Wilhelm Büchner Hochschule
Darmstadt
Fachbereich Informatik

Sichere Nutzung der AusweisApp

vorgelegt bei: Jürgen Kühnlein
von: Ulrich Gabele
Matr.-Nr. 870555
Anschrift: Usinger Straße 16 E, 65719 Hofheim
Abgabetermin: 01.01.2012

Vorwort

Ich möchte mich bei den Herren Andreas Fuchs und Sven Vowé vom Fraunhofer SIT bedanken, die sich für Gespräche mit mir Zeit genommen haben und mir wertvolle Informationen und Tipps für meine Arbeit gegeben haben. Vielen Dank an Herrn Jürgen Kühnlein für seine Hochschulbetreuung. Des Weiteren bedanke ich mich bei meiner Frau und meinen Kindern für ihre laufende Unterstützung.

Ulrich Gabele

Inhaltsverzeichnis

Abbildungsverzeichnis[1]

[1] Die Sequenzdiagramme in den Abbildungen 7, 11 und 15 sind mit UModel von Altova erstellt worden.

Tabellenverzeichnis

Materialanhang

DVD ISO-Image: Qubes OS Release beta 1

CD
- Ausschreibung der Diplomarbeit „Sichere Nutzung des nPA-Bürger-Clients“ des Fraunhofer-Instituts für sichere Informationstechnologie
- Installationsanleitung
- Software
    - AusweisApp: AusweisApp_010300_i686.deb
    - Treiber Kartenleser: scl011_2.06_linux_32bit
    - Keylogger: pykeylogger-1.2.1
    - Browser: firefox-3.0.tar.bz2
    - domscripttrustedbrowser
    - domscriptstart
    - ausweisapp-2-starte-trusted-Browser.desktop
    - ausweisapp-3-Kommunikationstarte-eID-trusted-Browser.desktop.

Abkürzungsverzeichnis

ADPU	Application Protocol Data Units
API	Application Programming Interface
AppVM	Virtuelle Maschine für Applikationen
AusweisApp	Ausweisapplikation
bit	binary digit
BITKOM	Bundesverband Informationswirtschaft, Telekommunikation und neue Medien e.V.
BSDG	Bundesdatenschutzgesetz
BSI	Bundesamt für Sicherheit in der Informationstechnik
BVA	Bundesverwaltungsamt
bzgl.	bezüglich
bzw.	beziehungsweise
CA	Chip Authentication
ca.	circa
CCC	Computer Chaos Club
CD	Compact Disc
CPU	Central Processing Unit
CSCA	Country Signing Certificate Authority
CVCA	Country Verifying Certificate Authority
CV-Certificate	Card Verifiable Certificate
CVE	Common Vulnerabilities and Exposures
d.h.	das heißt
DMA	Direct Memory Access
DV	Document Verifier
DVCA	Document Verifying Certificate Authority
DVD	Digital Versatile Disc
EAC	Extended Access Control
eCard	electronic Card
ECDH	Elliptic Curve Diffie-Hellman
ECDSA	Elliptic Curve Digital Signature Algorithmus
EDV	Elektronische DatenVerarbeitung

eID	elektronischer Identitätsnachweis
E-Mail	Electronic Mail
ePass	Biometrie-Anwendung
ePerso	elektronischer Personalausweis
eSign	Signaturanwendung
et al.	und andere
eth	ethernet interface
FH	Fachhochschule
Fraunhofer-SIT	Fraunhofer-Institut für sichere Informationstechnologie
FSK	Freiwillige Selbstkontrolle
G	Grundschutzkatalog
ggf.	gegebenenfalls
GUI	Graphical User Interface
HTML	HyperText Markup Language
HTTP	Hypertext Transport Protocol
HTTPS	Hypertext Transport Protocol Secure
ID	Identität
inkl.	inklusive
I/O	Input/Output
IP	Internet Protocol
ISO	International Organization for Standardization
IT	InformationsTechnik
ITEF	Internet Engineering Task Force
Kde	früher: K Desktop Environment, heute Bezeichnung der Entwicklungsgemeinschaft
KDF	Key Derivation Function
LAN	Local Area Network
MIME	Multipurpose Internet Mail Extensions
MHz	Mega Hertz
nPA	neuer Personalausweis
OASIS	Organization for the Advancement of Structured Information Standards
OS	Operating System

OWOK	One Web, one Key
P	Punktewertzahl
PA	Passive Authentication
PACE	Password Authenticated Connection Establishment
PAOS	Protocol Access Object Simple => reversed HTTP binding for SOAP
PAuswG	Personalausweisgesetz
PC	Personal Computer
PIN	Persönliche IdentifikationsNummer
PKI	Public Key Infrastructure
PUK	Personal Unblocking Key
PVUSB	ParaVirtualized Universal Serial Bus
QES	Qualifizierte Elektronische Signatur
RAM	Random Access Memory
RFC	Requests for Comments
RFID	Radio Frequency Identification
RPC	Remote Procedure Call
SAML	Security Assertion and Markup Language
SOAP	Simple Object Access Protocol
SSH	Secure Shell
SSL	Secure Socket Layer
S.	Seite
TA	Terminal Authentication
TLS	Transport Layer Security
TPM	Trusted Platform Module
TR	Technische Richtlinie
TXT	Trusted Execution Technology
URB	USB Request Block structure
USB	Universal Serial Bus
VM	Virtuelle Maschine
W	Gewichtung
WDR	Westdeutscher Rundfunk
VfB	Vergabestelle für Berechtigungszertifikate

vgl.	vergleiche
vif	virtual Interface
VT	VirtualisierungsTechnologie
VT-i	Intel Virtualization Technology for the Itanium architecture
VT-d	Intel Virtualization Technology for Directed I/O
VT-x	Intel Virtualization Technology for the IA-32 architecture
XML	eXtensible Markup Language
ZDA	Zertifizierungsdienstanbieter
z.B.	zum Beispiel
z.T.	zum Teil

