Architecture for Controlled Credential issuance Enhanced with Single Sign-On (ACCESSO)

Daniel Nemmert[1], Detlef Hühnlein[1], Tobias Wich[1], Tina Hühnlein[1]

Abstract: As more than half of the EU Member States already have rolled out electronic identity cards (eIDs) [Le13], it seems to be a rewarding approach to investigate whether and how eIDs may be used for the purpose of controlling the log-on process for operating systems and similar local access control facilities. While this paper shows that all currently rolled out eIDs may be used for such access control purposes, our investigation also reveals that for some types of eIDs it is significantly harder to support this kind of use case.

Keywords: electronic identity cards (eIDs), access control management, operating system log-on.

1 Introduction

Using two factor authentication for machine log-ins received a serious boost when Apple introduced a fingerprint reader on its mobile devices in 2013. Other manufacturers followed and provide similar biometric readers in their products. Looking at the desktop and laptop market, a quite different picture is shown as fingerprint readers are usually only added to expensive business models.

The other type of candidates for two-factor authentication are smart cards and hardware based authentication tokens in general. In the past this technology has mainly been available to enterprises with large scale deployments and their own Public-Key-Infrastructure (PKI). Nowadays many countries issue eID cards to their citizens leading to the broad availability of secure cryptographic hardware devices. However these cards are mostly intended for online eGovernment use and in some cases to perform digital signatures. Software to integrate these cards into the machine log-on procedure is usually not available or the documentation on how to configure these cards for this purpose is missing. Furthermore there are authentication tokens which are not capable of performing the necessary operations to simply hook into the available log-in frameworks, such as the German eID ("Personalausweis") or FIDO tokens (cf. [FI]) for example.

Information about procedures that can be used in the scope of an operating system log-in is mainly available in the form of standards (cf. RFC 4210 [Ad05], RFC 4556 [ZT06]) or open specifications (eg. Pluggable Authentication Module (PAM)[2]). The field of research concerning this topic seems mainly to be done by companies in internal projects or in the case of PAM in an open source development environment.

[1] {daniel.nemmert,detlef.huehnlein,tobias.wich,tina.huehnlein}@ecsec.de,ecsec GmbH, Sudetenstraße 16, 96247 Michelau

[2] `http://www.linux-pam.org`

This paper presents a universal Architecture for Controlled Credential issuance Enhanced with Single Sign On (ACCESSO), which abstracts the technical details of arbitrary authentication tokens in a way which allows to integrate them into major log-in frameworks, such as Microsoft's Winlogon [In15, Inf] or the Pluggable Authentication Module (PAM) architecture [Th97], which is supported by Linux and Mac OS X. The ACCESSO approach either directly uses the provided capabilities and X.509 certificates of the authentication tokens or generates the required X.509 certificates on the fly after an according authentication has been performed with some alternative authentication method.

2 Background

The ACCESSO proposal integrates eID technology into access control frameworks and hence, we will briefly provide the necessary background with respect to these topics here: In Section 2.1 the various widespread hardware tokens, especially eID cards and the authentication protocols they use, will be discussed. After that in Section 2.2 the architecture and extension capabilities of the major authentication frameworks are considered.

2.1 eID Cards and Authentication Tokens

With respect to currently rolled out eIDs in Europe (cf. FutureID D32.1 [Le13]) one may distinguish between X.509-based eIDs and non-X.509-based eIDs:

- *X.509-based eIDs*
  The majority of the currently deployed eIDs in Europe are equipped with an X.509 certificate, which can be used for authentication and signature purposes. Examples include Finland, Estonia, Sweden, Belgium, Portugal, Italy, Luxembourg and Spain. These type of eIDs may be used for plain challenge-response based authentication protocols in which the eID computes a digital signature over a challenge, which may be chosen at random or be derived from previous protocol messages as in the case of TLS (cf. RFC 5246 [DR08]) for example. While such eIDs may be utilized in a variety of applications, their drawback is that they are not optimal from a privacy point of view, because the legitimate use of the eID in the internet can trivially be misused to create user profiles.

