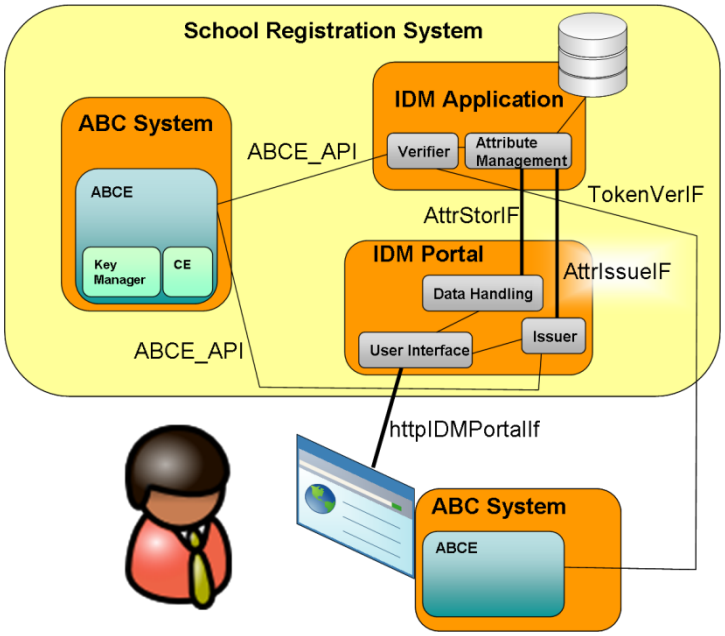


**Figure 9: Architecture of the School Registration System**

The functional components of IDM application and privacy-ABC Systems are clearly separated. The interface between both components has been described in Deliverable 2.1.

The IDM Portal is the self-administration portal of the school registration system. It is also the place where the user gets new credentials issued. Authentication toward the portal can be done by existing username/password, (national) ID card, One-Time-Password or other means. Later authentication with ABC4Trust credentials is also possible. However, as the IDM Portal supports use cases such as "Broken Smart Card", there needs to be other authentication methods than ABC4Trust credentials. The IDM portal is the frontend, which does not store any data, but relies on the IDM application to provide attributes and verify attributes.

The IDM Application has two purposes (1) to store and provide attributes, that origin from non Privacy-ABC systems, such as the pupils' names or classes. The other (2) is to provide verification of tokens or attributes, like authentication in a standard IDM Ecosystem.

The interface to fulfil purpose (1) between the ABC Engine and the IDM may utilize a custom protocol. This protocol is School Registration system internal. For larger or multi-vendor installations standard protocols are needed, good candidates are SAML, OAuth or in certain cases also OpenID.

Purpose (2), verifying attributes and tokens requires two separate interfaces one for receiving the ABC4Trust token and the other for providing the result of the evaluation to an external service depending on it. In this pilot the IDM portal is the only recipient of the verification. In other scenarios external recipients are conceivable in which case the IDM application could act as a Relying Party Token Proxy Service for these external recipients.

**Table 5 IDM School Registration System Interfaces**

| Name | Applications that share this interface | Protocols used | Purpose |
|---|---|---|---|
| **AttrIssueIf** | **IDM Application – IDM Portal** | | **Storage of attributes, e.g. changed home address.** |
| **AttrStoreIf** | **IDM Application – IDM Portal** | | **Retrieval of Value for issueing a credential** |
| **TokenVerIf** | **IDM Application – User Client** | | **User client providing the token to be verified** |
| **httpIDMPortalIf** | **IDM Portal – Web Browser** | **http** | **To view and administrate the user's data using standard web browser.** |
| **ABCE_API** | **ABC System – Verifier of IDM Application** | **API calls** | **To verify tokens** |
| **ABCE_API** | **ABC System – Issuer of IDM Portal** | **API calls** | **To issue tokens** |

#### 2.2.1.2 Söderhamn Portal

The Söderhamn portal is the webpage that holds links to different websites used in this pilot, for example to the IDM where a user can download his credentials to his smart card, to revocation authority, and to the restricted area section.

##### 2.2.1.2.1 Application Architecture and Interfaces

The portal consists of a standard web server and a small number of web pages. Its only interface is the http server for the user client's web browser.

**Table 6 Söderhamn Portal Interfaces**

| ID in Diagram | Name | Applications that share this interface | Protocols used | Purpose |
|---|---|---|---|---|
| **RevocationIf** | **Revocation Status Request** | **School Portal – Revocation Application** | **https** | **Request status of user to check if it is revoked** |
| **httpRestrictedAreaIf** | **Restricted Area Interface** | **School Portal – Restricted Area system** | **https** | **Communicate requests from Portal to RA System** |
| **ABCE_API** | **Check Smart Card** | **ABC Client – Website** | **https** | **Communicate with Card Reader and ABC System** |

| | | | | |
|---|---|---|---|---|
| **httpRAClient** | **Request RA** | **RA Client - Website** | **https** | **Request information from Restricted Areas Application** |
| **User Agent** | **httpSchoolPortalIf** | | **http/https** | **Send requests to web application from browser** |

#### 2.2.1.3 Restricted Area System

The restricted area system incorporates the business logic of the pilot. It implements the file upload, message forum and chat functionality. The restricted area system can store messages, files and maintain a list of aliases used, so that during an anonymous chat users can easily remember chat partners' names (or nicknames).

Restricted Areas web server is an interface of the whole Restricted Areas application. Web server is an interface to show the user pages of the RA website. User can get to a Restricted Areas' website or IdM Portal using the links on Söderhamn Portal.

By using Restricted Areas' website users can make benefit of RA application functionalities and access administration console implemented in the business logic.

Interactions between Restricted Areas application with ABC System and Revocation Authority reside on business logic layer using interfaces described more detailed in the next subsection and **Error! Reference source not found.**Table 7. Results of actions are shown to the user via HTML pages delivered through the Pilot website.

A restricted area can have one or several functionalities such as chat, forum (wall), documents (file upload) activated. And any content such as text messages and files in any format (ASCII, PDF, MS-word etc.) can be transferred. For each type of communication (use case) and functionality the owner/creator/administrator can change the setup/configuration of the restricted area. Each restricted area is protected by access policies that define, who should be able to access the restricted area. Users that meet the requirements of the restricted area policies can enter the restricted area and use its functionalities and access its content.

##### 2.2.1.3.1 Application Architecture and Interfaces

The restricted area system consists of one component including data storage. The restricted area web application is split in two different sections and consists of several web pages. One section for administration of restricted areas and one front-end section for using restricted areas.

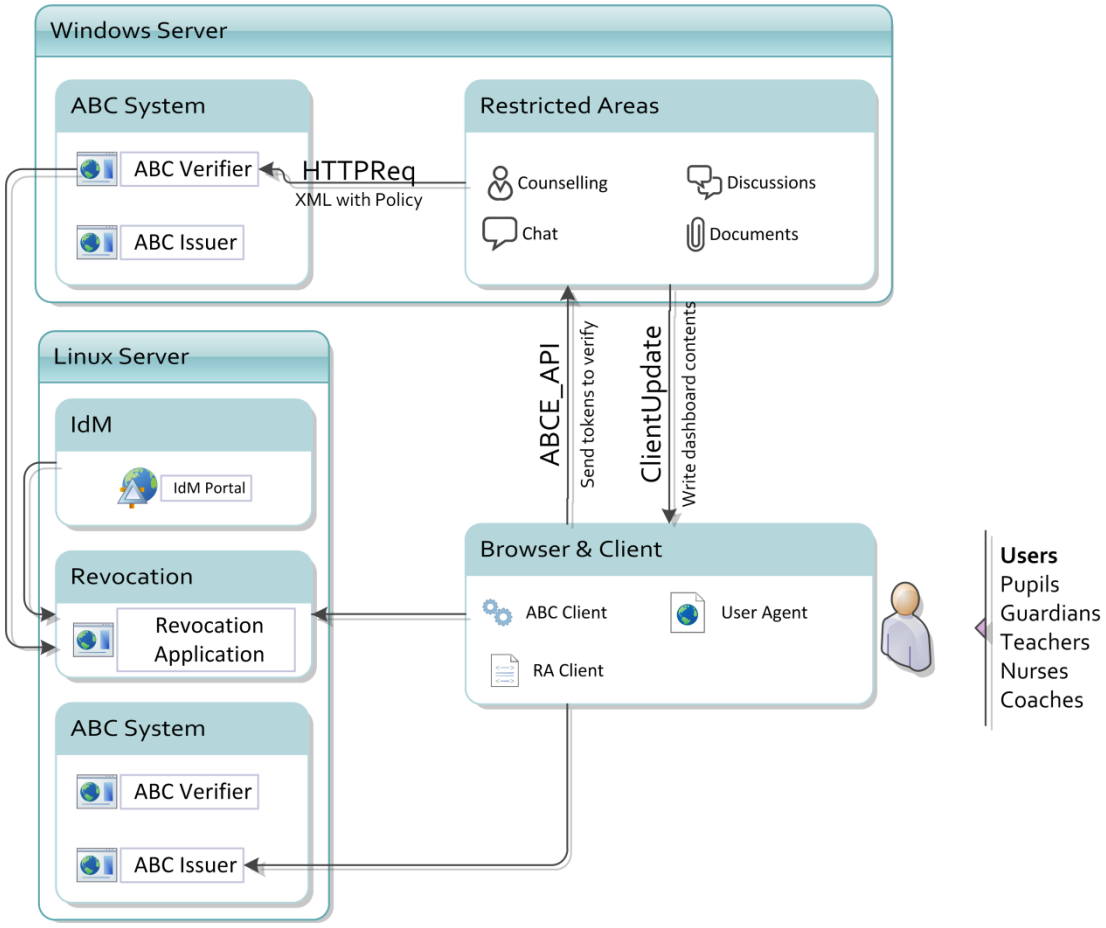


**Figure 10. Interfaces to Restricted Areas**

The administration section has a separate web page for

- Listing all restricted areas
- Administration of predefined access policies
- Configuration of restricted area
    - Functionality (chat, wall, documents etc.)
    - Access policy