1. Kurzdarstellung

Am 01.11.2010 wurde der neue Personalausweis zusammen mit der neuen AusweisApp und der elektronischen Identifikationsfunktion (eID-Funktion) eingeführt. Dies eröffnet ein breites Feld neuer Nutzungsmöglichkeiten. Bereits im Vorfeld der Neuerscheinung des Personalausweises wurde die Sicherheit der eID-Funktion diskutiert und Angriffe auf die Identität des Ausweisinhabers demonstriert. Im Rahmen dieser Arbeit wird die sichere Nutzung der (eID-Funktion) mit dem neuen Personalausweis unter Einsatz des Basis-Kartenlesers mit dem Ziel betrachtet, die Sicherheit zu verbessern. Im Zuge der Sicherheitsanalyse wird ersichtlich, dass das Sicherheitsniveau der neuen eID-Funktion unmittelbar vom Sicherheitsniveau des PCs abhängt und ein hoher Schutzbedarf für diese Funktion besteht. Es besteht ein nicht geringes Risiko, dass die Identität des Ausweisinhabers von einem Angreifer ausgenutzt wird. Dies kann z.B. durch den Einsatz eines Keyloggers und den Zugriff über den Browser auf den Personalausweis erfolgen. Zur Minderung des Risikos wird eine Verbesserung des Sicherheitsniveaus des PCs durch den Ansatz der Separierung, der nicht vertrauenswürdigen Anwendungen von den Anwendungen mit hohen Schutzanforderungen gewählt. Zusätzlich werden die Funktionen der AusweisApp entsprechend den Schutzanforderungen in zwei virtuelle Umgebungen geteilt und mit einem sicheren Kommunikationskanal verbunden. Dieses Lösungskonzept wird mit dem open source Betriebssystem: Qubes OS, das die Separierung durch Virtualisierung bietet, abgebildet. Zur Evaluierung wird ein Prototyp implementiert, der die identifizierten Risiken deutlich mindert und die Schutzanforderungen erfüllt.

2. Einleitung und Problemstellung

In diesem Kapitel wird die Aufgabenstellung erläutert und der Aufbau der Arbeit dargestellt.

2.1. Einführung des neuen Personalausweises

Im Rahmen des Durchlaufs vieler Geschäfts-, und Verwaltungsprozesse ist der Nachweis der Identität des Geschäftspartners notwendig. Dies erfolgt in der Regel anhand des Personalausweises. Ab dem 01.11.2010 wird der neue Personalausweis als Smartcard im Scheckkartenformat mit einer neuen elektronischen Identifikationsfunktion (eID-Funktion) ausgegeben. Diese wird z.T. die seither eingesetzte Authentisierung mit Benutzername und Passwort im Internet ersetzen [1].

Neben der Funktion des Personalausweises als Sichtausweis mit Lichtbild, werden gemäß der Technischen Richtlinie 3127 des Bundesamtes für Sicherheit in der Informationstechnik (BSI) nun folgende zusätzliche Funktionen auf der Smartcard zur Verfügung gestellt [2]:

- Auf Wunsch:
    - eID-Anwendung, (eID-Funktion): Online-Ausweisfunktion
    - Biometrie-Anwendung, (ePass): 2 Fingerabdrücke
- Auf Wunsch mit Zusatzkosten:
    - Signaturanwendung, (eSign): Qualifizierte elektronische Signatur (QES)

Im Rahmen dieser Arbeit wird nur die eID-Funktion betrachtet. Diese neue Funktion hat das Ziel, das Internet sicherer zu machen und rechtssichere elektronische Anwendungen zu ermöglichen. Es sind zahlreiche Anwendungsszenarien für den Einsatz des neuen elektronischen Personalausweises im Rahmen von Dienstleitungen im Internet denkbar. Schon heute können Behördengänge und Verwaltungsangelegenheiten, zum Beispiel die Zulassung eines Kraftfahrzeugs quasi „online“ über ein Internetportal erfolgen. Des Weiteren können z.B. eine Auskunft aus dem Verkehrszentralregister, die Schufa-Abfrage, der Zugang zu kommunalen Angeboten oder die Altersverifikation bei einem Online-Filmverleih durch geprüfte persönliche Daten und weitere Angebote online genutzt werden (vgl. [1]. [3]).

2.2. Problemstellung

Die nachfolgend aufgeführten Angriffsbeispiele und die Studie zeigen, dass das Sicherheitsniveau eines typischen PCs nicht ausreicht, um die eID-Funktion unter Einsatz des Basis-Kartenlesers ausreichend zu schützen.

Zur Nutzung des neuen elektronischen Personalausweiseses im Internet benötigt der Anwender neben dem Ausweis, einen Kartenleser für die Smartcard, sowie eine Software (AusweisApp), die die Kommunikation zwischen dem Ausweis und dem Computer ermöglicht. Es stehen grundsätzlich folgende 3 Arten Kartenlesern zur Verfügung [4]:

- Basis-Kartenleser
- Standard-Kartenleser
- Komfort-Kartenleser

Die elektronische Identifikationsfunktion wird durch die Eingabe einer persönlichen Identifikationsnummer (PIN) freigegeben. In der Basisversion des Kartenlesers besitzt dieser kein eigenes Tastenfeld zur Eingabe des PIN-Codes. Die PIN-Eingabe erfolgt in diesem Fall über die Tastatur des PCs. Die Eingabe der PIN über die Tastatur ist damit abhängig von den Sicherheitsvorkehrungen des Betriebssystems. Die Studie „Restrisiken beim Einsatz der AusweisApp“ der FH Gelsenkirchen belegt, dass Restrisiken beim Einsatz der AusweisApp mit Basis-Kartenleser bestehen und dass das Sicherheitsniveau des potenziell nicht vertrauenswürdigen PCs nicht ausreicht, um die sicherheitskritische AusweisApp mit der eID-Funktion ausreichend zu schützen [5].

Im Rahmen dieser Arbeit wird ausschließlich der Basis-Kartenleser betrachtet, der z.B. in der Computer-Bild -Ausgabe 26-2010- 400.000-fach als kostenlose Beigabe zur Zeitschrift vertrieben wurde. Computer Bild berichtet, dass Lesegeräte für den neuen Personalausweis im Wert von 24 Millionen Euro staatlich subventioniert wurden [6].