- *non-X.509-based eIDs*
  On the other hand there are non-X.509-based eIDs, such as the German Identity Card (Personalausweis) for example, which uses the Extended Access Control (EAC) protocol as defined in BSI-TR-03110 [Fe15a] for authentication. Such eIDs have great advantages with respect to privacy, as the protocol ensures that only identified entities which have a valid authorization certificate can access the data stored on the card and it is not possible to create user profiles in the internet.
  Furthermore there may be entirely different means for identification and authentication as the eIDAS-regulation (cf. [20114] and [20115]) is fully technology neutral

and the only standardized interface for integrating such eIDs, which is likely to be endorsed by the European Commission, is the SAML-based interface defined in the eIDAS Technical Specifcation [eTS15].

2.2 Access Control Frameworks

Apart from some exotic systems, all current operating systems use a log-in framework to authenticate and authorize their users. In the Unix world the Pluggable Authentication Module (PAM) framework [Th97] is the framework of choice, whereas Microsoft uses the Winlogon System [In15, Inf] for this purpose. We briefly introduce PAM and then focus mostly on the architecture of the Winlogon Framework, because it seems to be more widespread and has a very solid integration of X.509 based signature cards in combination with the Kerberos-based Active Directory (AD) system.

2.2.1 Pluggable Authentication Module (PAM) Framework

PAM was originally proposed by Sun Microsystems in 1995 in the form of an Open Software Foundation RFC [SS95] and later became the Unix standard "X/Open Single Sign-on Service" (XSSO) [Th97]. The specification provides a coarse architecture and a set of C APIs to build a log-in framework as it can be seen in Fig. 1, which has been extracted from the XSSO specification [Th97]. Module implementors must only follow the specification and their modules can run on any system that uses PAM. A wide variety of authentication modules exist for PAM. The most important ones are the Kerberos, LDAP and PKCS#11 authentication plugin. But there are also modules for fingerprint readers, OTP tokens, Bluetooth devices and even Google Authenticator.

2.2.2 Microsoft's Winlogon Framework

Winlogon, in contrast to PAM, is not just a specification by Microsoft, but also an implementation with a fine grained architecture and utility functionality. It has been improved with every Windows version. The biggest change happened in the switch to Windows Vista, where the Graphical IdentificatioN and Authentication (GINA) module [Ing] was replaced by the Credential Providers API [Inf]. Credential Providers are represented by different log-on tiles on the desktop. Credential Providers do not enforce policies or grant access to the operating system, but they are instead used to collect and serialize various types of credentials.

While the high level view of the Winlogon architecture as shown in Fig. 2 seems to be similar to the PAM architecture depicted in Fig. 1, there are two major differences: First there are user interfaces (UI), which provide rich features to display forms to communicate with the user. The second significant difference is represented by the Local Security Authority (LSA). It is a component taking care of authenticating the user with a remote authentication protocol or against some local password database for example. This leads to a looser

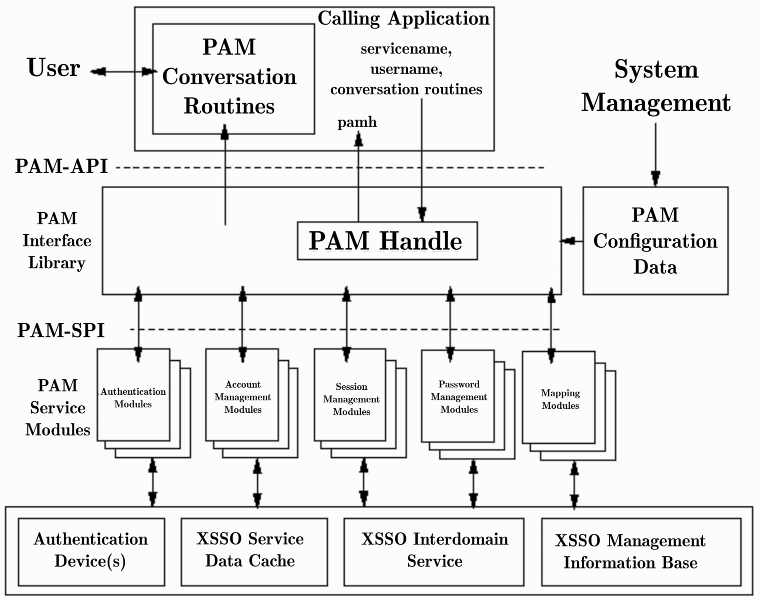

Fig. 1: PAM Framework (source: [Th97, Figure 4-1])

coupling between the authentication protocol such as Kerberos and the credential when compared to the PAM architecture. A high level overview of the Windows log-on process since Windows Vista is presented in [In14]. The Smart Card Infrastructure is outlined in [Ine].