The front end section has separate web pages for

- Listing all restricted areas available
- Access policy validation (comparing the users credentials with the access policies)
- Consuming the restricted area functionality
    - Chat
    - Wall

- Document upload
- Counseling
- Political discussions

**Table 7 Söderhamn Restricted Area Interfaces**

| ID in Diagram | Name | Applications that share this interface | Protocols used | Purpose |
|---|---|---|---|---|
| **HTTPReq** | verifyPolicy | **RA System – ABC verifier** | Web service (XML) | Check tokens to policies |
| **<none>** | Inspection | **RA System – ABC inspector** | manual | Inspection, admin gets the token from the database and manually sends it to the inspector |
| **ABCE_API** | **ABC API** | **ABC Client – ABCE** | **API** | Send credentials to verifier from client |
| **ClientUpdate** | **Update Client** | **RA System – ABC Client** | **Web service (XML)** | Send data from RA to client which updates user dashboard |

#### 2.2.1.4 Revocation Authority

Besides issuing credentials, a proper authentication system should also support revocation of issued credentials. In the architecture of ABC4Trust, the Revocation Authority (RevAuth) is the entity that is responsible for the revocation process (i.e., reacting to revocation requests, revoking credentials and disseminating the information about revoked credentials to stakeholders). The Role of Revocation Authority can be taken by the same entity that implements the Issuer or Verifier, or it can be implemented as a separate entity dedicated for revoking credentials.

In the Söderhamn pilot, the School Administration Office will perform the role of the Revocation Authority (as described in the deliverable"*D6.1-Application Description for the school deployment*"). The School administration can accept revocation requests from different parties (Revocation Requesters) or act as a Revocation Requester on its own, depending on the triggering reason.

Typical reasons for revoking credentials in the Söderhamn pilot are the following cases:

- A user left the school or repeatedly behaved disturbing;
- Some attribute attested in a credential is no longer valid, i.e. a pupil changes her class, opts out from a certain course;
- There is no legal basis for a user to continue using the system, i.e. a user (i.e. a pupil or her parent/caretaker) revokes her consent and wants to quit using the system;
- If the user loses the right to possess such a credential;
- A user loses her smart card, the smart card is damaged, or the secret key is compromised.

School Administration Office will take the necessary measures to react timely and duly to revocation requests, and perform a timely update of the latest revocation information. Furthermore, this authority will implement revocation processes such as to comply with the Swedish legislation with regard to the Data Protection Act. In this regard, proper measures will specify a certain period for data retention and

deletion after each credential revocation. Also, for audit needs, adequate measures will be implemented to assure a proper archiving process of the necessary data.

**Table 8 Söderhamn Revocation Interfaces**

| ID in Diagram | Name | Applications that share this interface | Protocols used | Purpose |
|---|---|---|---|---|
| **RevocationIf** | RevocationCheck | **Revocation Application – RevAuth System** | https | check revocation |

#### 2.2.1.5 Inspector

Anonymity in online communication may provide value for the users, but absolute anonymity may lead to service abuses such as spam, harassment, or fraud. Anonymity may increase users' trust in the technology, possibly increasing the adoption of the technology within the school.

However, there may be cases when unconditional anonymity may not be desired. Only in such cases and in a highly-controlled manner has the pilot foreseen lifting of certain user's anonymity, i.e. in case of an emergency situation, or when the health or life of a pupil is threatened. In ABC4Trust, the process of lifting the anonymity of a user is known as *inspection* and is performed by an *Inspector*, which is a separate (trusted) entity, different from the Verifier. The Inspector must on one hand be trusted by the User not to uncover identities unnecessarily, while at the same time be trusted by the Verifier to assist in the recovery when an abuse does occur. However, in order to be transparent to the users, not all credential attributes included in a presentation token can be inspected. Whether an attribute can be inspected will be notified to the user prior to its release.

In the Söderhamn pilot, the role of an ABC4Trust Inspector will be performed by the School Inspection Board (see D6.1), who will be the entity that receives inspection requests, analyses the inspection grounds and take appropriate actions. In case the inspection request is legitimate, this entity will then able to perform the required inspection. There is a discussion within the project to make the role of the Inspector shared between a certain number of persons, so that at least two out of a certain number of appointed Board members are required to be present during the inspection process. This is mainly thought of as a mean to avoid potential misuse of this feature and to add a higher level of trust and transparency in the process.

Moreover, the possibility for inspection will be communicated to the users in the Terms of Use of the pilot services. In case of an inspection actually taking place, the School Inspection Board (the Inspector) will take appropriate steps to inform the affected (inspected) user(s) as soon as circumstances permit it. For auditing purposes, a (highly-secure) logging and documentation of the inspection transcript will always be performed.

#### 2.2.1.6 User Client

The user client is responsible for saving and retrieving sensitive information about the user's activities to and from the user's smart card. An example of sensitive information is the list of visited restricted areas or the list of pseudonyms used. In order to protect such sensitive information from the system (the web application itself), the history of previously visited restricted areas and associated alias will be saved on the users' smart cards.

When a user wants to revisit a restricted area and uses the same alias, the user client, which runs locally on the user's computer, will show the list of restricted areas and does all the operations that

should not be revealed to the server. The logic to implement this feature needs to reside on the client to be privacy preserving.

#### 2.2.1.6.1 Application Architecture and Interfaces

- The client, based on the standard ABC client, is extended with additional functionality required by the Restricted Areas System. Extension of the ABC client and server-side business logic functions implements the following features
  - Handle the restricted area history
  - Handle the dashboard
  - Store pseudonyms used in restricted areas

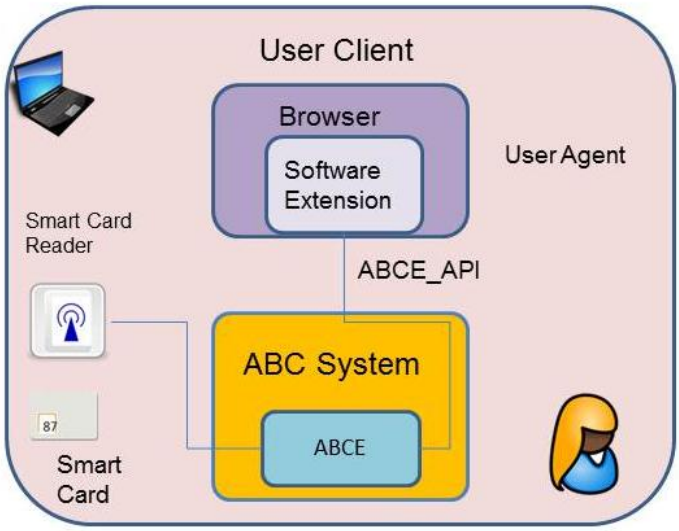


**Figure 11. User Client Interfaces**

**Table 9 Söderhamn User Client Interfaces**

| ID in Diagram | Name | Applications that share this interface | Protocols used | Purpose |
|---|---|---|---|---|
| **ABCE_API** | **Check Smart Card** | **ABC Client – ABC System** | **https** | **Communicate with Card Reader and ABC System** |
| **http** | **Request RA** | **RA Client – Website** | **https** | **Request information from Restricted Areas Application** |
| **ClientUpdate** | **ClientUpdate** | **RA System – ABC Client** | **http/https** | **Send requests to web application from browser** |

### 2.2.2 Properties of the Pilot

This Pilot deploys additional advanced Privacy-ABC technology features. It not only utilises verification, but also deploys revocation and inspection. Additionally the use of scope exclusive pseudonyms[2] is introduced. We envision social networks as the online service that requires most privacy technology, beside pseudonymisation of profiles (see also results of PRIMLIFE [LEH2010] or the DIASPORA project [DIASPORA]), chat forums with user selected nick names are important – pseudonyms bind these (human readable) names to one user. Hereby the Privacy-ABCs technology provides tools to ensure a pseudonym cannot be taken from another user, without "possessing it". Additionally to ensure that each user cannot make up several pseudonyms, scope exclusive pseudonyms are used.