Bereits vor Veröffentlichung der AusweisApp, hat der Computer Chaos Club CCC in der Sendung vom 22. September 2010 “Bericht aus Brüssel” im WDR vor Angriffen auf die PIN mittels Computer Trojanern gewarnt [7]. Jan Schejbal zeigt ein Angriff auf die PIN im Zusammenspiel mit dem nPA auf dem Basiskartenleser, indem er mittels JavaScript die Dialoge der AusweisApp nachbaut. Erkennt der Nutzer den Unterschied nicht, erhält der Angreifer die Ausweis-PIN [8]. Er hat ein weiteren Angriff zur

Ausnutzung der fremden Identität demonstriert. Ein Angriff auf den nPA ist möglich, wenn das OWOK-Plugin im Browser installiert ist. Per JavaScript könnte dabei ein Kanal zur Chipkarte geöffnet werden und die Antworten des nPAs gelesen werden [9].

2.3. Szenario für die Nutzung der AusweisApp

Als Szenario für diese Arbeit dient die Nutzung der nicht hoheitlichen eID-Funktion mit dem Basiskartenleser im Internet zum Identitätsnachweis. Die Nutzung der hoheitlichen eID-Funktion durch berechtigte Behörden wird nicht betrachtet. Das Szenario verläuft wie folgt: Ein Ausweisinhaber möchte über eine Webanwendung eines Dienstanbieters eine Leistung in Anspruch nehmen. Voraussetzung dafür ist der sichere Nachweis der Identität des Interessenten. Der Ausweisinhaber nutzt den neuen Personalausweis mit dem Basis-Kartenleser, um seine Identität gegenüber dem Dienstanbieter nachzuweisen. Beispielsweise möchte ein Anwender über eine Online-Videothek Filme erwerben, die einer Altersbeschränkung (FSK18) unterliegen. Im Rahmen der Altersverifikation muss der Dienstanbieter, in unserem Szenario die Online-Videothek, sein Berechtigungszertifikat und damit auch die Daten, die er lesen darf, anzeigen. Anschließend muss der Anwender die sechsstellige persönliche Identifikationsnummer (PIN) über die Tastatur des Personal Computers (PCs) eingeben. Erst mit diesem Nachweis der Volljährigkeit durch den neuen Personalausweis werden die FSK18-Videos angezeigt.

Die eID-Funktion ist besonders schutzwürdig. In diesem Szenario werden Angreifer von außen, aus dem Internet betrachtet, die die Identität des Ausweisinhabers rauben wollen. Interne Angreifer, wie Familienmitglieder oder Angestellte/Mitarbeiter werden in diesem Szenario nicht betrachtet.

2.4. Zielsetzung der Arbeit

Diese Diplomarbeit basiert auf der Ausschreibung „Sichere Nutzung des nPA-Bürgerclients“ des Fraunhofer Instituts für Sichere Informationstechnologie (siehe Materialanhang CD).

Ziel dieser Arbeit ist, das Sicherheitsniveau der AusweisApp bei Nutzung der eID-Funktion unter Einsatz des Basis-Kartenlesers zu verbessern. Im Zuge dieser Arbeit wird eine Sicherheitsanalyse für das gewählte Szenario durchgeführt, um das Risiko,

dass die Identität des Ausweisinhabers geraubt und ausgenutzt wird einzuschätzen. Basierend auf den Angriffsvektoren aus der Risikoanalyse wird ein Konzept zur Verbesserung des Sicherheitsniveaus der eID-Funktion erarbeitet. Das Konzept wird mit einem Prototyp umgesetzt. Anhand des Prototyps soll gezeigt werden, dass mit Anwendung des Konzeptes die Angriffsvektoren neutralisiert werden können. Des Weiteren wird eine Abschätzung des Sicherheitsgewinns durch die Anwendung des Konzeptes erfolgen.

Rahmenbedingungen für die Konzepterstellung sind:

- Die Verwendung des Basiskartenlesers
- Angriffsabwehr mittels Separierung der Prozesse zur PIN-Eingabe
- Die Anwendung des Betriebssystems „Qubes OS“

2.5. Vorgehen

Gemäß den Ausführungen von Eckert umfasst das Security Engineering die Tätigkeiten zur Konstruktion sicherer Systeme im Sinne der IT-Sicherheit. Um den Ablauf dieser Tätigkeiten zu beschreiben, werden zumeist Vorgehensmodelle aus dem Bereich der Softwareentwicklung, so zum Beispiel das Wasserfallmodell bzw. das V-Modell oder das Spiralmodell angewendet. Anders als der Bereich des Softwareengineerings, der sich mit der ingenieurmäßigen Entwicklung von Software befasst, sind die Maßnahmen und Methoden, die dem Bereich des Security Engineering zugeordnet werden, laut Eckert noch nicht ausreichend „methodisch ausgearbeitet“ [10].

Das Bundesamt für Sicherheit in der Informationstechnik stellt im Rahmen der IT-Grundschutz Kataloge den BSI-Sicherheitsprozess für die Weiterentwicklung der IT-Infrastruktur in Unternehmen zur Verfügung [11]. Die Entwicklungsphasen sind für die Entwicklung und Umsetzung sicherer IT-Systeme in Unternehmen gedacht. Die Phasen des Entwicklungskonzepts gliedern sich ausgehend aus dem Einsatzgebiet und den funktionalen Anforderungen in:

1. Initiierung des IT-Sicherheitsprozesses
2. IT-Strukturanalyse
3. Schutzbedarfsfeststellung
    a. Falls niedriger, mittlerer Schutzbedarf in
        i. IT-Grundschutzanalyse

b. Falls hoher Schutzbedarf in
    i. Bedrohungsanalyse
    ii. Risikoanalyse
    iii. Maßnahmen
4. Realisierungsplanung
5. Umsetzung
6. Aufrechterhaltung im laufenden Betrieb