3 Architecture for Controlled Credential issuance Enhanced with Single Sign On (ACCESSO)

In this paper we present a novel and unified system architecture for operating system login that supports arbitrary eID and authentication tokens called ACCESSO (Architecture for Controlled Credential issuance Enhanced with Single Sign On), which even allows to support the German eID for controlling access to computers running Windows, Linux or MacOS X for example.

3.1 Overview of ACCESSO

The ACCESSO architecture is depicted in Fig. 3 and comprises a Client, a Server and an Identity Management component:

The *ACCESSO Server* component is a regular Microsoft Domain Controller, which comprises the Active Directory Service (ADS) and a Key Distribution Center (KDC) [Inb], which supports the Kerberos protocol according to RFC 4120 [Ne05] together with the

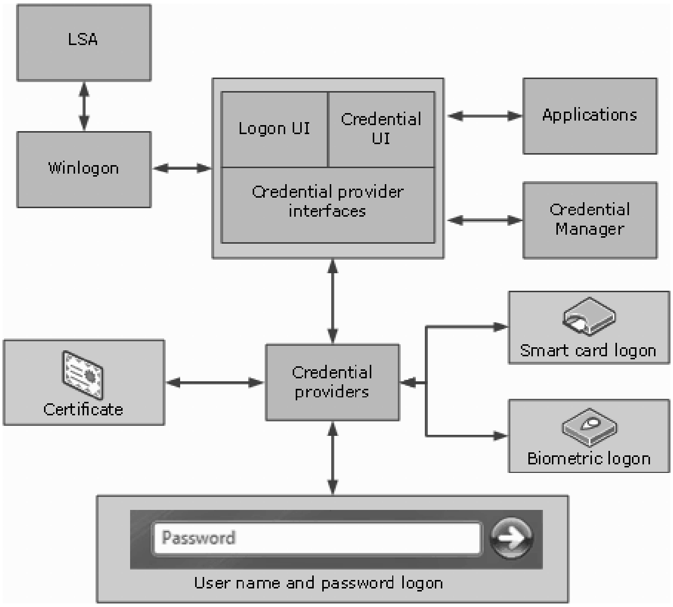

Fig. 2: Winlogon Framework Architecture (source: [In10])

PKINIT enhancements according to RFC 4556 [ZT06] as profiled in [Ind]. The KDC is started by its Local Security Authority (LSA) and is a part of the LSA's process. It is responsible for issuing Ticket-Granting Tickets (TGT) which are presented to the Ticket-Granting Service (TGS) of a domain to gain access to a system.

The *ACCESSO Identity Management* contains a variety of *Authentication Services* (AS) and a *Credential Producer* (CP), which upon request creates certificates and credentials, that are necessary for performing the log-in process. This component may be realized "on-premise" as additional server systems, or the necessary functionality could be obtained as a service from the SkIDentity Service which has received multiple German and European awards and is also patented[3].

The certificate authority used for the issuance of the required certificates has to be trustworthy enough to be suitable for usage in an operating system logon scenario. The validity period of the issued certificate is out of scope for this paper because it has to be defined on a case by case basis, since for a private user a much longer validity of the certificate would be feasible than for example for an institution that handles critical and classified information.

The *ACCESSO Client* contains various modules, which are introduced in the following:

- The *Logon* component represents the Winlogon and the PAM-Interface library on Windows and Unix systems respectively and initiates the Single Sign-On process. It

[3] cf. `https://skidentity.com`, [Hü15a] and [Hü15b] for example.