Another interesting aspect of this pilot is that it features a large user group, consisting of teenagers, parents and officials. Achieving acceptance of the restricted area system should be counted as a big success.

[2] A scope exclusive pseudonym ensures that only one user can hold a specific handle within a restricted area and that a user cannot hold more than one handle.

# 3 Common Denominator Elements

Common to both pilots' scenarios and also to all envisionable scenarios are the following common denominators. Additional generic elements exist (see Section 4), but they are not mandatory for deploying the ABC4Trust technology.

## 3.1 Issuer

In the state-of-the-art IDM Ecosystem, the IDM is the authorative[3] entity. The IDM knows the user profiles and can do assertions about the attributes. The Issuer has the same role in the Privacy-ABC technology, with the difference, that after handing out the credentials the Issuer is not involved in the actual proof process. The issuer will not know how often and to whom the user was revealing the attributes that the Issuer created credentials for.

### 3.1.1 Description of the function of an Issuer

The Issuer provides knowledge about the user (e.g. whether his is in a certain class, his birthday or gender) in form of credentials. Using these credentials users can prove possession of attributes. It is conceivable that different Issuers for different domains exist, for example the car registration issues credentials about 'his car is fit for the road', while the insurance company issues credentials for 'being covered in case of an accident'. Similarly in the Patras pilot the university registration system issues credential about being a student and subscribed to a certain course, while another entity issues the statement about each student having attended a certain lecture.

The function of the second Issuer is different as it does not have a database (i.e. attributes are not derived from stored data), but to act like a notary stating the observation, for example that a student attended a lecture. The missing database (or that there is no non-private record of issuing) creates a problem when the credentials are lost (e.g. a smartcard breaks). *At this point in time different mechanisms are being discussed. One would require the support of the IDM role, see section 3.5. More information will be provided later.* This type of Issuer would have a business model similar to those of the Privacy CA within trusted computing [C04]

State-of-the-Art IDM system like Microsoft's InfoCard system [INFOCARDS] know another type of attributes, namely self-certified attributes. In Privacy-ABC systems all credentials are required to origin from an Issuer[4].

Most commonly the Issuer will be connected with an IDM system that knows the user and issues credentials about the user. Different to a state-of-the Art IDM these Privacy-ABC credentials allow the user to have control over the information sharing by knowing when attribute gets disclosed and also choose to partially disclose an attribute, for example "pupil of 7$^{th}$ grade" instead of "pupil of class 7b". Additionally the credentials from the Issuer are untraceable, so even if the issuer colludes with the verifier (or several) it is not possible to tell which "specific pupil of the class 4b" just visited the verifier.

[3] In the OpenID standard based system, the entity knowing the user might have less authority than in traditional IDM systems, still this authority replies on behalf of the user.

[4] It is possible that the user runs an Issuer himself, without involving other legal entities for their self-certified properties.

Finally all Issuers have to participate in the revocation process, by keeping track of the revocation handles of issued credentials. If a credential needs to be revoked the revocation handles has to be transferred to the Revocation Authority.

### 3.1.2 When to take the role of an Issuer

Ecosystem participants that are holders of a user database, that shall be made available in a privacy-preserving manner, can take in the full advantages of the ABC-technologies' Issuer role.

However, it is not always necessary to issue credentials based on the stored attributes, in some business scenarios. Being a service provider with a non-user-profile based business logic and being an Issuer at the same time makes sense. In the Söderhamn pilot, there have been discussions whether the properties of pupils could be created on-the-fly. For example a teacher could assign the anonymous participants additional attributes such as "top of class". Later a new chat room could be created that require having the attribute "top of class". Another example is the class attendance credentials in the Patras pilot. To implement this feature the restricted area server would need to include an Issuer, which provides additional credentials to users, but it does not require a user database where the issued values are derived from.

### 3.1.3 Operation of an Issuer

The Issuer requires two actions in the setup phase,

- Connecting to the database holding the attributes (e.g. the IDM)
- Setting up the cryptographic material for issuing the ABC Tokens

Administrative Intervention during the issuing process is not necessary. The User will use self administration to issue credentials for attributes available in the database. Change of attributes is done by administrating the IDM database.

However for revocation, administrative effort is needed, as the handles of the revoked credentials need to be transferred to the revocation authority.

## 3.2 Verifier

The verifier is the entity that verifies the information that the user puts in the presentation token against what Issuer had put into the credentials. That information can be in full (i.e. exact birth date), but most often partially revealing the information is sufficient (e.g. above 18). Even though the business model of Verifiers usually does not need Privacy-ABC technology for differentiation reasons but to provide privacy friendly service it can be beneficial. There can be several Privacy-ABC technology systems that a verifier can implement. However, one the purposes of ABC4Trust is to bring together the Privacy-ABC technology of IBM and Microsoft, and to establish a framework, that also other and future Privacy-ABC technologies can utilise, thus reducing the overhead of Verifiers to implement several APIs.

### 3.2.1 Description of the function of a Verifier

The basic form of a Verifier consumes the ABC tokens, which are derived from ABC credentials, to check if a user satisfies a required policy, mostly for access, but possible also for customization

purposes. Additionally legal requirements may force a service provider to implement the Verifier functionality to ensure only a certain group can enter the service.

An advanced form of Verifier is similar to the RP-STS (Relying Party – Secure Token Service) known from the Web Service technology. In that case the service provider does not need to implement the Verifier functionality themselves, but rely on an intermediate party to do so. The intermediator implements a Verifier functionality, evaluates the policy needed by the service provider and hands back the result in a proprietary or standardized protocol (e.g. XACML Request/Response [MOASIS05]).

### 3.2.2 Operation of a Verifier

The Verifier needs to have one or several verification policies. A verification policy determines possession of which attributes needs to be shown to gain access, e.g. to that service provider. Included in the verification policy is whether certain attributes need to be proven in an inspectable way.

If sub-areas of the application require a more specific access policy, then additional verification policies need to be defined.

Additional to the basic access, a Verifier may need to handle pseudonyms, including their respective scope, created by the user. With a pseudonym a user can be recognized as "being the same as previous time", without being able to be recognized by her real identity. For example in the School Pilot (Söderhamn) there will be long time chats (forums) that the same user can post in several times. It must be prevented that users pretend to be a different user (i.e. 'take over a pseudonym'). The ABC4Trust technology provides the feature of pseudonyms and enforcing them is part of the Verification policy. The verifier is not responsible for solving the problem of the client to know which of his (many) pseudonyms needs to be proven for gaining access to a service. If the Verifier would be able to support the user, there would be a privacy issue.

It also be beneficial to ensure that each user can only have one vote such as during the course evaluation of the Patras pilot. For this purpose, scope exclusive pseudonyms are used. The verifier will provide the scope and ensures that each user within this scope can possess only one pseudonym.

Besides setting up the presentation policies, the verifier requires no additional administration, as the revocation authority does the revocation and users presents revocation proof themselves to the verifier as just another attribute.

## 3.3 Revocation process and the Revocation Authority

When a credential is issued, it is expected that it will be valid for its entire validity period. There can, however, be circumstances when an issued credential must be invalidated promptly after a short notice and before the initially assumed usage period. As such, revocation of credentials is considered to be a very important feature of any authentication system. Furthermore, there are even legal requirements in the European legislation with regard to revocation [EPC99].

Among the general reasons for revoking issued credentials, ISO/IEC 29115 [J1S27W5] identifies the following:

- A credential or the means to produce a credential has been reported lost, stolen or otherwise compromised;
- The basis for a credential no longer exists (e.g. some attribute values are not valid anymore);
- A credential has been used for unauthorized purposes; or
- A new credential is issued to replace the credential in question.

In the ABC4Trust architecture, a dedicated role (the Revocation Authority) has been identified to perform the process of revoking Privacy-ABCs. This entity exists both in the Söderhamn and Patras pilots and is therefore considered to be a common denominator for the two, even though two different actors are performing the respective roles for this entity: in Söderhamn, it is the School Administration Office that performs the role of the Revocation Authority, while in Patras this role is performed by the University Registration Office.