Dieser Entwicklungsprozess, zur Konstruktion sicherer IT-Systeme in Unternehmen, muss für die Bearbeitung der Aufgabenstellung dieser Arbeit angepasst werden. Die Tätigkeiten des Security-Engineerings umfassen üblicherweise eine Bestimmung der Schutzziele, eine Strukturanalyse, eine Schutzbedarfsermittlung, eine Bedrohungsanalyse, eine Risikobewertung, je einen Schritt für den Schutzentwurf und dessen Realisierung sowie eine Evaluierung des erreichten Schutzniveaus [10]. Die folgende Abbildung stellt die Tätigkeiten in der Reihenfolge der Bearbeitung - im Rahmen dieser Arbeit - in Form eines Wasserfallmodells dar:

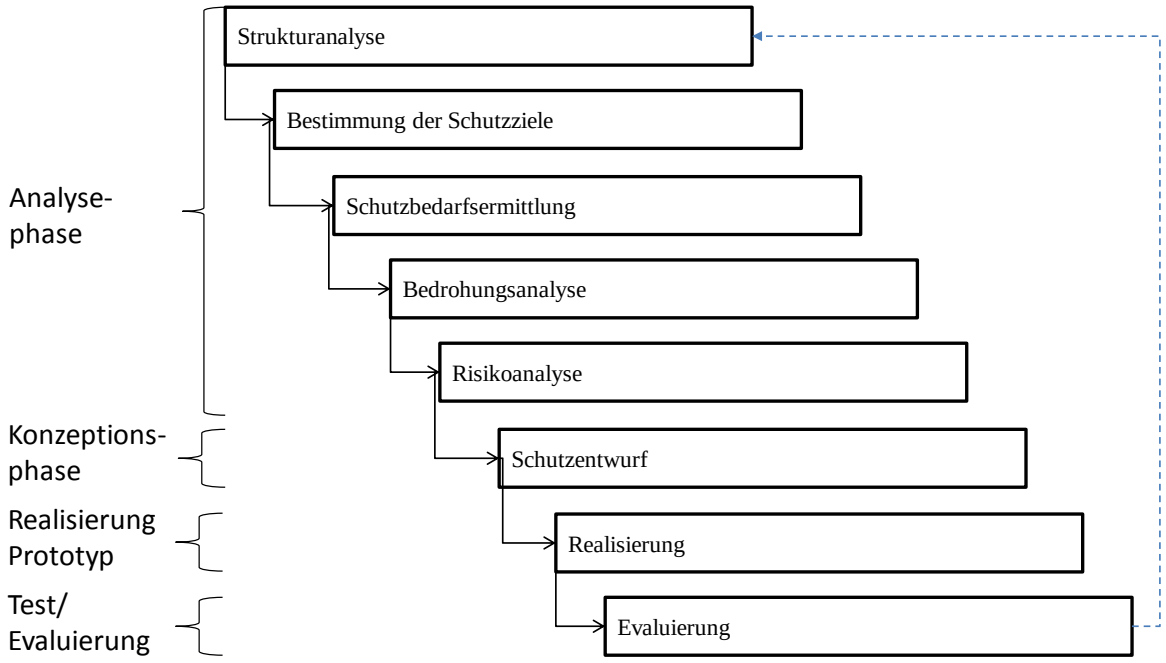

Abbildung 1: Tätigkeiten Security-Engineering im Wasserfallmodell [10]

Die vorliegende Arbeit ist in 9 Kapitel untergliedert. Kapitel 3 enthält die notwendigen Grundlagen zur IT-Sicherheit. Kapitel 4 umfasst die Strukturanalyse und die Sicherheitsanalyse der eID-Funktion. Im Rahmen der Strukturanalyse werden die Komponenten sowie deren Kommunikation untereinander und der Ablauf der eID-Funktion, inklusive der angewendeten Sicherheitskonzepte, analysiert. Im Zuge der Sicherheitsanalyse werden zunächst die Schutzziele und der Schutzbedarf der eID-Funktion erarbeitet. Anschließend werden die Gefährdungen für die eID-Funktion systematisiert. Die erfassten Bedrohungen zum Angriffsziel: „Rauben der Identität des Ausweisinhabers“ werden im Rahmen der Risikoanalyse bewertet. Das Kapitel schließt mit der Zusammenfassung der Analyseergebnisse und der Zusammenstellung der Anforderung an das Lösungskonzept. Im fünften Kapitel werden Lösungsalternativen zur Verbesserung des Sicherheitsniveaus des PCs erarbeitet und bewertet. Die Auswahl der geeignetsten Alternative wird mit einer Nutzwertanalyse durchgeführt. Abschließend wird in diesem Kapitel die gewählte Lösungsalternative ausgearbeitet. Im sechsten Kapitel wird das bestehende Konzept mit dem Betriebssystem Qubes OS umgesetzt. Hierzu werden die Sicherheitsmerkmale von Qubes OS beschrieben und anschließend auf das bestehende Konzept abgebildet. In Kapitel 7 wird das Konzept im Rahmen eines Prototyps implementiert. Anhand des Prototyps wird im achten Kapitel überprüft, inwieweit das Lösungskonzept die definierten Anforderungen erfüllt. Anschließend wird erneut eine Risikoanalyse durchgeführt und der Sicherheitsgewinn bewertet. Das Kapitel schließt mit der Darstellung, was im Rahmen der Arbeit nicht betrachtet wurde. Kapitel 9 enthält die abschließende Bewertung und einen Ausblick.

3. Grundlagen zur IT-Sicherheit

In diesem Teil der Arbeit soll ein Überblick zur IT-Sicherheit gegeben werden und eine Einordnung der Problemstellung dieser Arbeit erfolgen.

Eckert schreibt, dass die Sicherheit der informationstechnischen Systeme gewährleistet ist, wenn sie funktionssicher und informationssicher sind. Unter Funktionssicherheit versteht Frau Eckert, dass die zugesicherten Funktionen unter normalen Betriebsbedingungen ausführbar sind. Informationssicherheit bedeutet, dass nur solche Systemzustände angenommen werden, die zu keiner unautorisierten Informationsveränderung oder –gewinnung führen [12].