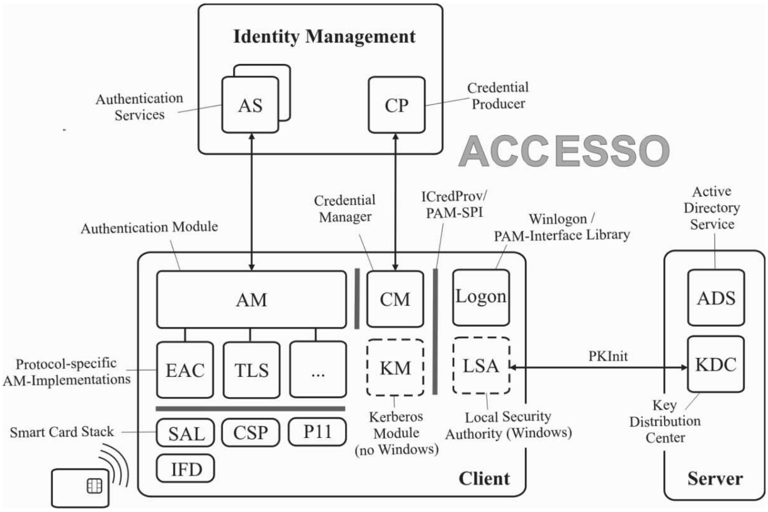

Fig. 3: Architecture for Controlled Credential issuance Enhanced with Single Sign-On (ACCESSO)

is also responsible for the communication with security enforcement points, like the Local Security Authority (LSA) in Windows operating systems.

- The *Credential Manager* (CM) is responsible for generating a key pair on demand and starting the authentication process. The CM also collects the created credential (e.g. a CloudID[4]) and serializes it before it is used in the access decision procedure.
- The *Authentication Module* (AM) provides several protocol-specific implementations that are dependent on which type of credential is being used. For X.509-based eIDs this component can be a TLS implementation. For the German eID an implementation of the EAC protocol is needed. To read information from smart cards, the Smart Card Stack provides components that enable reading of e.g. certificates from smart cards and possibly other types of authentication tokens. Possible modules in the Smart Card Stack include a PKCS#11 module, an Interface Device (IFD) for the German eID that implements an IFD-Interface specified in ISO/IEC 24727-4 [IS08b] and BSI TR-03112-6 [Fe12a] or a Service Access Layer (SAL) that provides an API for client-applications according to ISO/IEC 24727-3 [IS08a].
- The *Local Security Authority* (LSA) (cf. [Inc]) is part of the Windows log-on process and is responsible for authenticating users via an Authentication Package (AP) (cf. [Ina]) (typically Kerberos) and enforcing security and access decision from the AP. The Local Security Policy of a system is maintained through the LSA as well.
- In non-Windows systems there needs to be an additional *Kerberos Module* (KM), which would talk to the KDC in the ACCESSO Server.

[4] cf. `https://www.skidentity.de/en/system/cloud-identity`

3.2 The ACCESSO Protocol

The goal of ACCESSO is to make it possible that arbitrary eID cards and authentication tokens can be used for operating system logon. As some tokens (cf. Section 2.1) are not equipped with an X.509 certificate, the ACCESSO system creates the necessary certificate on the fly after an according authentication has taken place.

While it would seem to be straightforward to use the Certificate Management Protocol (CMP) specified in RFC 4210 [Ad05] for issuing the required credential, the required authentication is neither specified nor seriously considered in CMP and hence CMP does not seem to be a good choice, especially when incorporating complex protocols such as the Extended Access Control (EAC) protocol required for supporting the German eID card.

As only the certificate issuance is needed from the CMP protocol, an alternative solution would be to use authentication protocols capable of exchanging arbitrary data inside the protocol messages. An obvious choice in this regard would be SAML v2.0 [Ca05a], because of its extensibility features and the possibility to send arbitrary data in the authentication request, which is not always the case in other protocols. SAML also has good support in major identity management frameworks and services, such as SkIDentity for example.

In a classical Web SSO setting with SAML, a web browser provides the user agent part of the protocol, which communicates with the Service Provider (SP) and Identity Provider (IdP). However, since the Windows, Linux and Mac OS X log-ons are rather constrained environments, using a browser as the user agent is out of question. As a consequence the Web Browser SSO Profile is not applicable in this scenario. A possible solution would be the SAML 2.0 SSO Profile “Enhanced Client or Proxy” (ECP) [Sc13]. The profile was specifically designed for clients such as desktop applications and anything else which is not a web browser. The ECP application takes the role of the the user agent in the Web SSO SAML profile. A client application using the SAML ECP profile, may be capable of directly initiating an authentication protocol to be performed with the selected IdP as outlined in [Ja10]. As in the Web-SSO profile, there is no means of negotiating the authentication protocol or protocol endpoint, which may lead to complex and hard to maintain client configurations in case of dynamic IdP discovery and the support of multiple tokens is required.