### 3.3.1 Description

A credential is a container of certified values of attributes about an identity plus some credential-specific attributes, signed by the Issuer of the credential(s). The Revocation Authority publishes its *revocation parameters*, which contain information about where the Verifiers can check about the latest *revocation information* and what mechanism to use for this. Revocation information is a set of certified data about the revoked credentials published by the Revocation Authority, which Verifiers use to check that a certain presentation token presented by a User is not produced by a revoked credential or a combination of them. Depending on the mechanism used, the identifiers of the revoked credentials may or may not be visible from the revocation information. On the other hand, Users also maintain some information about the validity of their credentials, known as *non-revocation evidence*, which they must update for every credential they possess and against every Revocation Authority listed for that credential.

The ABC4Trust architecture makes a distinction between two types of revocation: *Issuer-driven* and *Verifier-driven* revocation, depending on the initiator of the revocation and its scope. However, in the two pilots, only Issuer-driven revocation is implemented. An Issuer-driven revocation is global in its scope, as a credential revoked in this manner can no longer be used with any Verifier. Issuer-driven revocation is performed through a *revocation handle*, which is a dedicated unique identifier, inserted as a credential attribute during the Issuance of the credential (which is by default not revealed in a presentation token) (D2.1).

### 3.3.2 Mechanisms for revoking anonymous credentials

In classical systems with public-key cryptography (such as X.509 certificates), revocation is usually performed by either the Issuer of the certificate or some dedicated Revocation Authority, which publishes the serial numbers of the revoked credentials in a signed data format known as a Certificate Revocation List (CRL). Verifiers then (periodically) check if a given serial number (which is an attribute in a certificate) is part of this list to confirm if the certificate has been revoked or not. However, this scheme would not fit in the ABC4Trust architecture, as this would require the User to reveal the certificate serial number to the Verifiers, which would uniquely identify the user and this would therefore violate unlinkability property of p-ABCs. Therefore, alternative solutions have been proposed, which preserve the anonymity of the credentials during its presentation (D2.1).

Three different types of mechanisms can be used to implement revocation of p-ABC without violating the unlinkability property, namely *Credential Revocation Lists*, *short-lived credentials* and *accumulators.*

In a solution based on *Credential Revocation Lists* (CRL), the Revocation Authority keeps a list ("blacklist") of revoked credentials, having users *prove* that their credential is not contained in this list. This solution preserves the anonymity of the users, who can generate a proof that their credential is not contained in the CRL without revealing some unique credential fingerprint. Verifiers can also update the latest revocation information from the list before validating a presentation token received from the user. However, such a solution is not scalable, as the size of the CRL and the size of the public key of the Revocation Authority are linear to the number of revoked credentials.

*Short-lived credentials* is another alternative, in which neither the Users nor the Verifiers need to maintain a revocation list. Instead, the credentials are issued with a short life (having short expiration

period), after which users must request an update from the Issuer every time they want their credential's lifetime extended [CKS10].

This solution however has some disadvantages, as it requires the Issuer (or some other dedicated entity) to be online in order to perform the interactive credential update protocol. Furthermore, this solutions does not support immediate revocation, in case a credentials needs to be revoked before its expiry time [CKS10].

A final mechanism proposes a solution based on *accumulators*, where the Revocation Authority "accumulates" a list of credentials into a single value, known as an accumulator. A white-list accumulator accumulates the list of valid credentials, while the solution with a black-list accumulator accumulates the list of revoked credentials. On the other hand, a universal accumulator is both white-list and black-list. A so-called dynamic accumulator is a solution, where the accumulator value is fixed regardless on the number of the revoked credentials (with a defined maximum number of values to accumulate) [CL02]. Therefore, this solution overcomes the scalability issue in the solution based on CRLs. Users can (cryptographically) prove that their credential is (or is not) contained in the accumulator by performing a *witness update* protocol with the Revocation Authority. This solution is suitable on scenarios where immediate revocation is required, therefore providing an additional advantage to the solution based on short-lived credentials. However, this solutions makes the process of proving (generating a presentation token) computationally more expensive, as users need to *prove* (to the Verifier) not only ownership of a credential, but also its validity (witness update can be slow, in some cases up to 10 minutes) [LKDN10].

No single solution fits all scenarios and each solution has its own characteristics, advantages and disadvantages. Therefore, a compromise between performance, timeliness and scalability may be required. However, a combination of different mechanisms is also possible [CKS10]. For instance, a revocation mechanism based on short-lived credentials could be combined with the accumulator solution. In scenarios where performance is more important than the frequent revocation information update, a Verifier could impose presentation of a credential based on its validity period, while more sensitive services could decide to combine both in order to validate a presentation token from a user.

### 3.3.3 Operation of Revocation Authority

The revocation authority needs contractual and technical relationship with the Issuer, to know about invalid (and valid) credentials.

Further on it needs an interface where revoked credential handles are submitted, as well as an interface the user client can run a non-revocation proof protocol with.

## 3.4 User Client and User

The user client is the critical part of the Privacy-ABC Technology, oppose to a simple storage, it is required to do cryptographic computations, as the credentials of the issuer need to be transformed into tokens in order to be send to the verifier.

### 3.4.1 User Client Interface

The user client requires an interface to the ABC Engine and to the smart card, if the credentials are stored there.

The communication to the issuer and verifier will likely take place using a web interface, a standard HTTP or REST interface is deployed.

### 3.4.2 Non-mandatory Components

Although not strictly required for secure operation, there are additional components a user client should include, so to provide a convenient operation.

#### 3.4.2.1 Pseudonym Management

In applications where pseudonyms are used, the user client should include components for making the pseudonyms handling more convenient for the user. This component should store which pseudonyms the user has used, or should be able to identify from a list of all pseudonyms which ones the user can prove possession of.

#### 3.4.2.2 Other

Other suggested components will be added in the next version of this deliverable, if required.

#### 3.4.2.3 Operation of User Client

From the operational perspective, the user client does not differ from user clients of other credential-based systems like InfoCard or a certificate store. Several attributes (in form of credentials) are managed, and the user can select which ones to reveal to a service provider (Verifier), so to gain access.

Usability issues, such as how to keep an overview of data revealed have been covered in the EU projects [PRIMELIFE] and [uPRIME], hence they are only secondary focus of ABC4Trust.

## 3.5 IDM

The IDM role changes compared to the state-of-the art system. On one side the IDM is required to have the same database that stores sensitive data about users and poses a risk to privacy like in a state-of-the art system. On the other hand, it needs to include the roles of the Privacy-ABC system, namely Issuer and Verifier. The Issuer functionality is needed to provide the information of the database in ABC-Technology usable form, namely credentials. The Verifier functionality is to provide the users the possibility of logging into the IDM system using the Privacy-ABC technology. However there also needs to be a non-ABC mechanism to log into the administrative portal, to handle the case where the smartcard is broken or lost.

### 3.5.1 Description of the function of the IDM

The function of the IDM can be described as a gate between the privacy protected and the non-privacy protected world, namely the Privacy-ABC-technology world and the traditional IDM systems.

The difference between the IDM and Issuer is small, but crucial. The IDM knows the user with his non-privacy protected identity. Thus the IDM can bootstrap and assist recovery processes with that knowledge.

In most cases a company owning a profile database takes the role of an Issuer (only). Credentials from different Issuers can be collected in the same credential storage (i.e. smart card). If that storage gets damaged or lost, a backup copy could be used to restore the information. However to prevent transfer of credentials from one card to another (e.g. in the Patras pilot one student wants to share the attendance credentials with a class-mate) there has to be a process protecting the restore of an arbitrary backup. *The exact process is not yet finalized and more information will be provided later.*

That key role in facilitating restoring of credentials by personalising the user client makes the IDM (with connected Issuer) to the central component of Real-Life Privacy-ABC Installations. The IDM will likely hold a set of official attributes, such as the address, birth date and internal ID. Thus being also the gate between the privacy-protected and non-privacy protected world.

### 3.5.2 Operation of the IDM

During a setup phase, trust relations of the IDM component need to be setup, e.g. by means of exchanging certificates and agreeing on SAML protocol, or by enabling one of the authentication modules.

Then the user database has to be filled, including the information that will be used later to issue credentials. This process is called provisioning. Provisioning can be done manually, one-by-one, or as mass provisioning utilising a CSV[5]-File. Other possible methods of provisioning include pulling the data from another IDM system, or having the data pushed into the system.

The IDM requires processes that ensure security (confidentiality, integrity), available and recovery. Especially the use cases of recovery of lost/broken credentials need the support of the IDM. For this authentication of the user toward the IDM by means not relying on the Privacy-ABC technology is needed (e.g. One Time Passwords, or Administrators Authenticating the user in person).

The administration of the IDM can include an LDAP directory and the standard tools to administrate information in them.

Users shall also have means for administrating their profiles from a self-service portal. This information includes password and basic attributes such as for example the address.

[5] Character Separated Values

# 4 Optional generic Elements

There are additional ABC-Technology components that are generic, but not used in both of our pilots and most likely they will not be deployed in every scenario.