3.1. Grundwerte bezüglich Sicherheit von IT-Systemen

Aus der Definition der IT-Sicherheit können entsprechende Schutzziele, wie z.B. Vertraulichkeit, abgeleitet werden. Das Bundesamt für Sicherheit in der Informationstechnik benennt nach der „IT-Grundschutz Vorgehensweise“ die folgenden Grundwerte der Informationssicherheit [13]:

1. Vertraulichkeit (Geheimhaltung von Daten)
2. Integrität (Nachweis, dass Daten nicht manipuliert werden)
3. Authentifizierung (Nachweis der Identität der Person bzw. der Anwendung)
4. Verfügbarkeit (Funktion steht zur Verfügung)

3.2. Angriffs- und Angreifer-Typen

Unter einem Angriff wird nach Eckert, einen nicht autorisierten Zugriff bzw. einen nicht autorisierten Zugriffsversuch auf das System verstanden. Sie unterscheidet dabei aktive und passive Angriffe. Beispiele für passive Angriffe sind das Abhören von Datenleitungen (englisch: Eavesdropping) oder das unautorisierte Lesen von Daten. Aktive Angriffe betreffen die unautorisierte Modifikation von Datenobjekten [14].

Folgende Angriffstypen werden in der Studie des Fraunhofer-Instituts für sichere Informationstechnologie (Fraunhofer-SIT) bei Anwendung eines kontaktlosen Chips beschrieben [15]:

- Sniffing/Eavesdropping (Abhören der Kommunikation)
- Skimming (Unberechtigter Auslesen)

- Spoofing und Replay-Attacken (Abhören der Kommunikation und unberechtigtes Manipulieren)
- Man-in-the-Middle (Zwischenschalten in eine laufende Kommunikation)
- Tracking (Profilbildung)
- Cloning & Emulation (nachbauen und duplizieren eines RFID-Chips)
- Denial of Service (Dienstblockade)
- Relay-Angriffe (Vortäuschung der physikalischen Präsenz des RFID-Chips)
- RFID-Malware (mittels Buffer-Overflow kann der Programmcode des RFID-Chips manipuliert werden)

Angreifer sind z.B. Hacker oder Cracker. Diese suchen Schwachstellen und entwickeln Angriffe, sogenannte Exploits, um diese Schwachstellen auszunutzen. Hacker wollen damit in der Regel keinen persönlichen Vorteil erlangen und veröffentlichen diese Exploits. Ein Cracker dagegen nutzt den Exploit zu seinem persönlichen Vorteil aus. Cracker gehören heute laut dem Lagebericht zur IT-Sicherheit 2011 des Bundesamtes für Sicherheit in der Informationstechnik meist in den Bereich der organisierten Kriminalität. Entsprechend dem Lagebericht, kennzeichnet die Angreifer in der Internetkriminalität eine zunehmende Kommerzialisierung und Professionalisierung [16].

3.3. Bedrohungen

Nach Ansicht von Eckert zielt eine Bedrohung darauf ab, eine Schwachstelle, über die die Sicherheitsdienste des Systems umgangen, getäuscht oder unautorisiert modifiziert werden können, auszunutzen [17].

Eine Klassifikation solcher Gefährdungsfaktoren, wie sie beispielsweise in den BSI-Lehrbriefen verwendet wird, zeigt Tabelle 1. Im Rahmen dieser Arbeit wird im Wesentlichen die Gefährdung durch organisatorische Mängel und Vorsatz betrachtet.

Höhere Gewalt	**Fahrlässigkeit**	**Technisches Versagen**	**Vorsatz**	**Organisatorische Mängel**
Blitzschlag	Irrtum	Stromausfall	*Manipulation*	*unberechtigter Zugriff*

Feuer	Fehlbedienung	Hardware-Ausfall	Einbruch	Raubkopie
Überschwemmung	unsachgemäße Behandlung	Fehlfunktionen	*Hacking*	ungeschultes Personal
Erdbeben			Vandalismus	
Demonstration			Spionage	
Streik			Sabotage	

Tabelle 1: Klassifikation von Gefährdungsfaktoren [18]

3.4. Risiko

In der Risikoanalyse erfolgt die Bewertung der Bedrohungen, durch die Abschätzung der Wahrscheinlichkeiten für das Eintreten der Bedrohungen sowie mit der Abschätzung des Schadenspotenzials. Laut Eckert hängen Schwachstellen, Bedrohungen und das Risiko wie folgt voneinander ab:

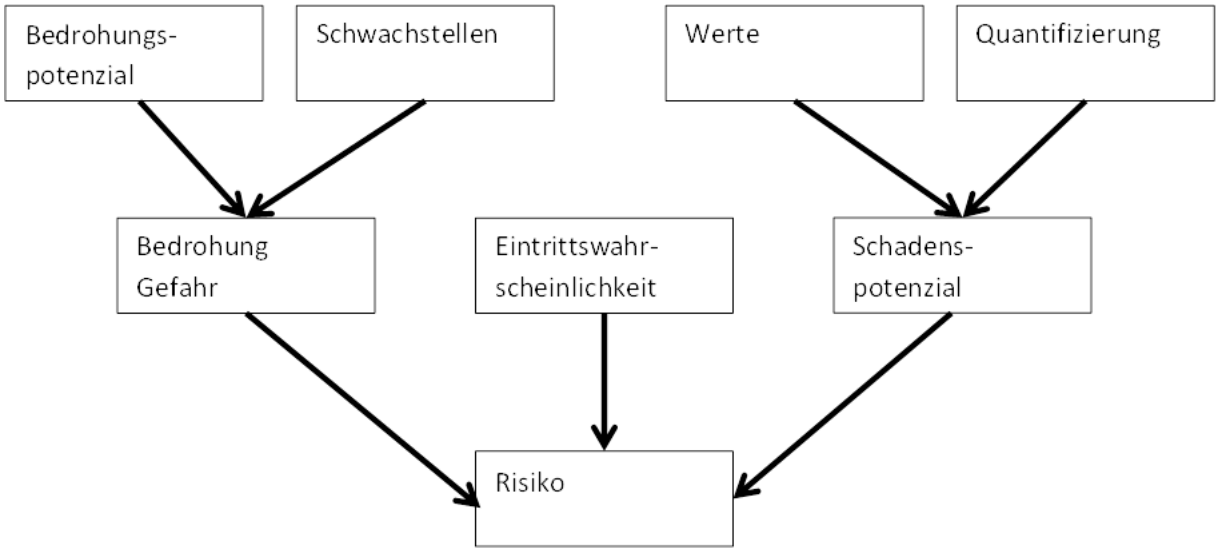

Abbildung 2: Zusammenhang zwischen Schwachstellen, Bedrohungen, Risiken [75]