A solution to the discovery issue is given in the “SAML Privacy-Enhancing Profile” [HTW14] that originated from the research project FutureID[5], which was funded by the European Commission in the 7th Framework Programme. This profile is based on the SAML Web Browser SSO profile [Ca05b] and the profile defined in the Technical Guideline TR-03124-1 [Fe15b] of the German Federal Office for Information Security. It allows users to know and choose the IdPs and their supported authentication protocols in advance, saving the user from contacting each possible IdP. Besides that this is primarily benefiting the privacy of the user, it also provides the missing details to use SAML ECP with arbitrary IdPs and credentials.

[5] cf. `http://futureid.eu`

3.3 Protocol Flow

The ACCESSO protocol is implemented in the communication between the CM and the CP and the authentication components (AS and AM). Fig 4 shows the communication between these components in the context of the complete log-on procedure. The following protocol flow corresponds to the flow in a Windows environment. The sequence diagram is meant to provide an overview of the procedure and omits or simplifies steps, which are not necessary for understanding the underlying idea.

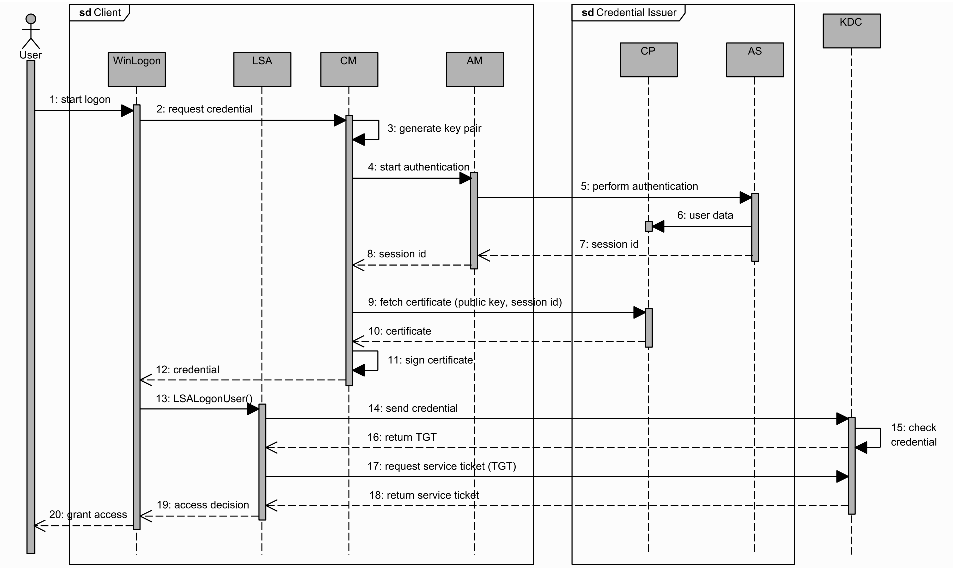

Fig. 4: ACCESSO Sequence Diagram

(1) The user starts the log-on process.

(2) The Winlogon process requests the user credential from the CM.

(3) The CM generates a public/private key pair, if required.

(4) The CM starts the authentication and sends the generated public key to the AM.

(5) In this step the authentication is performed by suitable communication between AM and AS.

(6) The AS sends the user data gathered during authentication, which may comprise an identifier or other identity attributes of the user, to the CP.

(7/8) Parallel to sending the user data to the CP, the AS sends the valid session id back to the CM via the AM.

(9/10) The CM uses the session id and probably additional holder-of-key data to fetch the certificate from the CP.

(11) The CM signs the credential with the generated private key.

(12) The CM serializes the credential according to the needs of the authorization package (in this case the KDC) and returns the created credential to the Winlogon process.

(13) The Winlogon process calls the LSA and provides the generated credential. The call LSALogonUser() starts the access decision procedure.