## 4.1 Inspector

*Disclaimer: Inspection is scheduled for completion during the summer of 2012. Thus additional information related to this section will be provided later.*

Anonymous services facilitate not only democracy, but also attract ill-minded individuals. For example in a school forum, someone might think it is funny to do a bomb threat, or pupil announces suicide. Legislation and community agreement has to identify the circumstances under which a lifting of the anonymity (of one specific user and token) is acceptable.

The ABC4Trust technology allows lifting anonymity by the role of Inspector. Not all tokens can be opened, but only those that were created with the option of inspection. The Verifier can choose to only allow tokens that enable inspection for his service.

### 4.1.1 Description

The Inspector is a role, similar to the User. It can also be the joint-action of several elected persons, so that not one person alone can lift the anonymity. For lifting the anonymity the Verifier provides a stored copy of the ABC token to the inspector. The Inspector then decrypts the inspection part to lift the anonymity of the token.

### 4.1.2 User Interface of the Inspector

*The user interface and operational aspects will be provided later.*

# 5 Hardware Requirements of the Pilots

This section introduces the hardware requirements of the two Pilots. They consist of three parts (a) the platforms that the pilots run on (b) the network layout showing how the different entities are protected, co-located or separated. The last part (c) is the hardware requirements of the two pilots, which takes small to medium setups into account. For large deployments additional scaling measures need to be introduced.

Users of the Privacy-ABC technologies that implement a closed system can benefit from the Söderhamn pilot, while users that need to integrate into wider IT infrastructure should review the Patras pilot, although no actual interconnection with the university administration system was done.

During the implementation of the 'On-Site Testing' in the environment of the Patras Pilot, it became clear which operating systems are needed to cope with the basic needs of the ABC4Trust technology. The main outcome of the analysis is that at least one .NET enabled operating system is required[6]. The following sub-chapters show that for the U-Prove crypto parts, .NET framework is required. For Idemix, off-the-shelf Linux-based operating systems like Ubuntu or SUSE can be regarded as being sufficient.

On top of operating system requirements for the servers, it is also necessary to have a closer look at the ABC4Trust User system. ABC4Trust Users need in addition to specific browsers also a locally installed web application to be able to participate in the trials.

*In case the hardware or network requirements need to be changed, a new version of this deliverable will include updates.*

## 5.1 Platforms of CTI, Eurodocs and NSN

### 5.1.1 Both Pilots

#### 5.1.1.1 NSN Issuer and Verifier:

Operating System for IDM Applications: openSUSE 11.3 / 11.4 (64 bit) as VirtualBox VMs.

Operating System for ABC Core components: Windows Server 2008 R2 as VirtualBox WM

### 5.1.2 Patras

#### 5.1.2.1 CTI Issuer:

**1st Round Patras:**
Operating System: Ubuntu 10.04 LTS (Lucid Lynx)

**2nd Round Patras:**
Operating System: Windows 7 or Ubuntu 10.04 LTS (Lucid Lynx) with Mono

#### 5.1.2.2 CTI User

Operating System: Windows (7, XP)
and
Linux (Ubuntu 8.04, 10.04, 11.10)

[6] It should be possible to utilize the MONO framework, so to run the ABC4Trust Engine on a Linux based platforms.

Browser: Firefox 9.0.1 or higher

#### 5.1.2.3 CTI Verifier:

Operating System: Ubuntu 10.04 LTS (Lucid Lynx) with Mono

### 5.1.3 Söderhamn

#### 5.1.3.1 Eurodocs User and Inspector:

Operating System: Windows 7 or Windows XP

Browser: Internet Explorer 8 (8.0.6001.18702) or later

#### 5.1.3.2 Eurodocs Verifier:

Operating System: Windows 7 or Windows XP

## 5.2 Söderhamn Pilot Network Overview

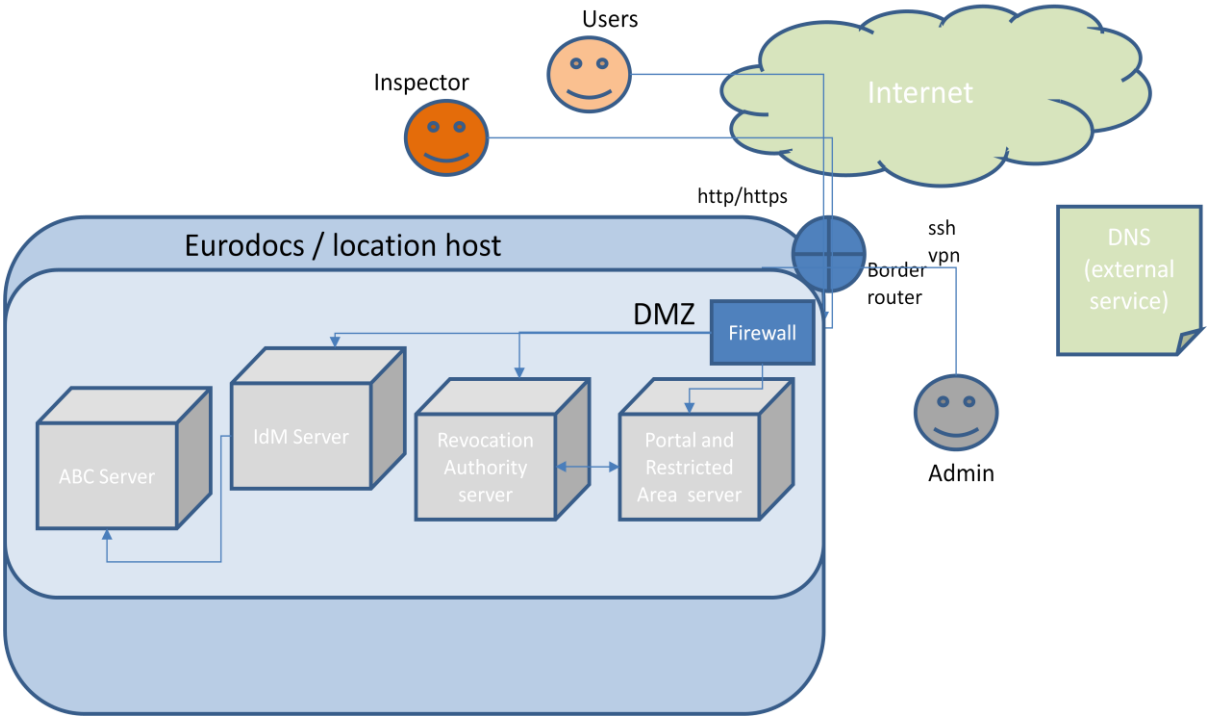


**Figure 12 Söderhamn Pilot Network Overview**

Figure 12 shows the Söderhamn Pilot Network Overview. The servers in the Söderhamn pilot will reside on a DMZ network, each with its own real IP address, accessible from the Internet. The DMZ

segment used by the school pilot and Söderhamn portal is logically separated from other traffic and set up at a location host who provides internet access but does not have access to the servers. The servers are managed exclusively by Eurodocs.

The DMZ segment itself is protected by a border router and a firewall, that let in http and https traffic, plus ssh / vpn traffic from server administrator's computers. All other traffic is refused at the border.

In addition to this border protection, all servers are protected by software firewalls, configured to only permit traffic on explicitly opened ports. They will be open for standard http and https port access via the external firewall, for database access to the restricted area database only from the restricted area- and revocation-servers, and for communication between the IDM server and the ABC server.

## 5.3 Patras Pilot Network Overview

Figure 13 presents Patras Pilot network infrastructure. More precisely, CTI has a modern corporate network located in three different sites: The main site is the building "D. Maritsa" Campus of the University of Patras and the other two smaller sites are at Department of Computer Engineer and Informatics and at the Athens CTI premises. The connection between the sites is realized by optical fiber between the building "D. Maritsa" and Patras University and by virtual private networking (VPN) with a capacity of 15Mbps between building "D. Maritsa" and Athens CTI premises. CTI is connected to the Internet through GRNET using a 1 Gbps speed connection by optical fibers in the building "D. Maritsa". CTI has its own public address space with 32766 available IP addresses and border gateway autonomous system. Moreover the University Registration System and Course Evaluation System will have their own, static, non-shared IP addresses which are reachable via the Internet. The active network ports in the building "D.Maritsa" are approximately 600. For the Patras University and Athens CTI premises sites the active ports are 50 and 170 respectively.

The network security is ensured by the existence of a pair of firewalls (Cisco Pix-535) connected for high availability configuration (active-standby, and without NAT). Firewalls are connected between the border router (Cisco 7300 series) and the CTI internal network, inspecting incoming and outgoing traffic, ensuring network security and protection against malicious attacks. Firewalls can block source IP addresses in the case of DoS attacks and traffic to non-authorized addresses in CTI’s internal network but cannot block packets with malicious content. We have also implemented a DMZ subnet. In the DMZ we have all the servers (e.g web and VPN servers) that offer public services and they don’t have any connection to the internal network of CTI. The VPN servers support the VPN connections that are necessary for the administration of Patras Pilot.