4. Analyse

In diesem Kapitel erfolgt die Analyse der Komponenten, der nicht hoheitlichen Online-Ausweisfunktion, deren Funktionen, inklusive der Sicherheitsmerkmale. Darauf aufbauend wird eine Sicherheitsanalyse durchgeführt. Diese besteht aus der Schutzbedarfs-, und Bedrohungsanalyse, sowie der Risikoanalyse. Das Kapitel schließt mit der Zusammenfassung der Analyseergebnisse und den für die Konzeption abgeleiteten Anforderungen.

4.1. Strukturanalyse des elektronischen Identitätsnachweises (eID)

In diesem Abschnitt werden die funktionalen Eigenschaften, die Einsatzumgebung und der Verwendungszweck analysiert. Dazu werden die benötigten Systemkomponenten und –dienste sowie deren Funktionalität und die Sicherheitsmerkmale untersucht.

Nach § 1 Personalausweisgesetz (PAuswG) besteht für meldepflichtige Personen, die über 16 Jahre sind, eine Ausweispflicht. In § 2 PAuswG ist geregelt, dass berechtigte Behörden als hoheitliche Maßnahme zur Erfüllung ihrer gesetzlichen Aufgaben die Identität von Personen anhand des Personalausweises feststellen dürfen (hoheitliche Ausweisfunktion). Ebenso ist hier geregelt, dass berechtigte Dienstanbieter zur Erfüllung eigener Geschäftszwecke Zugriff auf Identitätsmerkmale des Ausweisinhabers haben dürfen (nicht-hoheitliche Ausweisfunktion) [19].

4.1.1. Der Prozess der elektronischen Identifikation im Internet

Laut der Broschüre „eID-Service Pocketguide 2011“ läuft beispielhaft der nicht-hoheitliche elektronische Identifikationsnachweis im Rahmen des Besuches eines Dienstanbieters (Online-Shop im Internet) wie folgt ab [20]:

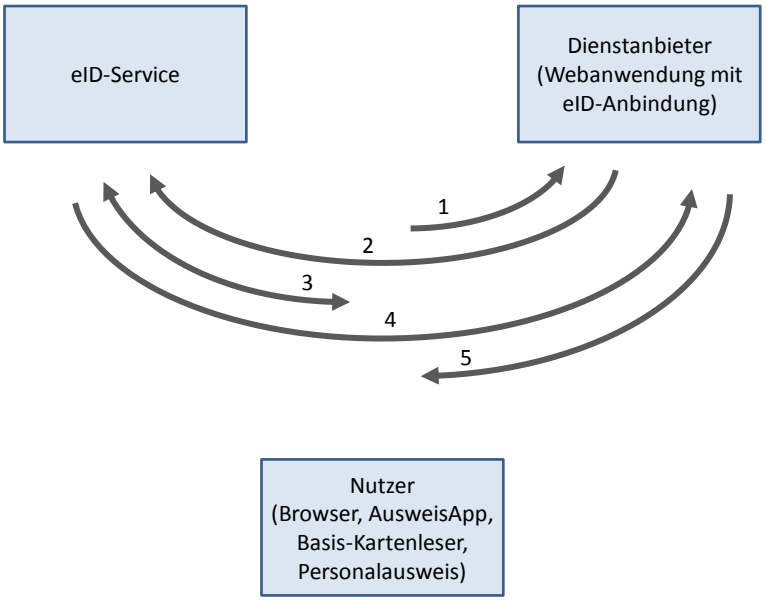

Abbildung 3: Ablauf eID-Service [20]

1. Ein Nutzer möchte beispielsweise ein Produkt bei einem Online-Shop im Internet erwerben und mittels des elektronischen Identitätsnachweises seine Identität nachweisen. Er sendet eine entsprechende Anfrage an den Online-Shop.
2. Um die Identität des Käufers zu authentifizieren, wird die Anfrage an den eID-Service weitergeleitet.
3. Der eID-Service übermittelt dem Nutzer das Berechtigungszertifikat des Dienstanbieters. Darin enthalten sind die Berechtigungen des Online-Shops auf bestimmte Daten aus dem neuen Personalausweis (nPA). Dem Nutzer wird die Auswahl der zu übermittelnden Daten angezeigt. Er schränkt diese gegebenenfalls ein und gibt seine persönliche eID- PIN über die Tastatur ein. Über eine gesicherte Verbindung werden die Daten aus dem Ausweis ausgelesen.
4. Der eID-Service übermittelt die personenbezogenen Daten an den Online-Shop.
5. Der Online-Shop bestätigt die Anfrage des Nutzers und leitet die weiteren Schritte, so zum Beispiel den Versand der Ware ein.