(14) The LSA sends the credential to the Authorization Package in use, in this case Kerberos.

(15) The KDC checks the credential against a database (typically an Active Directory) to find a valid user for it.

(16) If the credential is valid, the KDC returns a Ticket Granting Ticket (TGT) to the LSA.

(17) The LSA enforces the access decision and returns the decision to the Winlogon process.

(18) Winlogon informs the user about the access decision and grants or denies access.

4 Related Work

For the scope of this paper, the following topics provided relevant information.

The concept of **derived credentials** is not a new one. In october 2014 the National Institute of Standards and Technology (NIST) released Special Pubblication 800-157 [Naa] which contains guidelines for derived “Personal Identity Verification (PIV)” credentials.
One implementation performing the creation of derived credentials is SkIDentity[6]. The service uses the identity information provided by an electronic identity document (eID) to create a so-called *Cloud Identity* (CloudID) which theoretically enables the use of any electronic identity document for authentication. The created CloudID can even be transferred to a mobile phone for (pseudonymous) authentication purposes with a strong identity token.
In 2013 Schröder and Morgner explained the concept of eIDs with derived credentials [SM13] and the advantages over the classic username and password, which is still the most widespread log-in procedure today.

The Technical Guideline BSI-TR-03124-1 [Fe15b] defines how an eID-Client for the **German eID-Card** (defined in BSI-TR-03127 [Bu15]) interacts with an eID-Server in order to perform an authentication. Credentials such as SAML Tokens can then be retrieved from the eID-Server according to the Technical Guideline BSI-TR-03130 [Fe12b].

An example for an upcoming project in the field of derived credentials is **FIDELIO**[7] by the German Federal Office for Information Security which aims to link FIDO and the German eID-Card for online authentication. Results of this project will include the implementation

[6] https://www.skidentity.com

[7] https://www.evergabe-online.de/tenderdocuments.html?1&id=129326

of a FIDELIO-Authenticator-Client and FIDELIO-eService and the operation a FIDELIO-Server for test purposes.

Operating System Logon solutions are provided by various companies. SecureAuth offers a Windows Logon module with the SecureAuth Identity Provider[8]. Yubico offers the possibility to associate a YubiKey with a Microsoft Windows Account[9] for both enterprises and individuals. Enterprise users can use certain YubiKey versions like any other smart card supporting NIST SP 800-73 Personal Identity Verification Card (PIV) [Nab].

5 Conclusions and Outlook

The ACCESSO architecture and protocol outlined in the present paper makes it possible to use arbitrary authentication tokens, including the German eID card, FIDO tokens or OTP tokens for example, for secure operating system logon. This is achieved by generating the X.509 certificates necessary for the Windows logon on the fly. As the protocol may be based on existing SAML profiles, it should be easy to integrate into existing infrastructures based on this protocol.

Besides future research on the topic, further work needs to be spent on creating production ready implementations for the major operating systems. While our research investigated the feasibility of the ACCESSO approach for Microsoft Windows and the PAM implementation used in most Linux distributions, it is not said that other implementations such as the one in Apple's OSX allow these kind of extensions to the login framework.

References

[20114] Regulation (EU) No 910/2014 of the European Parliament and of the council on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC), 2014. http://data.europa.eu/eli/reg/2014/910/oj.

[20115] Commission Implementing Regulation (EU) 2015/1501 of 8 September 2015 on the interoperability framework pursuant to Article 12(8) of Regulation (EU) No 910/2014 of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market (Text with EEA relevance)), 2015. http://data.europa.eu/eli/reg_impl/2015/1501/oj.

[Ad05] Adams, C.; Farrell, S.; Kause, T.; Mononen, T.: . Internet X.509 Public Key Infrastructure Certificate Management Protocol (CMP). Request For Comments – RFC 4210, 2005. http://www.ietf.org/rfc/rfc4210.txt.

[Bu15] Bundesamt für Sicherheit in der Informationstechnik: . Architektur elektronischer Personalausweis und elektronischer Aufenthaltstitel. Technische Richtlinie (BSI-TR-03127), Version 1.16, 14.10.2015, 2015. https://www.bsi.bund.de/DE/Publikationen/TechnischeRichtlinien/tr03127/tr-03127.html.