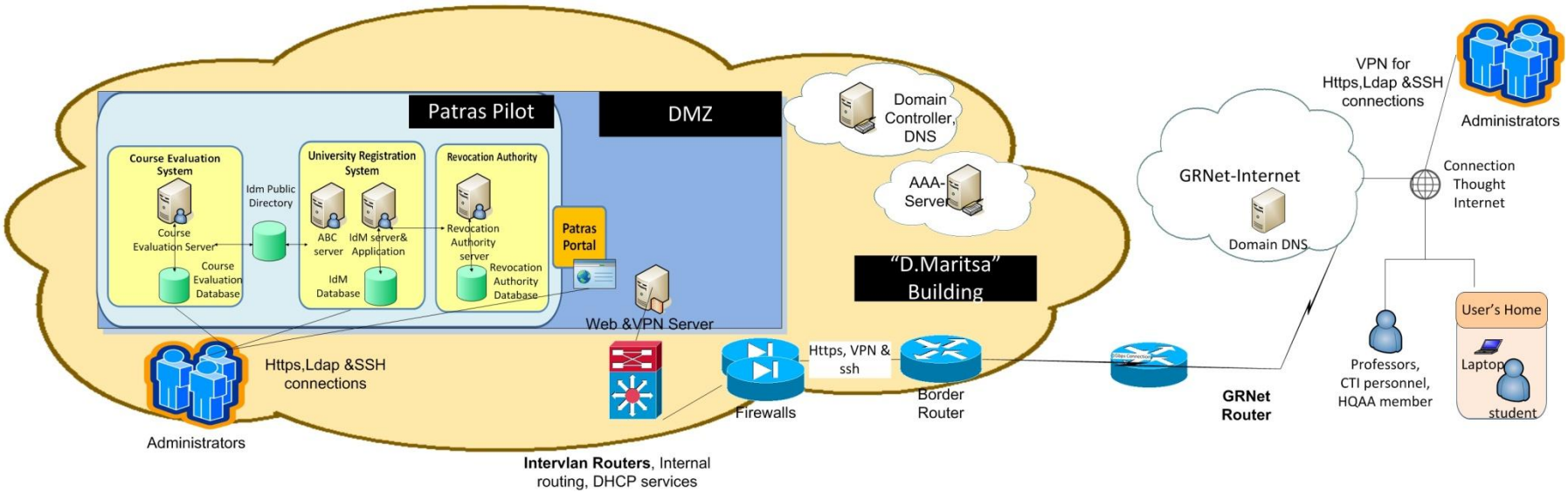


**Figure 13: Patras Pilot network overview**

The Patras Portal, the University Registration System, the Course Evaluation System and the Revocation Authority will be placed on DMZ and they will have their own IP addresses which can be reached from the Internet. The Course Evaluation System and Revocation Authority are equipped with their own servers and database repositories. The University Registration System is equipped with the Privacy-ABC system server, the IDM server and its repository. DMZ includes both Patras Portal and the IDM public directory. All incoming http/https requests go thought DMZ. In order to protect http/https incoming traffic we will implement access lists and rules according risk management requirements of Patra's Pilot. In order to have different roles of users, University Registration System and Course Evaluation System will be employed with different user accounts. The students will have local user accounts on the University Registration System. Administrators will use Active Directory in order to have access to the Course Evaluation System or the University Registration System via CTI's internal network or internet. Moreover lectors will have local user accounts on the Course Evaluation System for uploading their course questionnaire.

In the University Pilot there are several groups of users (for more details see section 1.4.1 of D7.1). Students, lecturers, CTI personnel and members of HQAA can access the University pilot via the Internet. The Course lecturers will make an http request to the Course Evaluation System in order upload the questionnaires. Students will establish an http connection (via User's Home) with the Course Evaluation System for evaluating courses, with the Patras Portal for getting instructions and with, University Registration System for registering. When the evaluation procedure is completed, CTI members and HQAA members will be able to make an http request to Course Evaluation System for processing or viewing the evaluation results. Professors, HQAA members and administrators will access the Course Evaluation System by using login name and password authentication. When a user make an http request via internet, his http connection will be routed from GRNet router to CTI's border router, and then after passing the firewalls and the internal routers the user will access the University Registration System or the Course evaluation Application in the DMZ by using the ABC technology.

Administrators of the Patras Pilot can access the DMZ via Https/http/ssh/ldap connections through CTI's internal network or the Internet via a VPN connection. Administrators will follow the above traffic path like users in order to establish a Https/http connection. When an administrator wants to establish ssh/ldap connection via te Internet, he must establish a VPN connection though a secure channel in order to get access the CTI's internal network and get a virtual IP at CTI's internal network. For more details for the authentication of VPN connections [TN1029] .

The Patras Portal, the University Registration System, the Course Evaluation System and the Revocation Authority will be placed on DMZ and they will have their own IP addresses which can be reached from the Internet. The Course Evaluation System and Revocation Authority are equipped with their own servers and database repositories. The University Registration System is equipped with the p-ABC system server, the IDM server and its repository. DMZ includes both Patras Portal and the IDM public directory. All incoming http/https requests go thought DMZ. In order to protect http/https incoming traffic we will implement access lists and rules according risk management requirements of Patra's Pilot. In order to have different roles of users, University Registration System and Course Evaluation System will be employed with different user accounts. The students will have local user accounts on the University Registration System. Administrators will use Active Directory in order to have access to the Course Evaluation System or the University Registration System via CTI's internal network or internet. Moreover lectors will have local user accounts on the Course Evaluation System for uploading their course questionnaire.

In the University Pilot there are several groups of users (for more details see section 1.4.1 of D7.1). Students, lecturers, CTI personnel and members of HQAA can access the University pilot via the Internet. The Course lecturers will make an http request to the Course Evaluation System in order upload the questionnaires. Students will establish an http connection (via User's Home) with the Course Evaluation System for evaluating courses, with the Patras Portal for getting instructions and with, University Registration System for registering. When the evaluation procedure is completed,

CTI members and HQAA members will be able to make an http request to Course Evaluation System for processing or viewing the evaluation results. Professors, HQAA members and administrators will access the Course Evaluation System by using login name and password authentication. When a user make an http request via internet, his http connection will be routed from GRNet router to CTI's border router, and then after passing the firewalls and the internal routers the user will access the University Registration System or the Course evaluation Application in the DMZ by using the ABC technology.

Administrators of the Patras Pilot can access the DMZ via https/http/ssh/ldap connections through CTI's internal network or the Internet via a VPN connection. Administrators will follow the above traffic path like users in order to establish an https/http connection. When an administrator wants to establish ssh/ldap connection via the Internet, he must establish a VPN connection though a secure channel in order to get access the CTI's internal network and get a virtual IP at CTI's internal network. For more details for the authentication of VPN connections see [TN1029].

A university registration office employee can make a request to an administrator for revoking a student credential. An administrator has to establish VPN connection (via internet) to University Registration System in order to send the revocation information to the Revocation Authority.

The Domain Controller server verifies the administrator's credentials and AAA server[7] authenticates him and allows him to get access to CTI's internal network. Once the administrator can connect to CTI's internal network he can access to the University Registration System and Course Evaluation System in DMZ.

Also at the edge of our network there is a border router deployed in front of CTI's firewalls and performs some basic checks on network activity, such as ingress and egress filtering[8], that may be helpful for stopping some Internet-based worms from reaching the CTI's firewall. In border router we have implemented some generic access lists in order to increase the level of security and confront some types of malicious attacks like DoS or DDoS. Additionally in the two internal routers we have access lists, and through them we are able to specify which users or system processes are granted to access objects, as well as what operations are allowed on given objects, where objects are network devices, servers and workstations.

## 5.4 Hardware Requirements for Common Denominators

The ABC4Trust project has chosen two different locations for the pilots and two different entities responsible for setting up the required infrastructure. The already available CTI site will be enhanced with ABC4Trust network elements whereas EURODOCS will set up the pilot from scratch.

The school and university registration system provided by NSN consists of no specialised hardware components, besides a WLAN router with customized software to perform NAT.

By having a closer look at the pilots and at the number of participants it becomes clear, that speed is not an issue. This means that any state-of-the-art HW can easily host Privacy-ABC technology. For example: the entire IDM System including the DNS server, the LDAP DB and both IDM applications can be installed and would run easily on a single Linux Laptop with 4 GB RAM.

[7] Authentication Authorisation Access Server; a common element in Mobile Networks
[8] Ingress and egress filtering is done for intrusion detection, by establishing the origin of the communication.