Zur Durchführung dieses Prozesses sind viele Beteiligte in unterschiedlichen Rollen notwendig. In den Rollen werden jeweils Aufgaben einer Person bzw. einer

Personengruppe zusammengefasst. Tabelle 2 gibt einen Überblick über die Rollen der Beteiligten bei Nutzung der eID-Funktion mit dem neuen Personalausweis:

Objekt/ Subjekt	**Inhaltliche und/oder organisatorische Aufgabe**
Bürger	• Erhält Personalausweis von der Personalausweisbehörde • Nutzt Dienstleistungen eines Dienstanbieters, die eine Authentifizierung erfordern • Identifiziert sich gegenüber dem Dienstanbieter mit dem nPA
Dienstanbieter	• Stellt Dienste, die eine sichere Authentifizierung benötigen, zum Beispiel Online-Filmeverleih, bereit • Bezieht Berechtigungszertifikat von der Vergabestelle für Berechtigungszertifikate (VfB) • Beauftragt einen eID-Serviceanbieter oder betreibt diesen Service selbst
Vergabestelle für Berechtigungszertifikate (VfB), Bundesverwaltungsamt	• Nimmt Anträge der Dienstanbieter auf Berechtigungszertifikate entgegen (Registration Authority) • Prüfung der Anträge und Erteilung der Berechtigung für Berechtigungszertifikate
Personalausweisbehörde	• Bezieht Personalausweis vom Hersteller • Nimmt Anträge zum nPA vom Bürger entgegen • Übergabe des nPA an Bürger und ggf. Aktivierung der eID-Funktion sowie ggf. Änderung der Daten auf dem nPA
Hersteller Personalausweis (Bundesdruckerei)	• Stellt Personalausweise her • Initialisiert und personalisiert den nPA • Versenden des Briefes mit PIN, PUK und Sperrkennwort
Zertifizierungsdienstanbieter,	• Erstellt die technischen Berechtigungszertifikate

Sperrlistenbetreiber (z.B. D-Trust, Trustcenter der Bundesdruckerei)	(Terminalzertifikate) auf Basis der Berechtigungen der Dienstanbieter der Vergabestelle für Berechtigungszertifikate • Erstellt und aktualisiert Sperrlisten und stellt diese den Dienstanbietern und eID-Service-Betreibern zur Verfügung
eID-Servicebetreiber	• Stellt Hard- und Softwarekomponenten (eID-Server) zur Verfügung • Bezieht neue Berechtigungszertifikate und aktualisierte Sperrlisten
Bundesamt für Sicherheit in der Informationstechnik	• Root-Certification Authority (CVCA) • Stellt Spezifikationen, Protokolle sowie Technische Richtlinien zur Verfügung • Liefert globale Sperrliste an die Zertifizierungsdienstanbieter aus

Tabelle 2: Rollenmodell zur eID-Funktion des nPAs [21] [1] [22]

Im Rahmen des eID-Prozesses können vom neuen Personalausweis folgende Informationen bereit gestellt werden [1]:

- Familienname
- Vorname(n)
- Akademischer Titel
- Künstlername
- Geburtsdatum
- Geburtsname
- Geburtsort
- Anschrift
- Ausgebender Staat
- Angabe, dass es sich um einen Personalausweis handelt (Dokumententyp)
- Ablaufdatum

Zusätzlich können folgende Funktionen ausgeführt werden [1]

- Gültigkeitsprüfung

- Altersverifikation
- Wohnortverifikation
- Pseudonym-Funktion
- Berechnung des Sperrmerkmals

4.1.2. Kommunikation zwischen den Komponenten

Abbildung 4 zeigt entsprechend der Technischen Richtlinie 3130 eine Integrationsmöglichkeit des eID-Servers in die eID-Anwendung [23]. Sie stellt die notwendigen Komponenten dar, die im Anschluss erläutert werden:

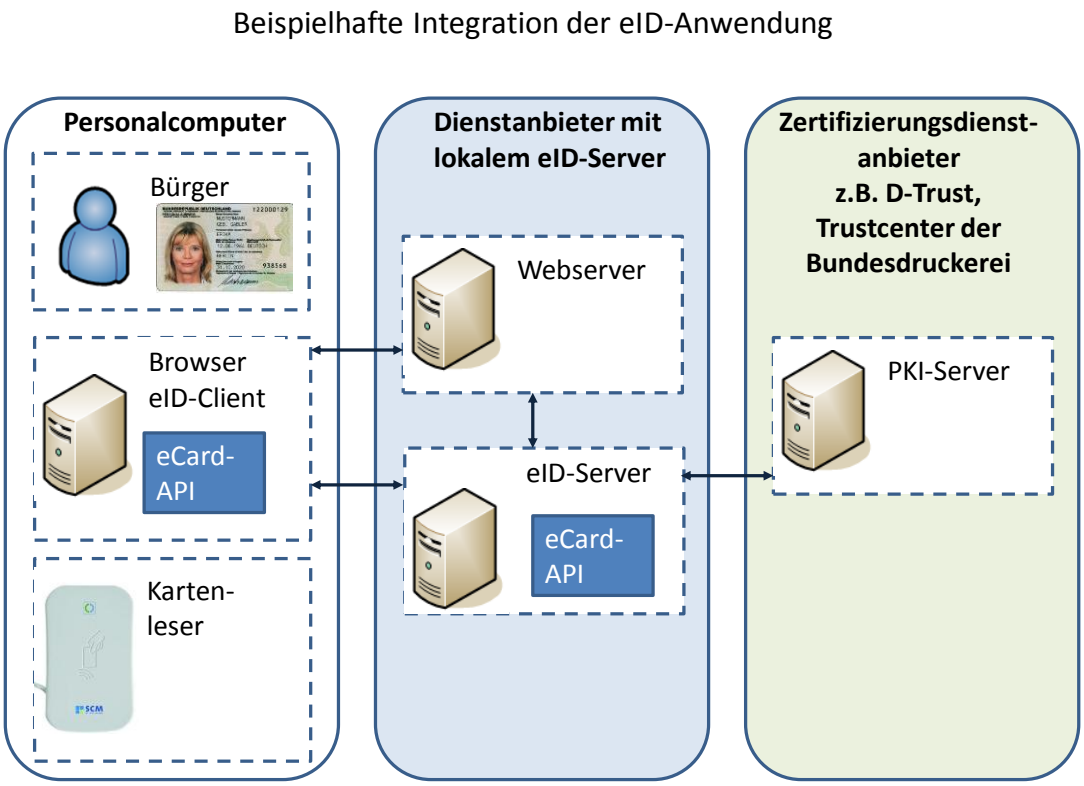

Abbildung 4: Komponenten zur Nutzung der eID-Funktion [23]