[8] https://www.secureauth.com/IdP.aspx

[9] https://www.yubico.com/why-yubico/for-individuals/computer-login/windows-login/

[Ca05a] Cantor, Scott; Kemp, John; Philpott, Rob; Maler, Eve: . Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V2.0. OASIS Standard, 15.03.2005, 2005. http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf.

[Ca05b] Cantor, Scott; Kemp, John; Philpott, Rob; Maler, Eve: . Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0. OASIS Standard, 15.03.2005, 2005. http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf.

[DR08] Dierks, T.; Rescorla, E.: . The Transport Layer Security (TLS) Protocol Version 1.2. Request For Comments – RFC 5246, August 2008. http://www.ietf.org/rfc/rfc5246.txt.

[eTS15] eIDAS Technical Subgroup: . eIDAS Technical Specifications v1.0, November 2015. https://joinup.ec.europa.eu/software/cefeid/document/eidas-technical-specifications-v10.

[Fe12a] Federal Office for Information Security (Bundesamt für Sicherheit in der Informationstechnik): . eCard-API-Framework – IFD-Interface. Technical Directive (BSI-TR-03112), Version 1.1.2, Part 6, 2012. http://docs.ecsec.de/BSI-TR-03112-6.

[Fe12b] Federal Office for Information Security (Bundesamt für Sicherheit in der Informationstechnik): . eID-Server. Technical Directive (BSI-TR-031030), Version 1.6, 20.04.2012, 2012. http://docs.ecsec.de/BSI-TR-03130.

[Fe15a] Federal Office for Information Security (Bundesamt für Sicherheit in der Informationstechnik): . Advanced Security Mechanisms for Machine Readable Travel Documents and eIDAS Token. Technical Directive (BSI-TR-03110), Version 2.20, Part 1-4, 2015. https://www.bsi.bund.de/DE/Publikationen/TechnischeRichtlinien/tr03110/index_htm.html.

[Fe15b] Federal Office for Information Security (Bundesamt für Sicherheit in der Informationstechnik): . eID-Client – Specifications. Technical Directive (BSI-TR-03124), Version 1.2, Part 1, 2015.

[FI] FIDO Alliance: . FIDO Alliance Specifications (UAF and U2F). https://fidoalliance.org/specifications/download/.

[HTW14] Horsch, Moritz; Tuengerthal, Max; Wich, Tobias: SAML Privacy-Enhancing Profile. In (Hühnlein, Detlef; Rossnagel, Heiko, eds): Proceedings of Open Identity Summit 2014. volume 237 of LNI. GI, pp. 11–22, 2014.

[Hü15a] Hühnlein, Detlef; Tuengerthal, Max; Wich, Tobias; Hühnlein, Tina; Biallowons, Benedikt: Innovative Building Blocks for Versatile Authentication within the SkIDentity Service. In (Hühnlein, Detlef; Rossnagel, Heiko; Kuhlisch, Raik; Ziesing, Jan, eds): Proceedings of Open Identity Summit 2015. volume 251 of LNI. GI, pp. 141–152, 2015. http://www.ecsec.de/pub/OID_2015_SkIDentity_Service.pdf.

[Hü15b] Hühnlein, Tina: . Method and device for authentification. European Patent, EP 2439900, 2015. https://register.epo.org/application?number=EP11184139&lng=en&tab=main.

[Ina] Inc., Microsoft: . Authentication Packages. https://msdn.microsoft.com/en-us/library/windows/desktop/aa374733%28v=vs.85%29.aspx.

[Inb] Inc., Microsoft: . Key Distribution Center. https://msdn.microsoft.com/de-de/library/windows/desktop/aa378170%28v=vs.85%29.aspx.

[Inc] Inc., Microsoft: . LSA Authentication. https://msdn.microsoft.com/en-us/library/windows/desktop/aa378326%28v=vs.85%29.aspx.

[Ind] Inc., Microsoft: . Public Key Cryptography for Initial Authentication (PKINIT) in Kerberos Protocol. http://download.microsoft.com/download/9/5/E/95EF66AF-9026-4BB0-A41D-A4F81802D92C/[MS-PKCA].pdf.