Nevertheless, it is recommended to use scalable HW. For large installations with millions of users special hardware (i.e. 'carrier-grade') is required. Additionally one should consider active/standby entities (as it is done in Patras at the firewall). It should be noted that active/standby support requires significant additional coding in the applications that are out of scope of the two ABC4Trust pilots.

Some servers allow 'hot swapping' of the disks. In this case the operating systems would need to support this feature and replicate the data to both disks. But here the security risks must be compared to the advantages such a feature would bring. Removing one disk during operation wouldn't be noticed for some time so physical data theft would be very much easier.

For a basic installation an off-the-shelf rack configuration can be recommended, provided some scalability is required and the number of users stays below a few hundred. Examples for rack servers can be found in the Internet. The costs for a single rack server with 16GB RAM, 6 TB hard disk and an 8 core CPU with 3,6 GHz at this point in time were less than 600 Euros. Such a server can host multiple 'virtual servers' with different operating systems and different IP addresses. In effect, a single rack server can host all ABC4Trust entities (Issuer, Verifier, Restricted Area and Portal, etc..).

Every infrastructure requires routers and other equipment. This equipment could also be hosted by the above recommended rack server, by any other server (e.g. a laptop) or by specialized HW. CTI will deploy a firewall of type Cisco Pix-535 configured as active-standby, but it must be noted, that in smaller deployments, the hosted firewalls of any operating system can be sufficient.

Hubs and connectors are considered as being out-of-scope of this chapter.

## 5.5 Additional Generic Hardware Requirements

Both pilots make use of smart cards which can store (amongst other items) the User Secret and the credentials. So far, smart cards have not yet been deployed in the pilots so specification details of recommended cards are unknown at this point in time. The same applies for smart card readers. It must be noted that smart cards are not required for Privacy-ABC technology, but they do increase the level of security. The project will therefore prove in the Söderhamn pilot that selected Users can participate in the pilot without the usage of smart cards.

# 6 Outlook

The Privacy-ABC technology mechanisms are being deployed in the two pilots, thus showing that the Privacy-ABC technology is fit for everyday use. Dissemination planning is done in WP8 and current discussions include proposing ABC mechanisms to the standardization.

With the feedback of 300 pupils in Söderhamn and about 100 students in Patras, the lessons learned will improve not only the technology, but also the processes around the privacy-ABC technology. After this we hope to improve the surrounding ecosystem, so that the practical use can be validated in a much wider scope.

This deliverable provides a first step toward this adoption. Further updates will improve, as it reflects only the current state of the project. Additional information about the project, dissemination efforts and privacy-ABC technology development kit will be provided in future deliverables and the project homepage: https://abc4trust.eu .

# 7 Glossary

ABC4Trust Engine

The ABC4Trust engine implements the unified API interface to the different privacy-ABC technologies.

Anonymous

Anonymity of a subject means that the subject is not identifiable within a set of subjects, the anonymity set.

Attribute

A piece of information, possibly certified by a credential, describing a characteristic of a natural person or entity, or of the credential itself. An attribute consists of an attribute type determining the semantics of the attribute (e.g., first name) and an attribute value determining its contents (e.g., John).

Certified pseudonym

A verifiable pseudonym based on a user secret that also underlies an issued credential. A certified pseudonym is established in a presentation token that also demonstrates possession of a credential bound to the same User (i.e., to the same user secret) as the pseudonym.

Credential

A list of certified attributes issued by an Issuer to a User. By issuing a credential, the Issuer vouches for the correctness of the contained attributes with respect to the User.

Credential specification

A data artefact specifying the list of attribute types that are encoded in a credential.

Data Controller

"'Controller' shall mean the natural or legal person, public authority, agency or any other body which alone or jointly with others determines the purposes and means of the processing of personal data...", Art. 2 (d) of Directive 95/46/EC. In the area of Privacy-ABCs the Issuer, Verifier, the Revocation Authority and the Inspector are Data Controllers with the respective duties arising from the law.

Data Processor

"'Processor' shall mean a natural or legal person, public authority, agency or any other body which processes personal data on behalf of the controller", Art. 2 (e) of Directive 95/46/EC. Data Controllers processes personal data on behalf of the data Controller.

Data Subject

A data subject is an identified or identifiable natural person, Art. 2 (a) of Directive 95/46/EC. In the area of Privacy-ABCs the User and any other national person of which personal data is processes is a data subject. Data subjects have data subjects' rights assigned such as the right of access, rectification, erasure and blocking, Art. 12 of Directive 95/46/EC.

User binding

An optional credential feature whereby the credential is bound to a strong secret embedded in a dedicated hardware device so that any presentation token involving the credential requires the presence of the device.

Entity

Entity is anything that has a distinct existence; it is the fundamental "thing" that can be identified.

1. Digital entity is any Entity which primarily exists in some digital context, e.g., as a digitally encoded information or as a running computer program.
2. Legal entity is any Entity which has some sort of legal subjectivity, or which is legally recognized in a judicial system. *For the commentary text: Examples include besides natural persons (humans) also companies that have been granted legal subjectivity by the law such as stock corporations, limited liability companies etc.*
3. Physical entity is an entity for which some sort of physical constituent is compulsory.

InfoCards

InfoCards is a Microsoft Technology, where the user can choose which attributes to reveal by selecting a "business card". This technology facilitates thinking about privacy with its business card metaphor – not all people I will give my private/company business card.

Inspection

An optional feature allowing a presentation token to be de-anonymized by a dedicated Inspector. At the time of creating the presentation token, the User is aware (through the presentation policy) of the identity of the Inspector and the valid grounds for inspection.

Inspection grounds

The circumstances under which a Verifier may ask an Inspector to trace the User who created a given presentation token.

Inspection Requester

Entity requesting an inspection from the Inspector, asserting that inspection is compliant with the inspection grounds specified or is legally required. In most cases this will be the Verifier, but also may be the police, or other legally authorised entity.

Inspector

A trusted entity that can trace the User who created a presentation token by revealing attributes from the presentation token that were originally hidden from the Verifier.

Issuance key

The Issuer's secret cryptographic key used to issue credentials.

Issuer

The party who vouches for the validity of one or more attributes of a User, by issuing a credential to the User.

Issuer parameters

A public data artefact containing cryptographic and other information by means of which presentation tokens derived from credentials issued by the Issuer can be verified.

Linkability

See *unlinkability*.

PDP, Policy Decision Point

A server that evaluates a access request in accordance to a given policy and replies to the PEP with the decision.

PEP, Policy Enforcement Point

An entity that incorporates a functionality which only a selected group has access to, these people will request access, a access request will be send to the policy decision point and if it replies with a “grant” access given.

Personal data

“‘Personal data' shall mean any information relating to an identified or identifiable natural person ('data subject'); an identifiable person is one who can be identified, directly or indirectly, in particular by reference to an identification number or to one or more factors specific to her physical, physiological, mental, economic, cultural or social identity”, Art. 2 (a) of Directive 95/46/EC. Within this deliverable personal data is the terminology used for legal considerations. See also *Personally Identifiable Information*.

Personally Identifiable Information (PII)

Personally Identifiable Information is defined as any information about an individual maintained by an [entity], including any information that can be used to distinguish or trace an individual‘s identity, such as name, social security number, date and place of birth, and any other information that is linked or linkable to an individual ([NIST10] p. 2-1). PII is a widely used terminology for *personal data* in the domain of information security. Within this document PII is used in relation to information security.

Presentation policy

A policy created and published by a Verifier specifying the class of presentation tokens that the Verifier will accept. The presentation policy contains, among other things, which credentials from which Issuers it accepts and which information a presentation token must reveal from these credentials.

Presentation token

A collection of information derived from a set of credentials, usually created and sent by a User to authenticate to a Verifier. A presentation token can contain information from several credentials, reveal attribute values, prove that attribute values satisfy predicates, sign an application-specific message or nonce or support advanced features such as pseudonyms, device binding, inspection, and revocation. The presentation token consists of the presentation token description, containing a technology-agnostic description of the revealed information, and the presentation token evidence, containing opaque technology-specific cryptographic parameters in support of the token.

Privacy-ABC, p-ABC

The technology that employs Attribute Based Credentials (ABC) to protect privacy.

Pseudonym

See *verifiable pseudonym*.

Pseudonym scope

A string provided in the Verifier's presentation policy as a hint to the User which previously established pseudonym she can use, or to which a new pseudonym should be associated. A single User (with a single user secret) can generate multiple verifiable or certified pseudonyms for the same scope string, but can only generate a single scope-exclusive pseudonym.

Pseudonymous

The state where an Entity (User) is known to a party (Verifier, Issuer) by a Pseudonym, i.e., by a Partial Identity.

Revocation

The act of withdrawing the validity of a previously issued credential. Revocation is performed by a dedicated Revocation Authority, which could be the Issuer, the Verifier, or an independent third party. Which Revocation Authorities must be taken into account can be specified by the Issuer in the issuer parameters (Issuer-driven revocation) or by the Verifier in the presentation policy (Verifier-driven revocation).

Revocation Authority

The entity in charge of revoking credentials. The Revocation Authority can be an Issuer, a Relying Party, or an independent entity. Multiple Issuers or Verifiers may rely on the same Revocation Authority.

Revocation information

The public information that a Revocation Authority publishes every time a new credential is revoked or at regular time intervals to allow Verifiers to check that a presentation token was not derived from revoked credentials.

Revocation parameters

The public information related to a Revocation Authority, containing cryptographic information as well as instructions where and how the most recent revocation information and non-revocation evidence can be obtained. The revocation parameters are static, i.e., they do not change every time a new credential is revoked or at regular time intervals like the revocation information and non-revocation evidence (may) do.

RP – STS (Relying Party Secure Token Service)

Is a proxy service, that represents the relying party in one technical domain (e.g. SAML world) and issues a secure token in the web service world. This way applications of both worlds (mostly enterprise and web service) can share attributes.

Non-revocation evidence

The User-specific or credential-specific information that the user agent maintains, allowing it to prove in presentation tokens that the credential was not revoked. The non-revocation evidence may need to be updated either at regular time intervals or when new credentials are revoked.

Scope

See *pseudonym scope.*

Scope-exclusive pseudonym

A certified pseudonym that is guaranteed to be cryptographically unique per scope string and per user secret. Meaning, from a single user-bound credential, only a single scope-exclusive pseudonym can be generated for the same scope string.

Traceability

See *untraceability*.

Unlinkability

The property that different actions performed by the same User, in particular different presentation tokens generated by the same User, cannot be linked to each other as having originated from the same User.

Untraceability

The property that an action performed by a User cannot be traced back to her identity. In particular, the property that a presentation token generated by a User cannot be traced back to the issuance of the credential from which the token was derived.

User

The human entity who wants to access a resource controlled by a verifier and obtains credentials from Issuers to this end.

User agent

The software entity that represents the human User and manages her credentials.

User binding

An optional credential feature whereby the credential is bound to an underlying user secret. By requiring multiple credentials to be bound to the same user secret, one can prevent Users from "pooling" their credentials.

User secret

A piece of secret information known to a User (either a strong random secret or a human-memorizable password or PIN code) underlying one or more issued credentials or pseudonyms. A presentation token involving a pseudonym or a user-bound credential implicitly proves knowledge of the underlying user secret.

Verification Policy

The verification policy is the policy that determines who gets access e.g. to a certain service. Part of the policy is the list of attributes (properties) a user has to proof and in which way (e.g. inspectable way) it has to be done.

Verifiable pseudonym

A public identifier derived from a user secret allowing a User to voluntarily link different presentation tokens created by her or to re-authenticate under a previously established pseudonym by proving knowledge of the user secret. Multiple unlinkable pseudonyms can be derived from the same user secret.

Verifier

The party that protects access to a resource by verifying presentation tokens to check whether a User has the requested attributes. The Verifier only accepts credentials from Issuers that it trusts

# List of Acronyms

| | |
|---|---|
| Abbr | Abbreviation |
| ABCs | Attribute Based Credentials |
| Privacy-ABCs | Privacy Attribute Based Credentials (privacy ABCs) |
| ABCE | ABC Engine |
| CA | Certificate Authority |
| CE | Crypto Engine |
| CSV | Character Seperated Values, a file format |
| DFD | Data Flow Diagrams |
| DoW | Description of Work |
| DoS | Denial Of Service |
| GUI | Graphical User Interface |
| HTTP | Hypertext Transfer Protocol |
| HTTPS | HyperText Transfer Protocol Secure (HTTP secured by TLS or SSL) |
| HQAA | Hellenic Quality Assurance Agency |
| HW | Hard Ware |
| ID | Identifier |
| Idemix | IBM Identity Mixer |
| IDM | Identity Manager |
| LDAP | Light Directory Access Protocol |
| ISP | Internet Service Provider |
| NFC | Near Field Communication |
| OAuth | Open Authentication/Authorisation |
| OpenID | Open Identity |
| P-ABC | privacy Attribute Based Credentials |
| PC | Personal Computer |
| PIN | Personal Identification Number |
| PUK | PIN Unlock Key |
| RP | Relying Party |
| SAML | Secure Assertion Markup Language |
| SC | Smart Card |
| SCI | Smart Card Interface |
| SSL | Secure Sockets Layer |
| STS | Secure Token Service |

| | |
|---|---|
| TTP | Trusted Third Party |
| TLS | Transport Layer Security |
| URI | Uniform Resource Identifier |
| WP | Work Package |
| XML | eXtensible Markup Language |

# 8 Bibliography


[DSTICICC09] Directorate for Science, Technology and Industry Committee for Information, Computer and Communications. *POLICY, THE ROLE OF DIGITAL IDENTITY MANAGEMENT IN THE INTERNET ECONOMY: A PRIMER FOR POLICY MAKERS, 2009 http://www.oecd.org/dataoecd/55/48/43091476.pdf*

[BHRR07] P. Bramhall, M Hansen, K. Rannenberg, T Roessler *User-Centric Identity Management: New Trends in Standardization and Regulation* in Security & Privacy, IEEE , vol.5, no.4, pp.84-87, July-Aug. 2007

[RaRa2007] By Rakesh Radhakrishnan *Identity & Security: A Common Architecture & Framework For SOA and Network Convergence* Futuretext 2007

[ZLWLFZYDL09] Ye, Liu, Wang, Lv, Feng, Zhou, Yokota, Deng and Liu *Decomposition: Privacy Preservation for Multiple Sensitive Attributes* in Database Systems for Advanced Applications, p. 486-490, LNCS 5463, 2009

[LEH2010] Leif-Erik Holtz , Schwerpunkt*: Datenschutzkonformes Social Networking: Clique und Scramble!* In Datenschutz und Datensicherheit – DuD Volume 34, Number 7 (2010), 439-443, DOI: 10.1007/s11623-010-0125-0

[DIASPORA] http://diasporaproject.org *, last checked 30.5.2012*

[INFOCARDS] Information Cards.: Information Cards Foundation, 2009. Available at: http://informationcard.net/ , last checked 30.1.2010

[BN89] Dr. David F.C. Brewer and Dr. Michael J. Nash, *The Chinese Wall Security Policy*, in IEEE Symposium on Research in Security And Privacy May 1989, pp 206-214

[C04] Jan Camenisch, *Better Privacy for Trusted Computing Platforms,* Computer Security – ESORICS 2004, Lecture Notes in Computer Science, 2004, Volume 3193/2004, 73-88, DOI: 10.1007/978-3-540-30108-0_5

[MOASIS05] T Moses, *extensible access control markup language (xacml) version 2.0* Oasis Standard, 2005

[EPC99] Directive 1999/93/EC of the European Parliament and of the Council of 13 December 1999 on a Community framework for electronic signatures. Downloaded from http://europa.eu/legislation_summaries/information_society/other_policies/l24118_en.htm on 20 March 2012.

[J1S27W5] JTC 1/SC 27/WG 5, "ISO/IEC 29115 (DIS): Information technology - Security techniques - Entity authentication assurance", 2011.

[CKS10] Jan Camenisch, Markulf Kohlweiss and Claudio Soriente. *Solving revocation with efficient update of anonymous credentials.* In Proceedings of the 7th international conference on Security and cryptography for networks, SCN'10.

[LKDN10] Jorn Lapon, Markulf Kohlweiss, Bart De Decker, and Vincent Naessens. *Performance analysis of accumulator-based revocation mechanisms*. In Proceedings of the 25th International Conference on Information Security (SEC 2010), IFIP Conference Proceedings, page 12, Brisbane, AU, 2010. Springer-Verlag.

[CL02] Jan Camenisch and Anna Lysyanskaya. Dynamic accumulators and application to efficient revocation of anonymous credentials. In Moti Yung, editor, Advances in Cryptology —

CRYPTO 2002, volume 2442 of Lecture Notes in Computer Science, pages 61–76. Springer Verlag, 2002.

[PRIMELIFE] http://www.primelife.eu/, last checked 30.5.2012

[uPRIME] http://www.kau.se/en/computer-science/research/research-projects/u-prim , last checked 30.5.2012

[TN1029] http://technet.microsoft.com/en-us/library/cc785368%28v=ws.10%29.aspx , last checked 30.5.2012

[Zwi11] Harald Zwingelberg. Necessary processing of personal data: The need-to-know principle and processing data from the new German identity card. In Simone Fischer-Hübner, Penny Duquenoy, Marit Hansen, Ronald Leenes, and Ge Zhang, editors, *Privacy and Identity Management for Life*, volume 352 of *IFIP Advances in Information and Communication Technology*, pages 151–163. Springer Boston, 2011.