[Ine] Inc., Microsoft: . Windows Vista Smart Card Infrastructure. https://msdn.microsoft.com/en-us/library/bb905527.aspx.

[Inf] Inc., Microsoft: . Winlogon and Credential Providers. https://msdn.microsoft.com/de-de/library/windows/desktop/bb648647%28v=vs.85%29.aspx.

[Ing] Inc., Microsoft: . Winlogon and GINA. https://msdn.microsoft.com/de-de/library/windows/desktop/aa380543%28v=vs.85%29.aspx.

[In10] Inc., Microsoft: . Windows Interactive Logon Architecture, February 2010. https://technet.microsoft.com/en-us/library/ff404303%28WS.10%29.aspx.

[In14] Inc., Microsoft: . Credentials Processes in Windows Authentication, May 2014. https://technet.microsoft.com/en-us/library/dn751047(v=ws.11).aspx.

[In15] Inc., Microsoft: . How Interactive Logon Works, March 2015. https://technet.microsoft.com/en-us/library/cc780332%28v=ws.10%29.aspx#w2k3tr_intlg_how_tpxs.

[IS08a] ISO/IEC: . Identification cards – Integrated circuit cards programming interfaces – Part 3: Application programming interface, ISO/IEC 24727-3. International Standard, 2008.

[IS08b] ISO/IEC: . Identification cards – Integrated circuit cards programming interfaces – Part 4: API Administration, ISO/IEC 24727-4. International Standard, 2008.

[Ja10] Jan Eichholz and Detlef Hühnlein and Gisela Meister and Johannes Schmölz: New Authentication concepts for electronic Identity Tokens. In: Proceedings of ISSE 2010. Vieweg, pp. 26–38, 2010. https://www.ecsec.de/pub/ISSE2010.pdf.

[Le13] Lehmann, Anja et al.: . Survey and Analysis of Existing eID and Credential Systems. FutureID Deliverable, D32.1, Version 1, March 2013. http://www.futureid.eu/data/deliverables/year1/Public/FutureID_D32.1_WP32_v1.0_Survey%20of%20existing%20eID%20and%20credential%20systems.pdf.

[Naa] National Institute of Standards and Technology: . Guidelines for Derived Personal Identity Verification (PIV) Credentials. NIST Special Publication 800-157. http://dx.doi.org/10.6028/NIST.SP.800-157.

[Nab] National Institute of Standards and Technology: . Interfaces for Personal Identity Verification – Part 1: PIV Card Application Namespace, Data Model and Representation. NIST Special Publication 800-73-4. http://dx.doi.org/10.6028/NIST.SP.800-73-4.

[Ne05] Neuman, C.; Yu, T.; Hartman, S.; Raeburn, K.: . The Kerberos Network Authentication Service (V5). Request For Comments – RFC 4120, July 2005. http://www.ietf.org/rfc/rfc4210.txt.

[Sc13] Scott Cantor et al.: . SAML V2.0 Enhanced Client or Proxy Profile Version 2.0. OASIS Committee Specification 01, 26.08.2013, 2013. docs.oasis-open.org/security/saml/Post2.0/saml-ecp/v2.0/cs01/saml-ecp-v2.0-cs01.html.

[SM13] Schröder, Martin; Morgner, Frank: . eID mit abgeleiteten Identitäten. Datenschutz und Datensicherheit (DUD), 2013. https://www.bundesdruckerei.de/sites/default/files/documents/2013/08/fachartikel_dud_abgeleitete_identitaeten.pdf.

[SS95] Samar, V.; Schemers, R.: . Unified Login with Pluggable Authentication Modules (PAM). Open Software Foundation Request for Comments 86.0, October 1995. http://www.kernel.org/pub/linux/libs/pam/pre/doc/rfc86.0.txt.gz.

[Th97] The Open Group: . X/Open Single Sign-on Service (XSSO) - Pluggable Authentication Modules. X/Open Document Number: P702, Preliminary Specification, 1997. http://www.opengroup.org/onlinepubs/008329799/toc.htm.

[ZT06] Zhu, L.; Tung, B.: . Public Key Cryptography for Initial Authentication in Kerberos (PKINIT). Request For Comments – RFC 4556, June 2006. http://www.ietf.org/rfc/rfc4559.txt.