Der Bürger setzt zur Nutzung der eID-Funktion den neuen Personalausweis mit dem kontaktlosen RFID-Chip ein. Die Radio Frequency Identification (RFID) wurde entwickelt, um eine eindeutige Kennzeichnung von Objekten bzw. Subjekten durch elektronisch gespeicherte Daten zu ermöglichen. Ein RFID-System besteht technologisch betrachtet aus zwei Komponenten, einem Transponder und einem Lesegerät. Die Kommunikation erfolgt entsprechend dem ISO 14443-Standard drahtlos

über einen Funkfrequenzkanal (13,56 MHz) [24]. Die Datenübertragung mit einer Reichweite von 10 bis 15 cm erfolgt durch Lastmodulation auf Abruf [25]. Der nPa unterstützt den ISO 14443-Standard und enthält gemäß TR-03127 auf dem Chip einen kryptographisch starken Zufallszahlengenerator und unterstützt Kryptographie mittels elliptischer Kurven gemäß TR-03111 [26].

Als Lesegerät können, entsprechend der Technischen Richtlinie TR-03119, 3 Klassen von Kartenlesern zur eID-Funktion eingesetzt werden [4]:

1. Basis-Chipkartenleser
2. Standard-Chipkartenleser
3. Komfort-Chipkartenleser

Der im Rahmen dieser Arbeit betrachtete Basis-Chipkartenleser verfügt über eine Schnittstelle zum Host-Rechner (USB) und die oben beschriebene kontaktlose Schnittstelle nach ISO 14443. Der Basis-Kartenleser unterscheidet sich vom Standard-Kartenleser durch das fehlende PIN-PAD mit sicherer PACE-Unterstützung und die fehlende Funktion „Firmwareupdate“. Der Komfort-Kartenleser bietet darüber hinaus noch weitere Funktionen - wie z.B. ein Display - und ist zwingend vorgeschrieben für die Nutzung der qualifizierten elektronischen Signatur (eSign-Funktion).

Auf dem Personal Computer des Bürgers ist entsprechend der Spezifikation TR-03127, die AusweisApp bestehend aus dem Browser-Plugin und dem eID-Client mit der eCard-API notwendig [2]. Die AusweisApp bildet den Client und steht laut BSI für die Betriebssysteme Windows, Debian, Ubuntu und OpenSUSE zur Verfügung (Stand 17.12.2011 [27]). Sofern ein Dienst über den Browser vom Webserver aufgerufen wird, kann dieser die Funktion des eID-Servers nutzen und die personenbezogenen Daten erhalten. Durch die AusweisApp kann der Nutzer auswählen, welche Datengruppen aus seinem nPA ausgelesen und auf welchen Datengruppen Funktionen (z.B. Altersverifikation) ausgeführt werden dürfen. Beim Aufruf der eID-Funktion über einen Button auf den Seiten des Betreibers, sendet dessen Webserver die Authentifizierungsanfrage an den eID-Server weiter. Der eID-Server sendet ein spezielles HTML-Object mit MIME-Type (application/vnd.ecard-client) unter anderem mit Parametern zum PAOS-Protokoll, der Server-Adresse und dem Session-Identifier an das Plugin des Client-Browsers. Der MIME-Type-Handler des Browser-Plugin liest die Parameter aus und initialisiert den eID-Client [28]. Das eCard-API-Framework,

spezifiziert in den Technischen Richtlinien TR-03112, bildet eine plattformunabhängige Schnittstellendefinition, die zwischen Smartcards und den möglichen Anwendungen auf Basis der eCard-Strategie der Bundesrepublik Deutschland aufsetzen [29]. Die eCard-API nutzt den ISO 24727-Standard für die Interaktion von externen Applikationen mit auf ISO 7816-4 basierenden Chipkarten [30] [31]. Schnittstellen nach diesem Standard stellen generische Dienste bereit und sind unabhängig von den technischen Eigenschaften der Chipkarten. Der eCardInitiator startet die AusweisApp und initiiert den Verbindungsaufbau zwischen dem eID-Service und dem Chip auf dem nPA.

Die Technische Richtlinie eID-Server TR-03130 gibt vor, dass der eID-Server die Authentisierungszertifikate speichert und verwaltet. Der Server stellt die kryptografischen Protokolle bereit. Er kommuniziert über folgende drei Schnittstellen:

1. eID-Schnittstelle
2. eCard-API-Schnittstelle
3. PKI-Schnittstelle

Zusätzlich zu den Kommunikationsschnittstellen wird die Administrative Management-Schnittstelle für die Konfiguration des eID-Servers benötigt, um initiale Einstellungen und Schlüssel zu erhalten. Die Technische Richtlinie TR-03130 spezifiziert die Außenkommunikation des eID-Servers über die eID-Schnittstelle mit der Web-Anwendung. Die vorrangige Aufgabe der eID-Schnittstelle ist die gegenseitige Authentisierung, die Anforderung von Daten aus dem nPA des Anwenders sowie deren Bereitstellung für die Web-Anwendung. Laut TR-03130 besteht die eCard-API aus zwei Komponenten, dem Client und dem Server. Der Client, auch AusweisApp genannt, reagiert auf Anfragen, die er durch den Browser des Nutzers erhält. Der Client verbindet sich daraufhin mit dem eCard-API Server. Mit dieser Verbindung liest der eID-Server die Daten aus dem nPA aus [23]. Laut der TR-03127 erfolgt über die PKI-Schnittstelle der automatisierte Abruf der Terminal-Berechtigungszertifikate sowie der Abruf der Sperrlisten. Die hierzu notwendige Public Key Infrastructure (PKI) besteht aus der Verwaltung und Bereitstellung aller für die Abwicklung des EAC-Protokolls notwendigen Zertifikate sowie die Sperrlisten der Dienstanbieter. [2]. Entsprechend der Technischen Richtlinie TR-03128 hat die eID-PKI folgende Grundstruktur [21]:

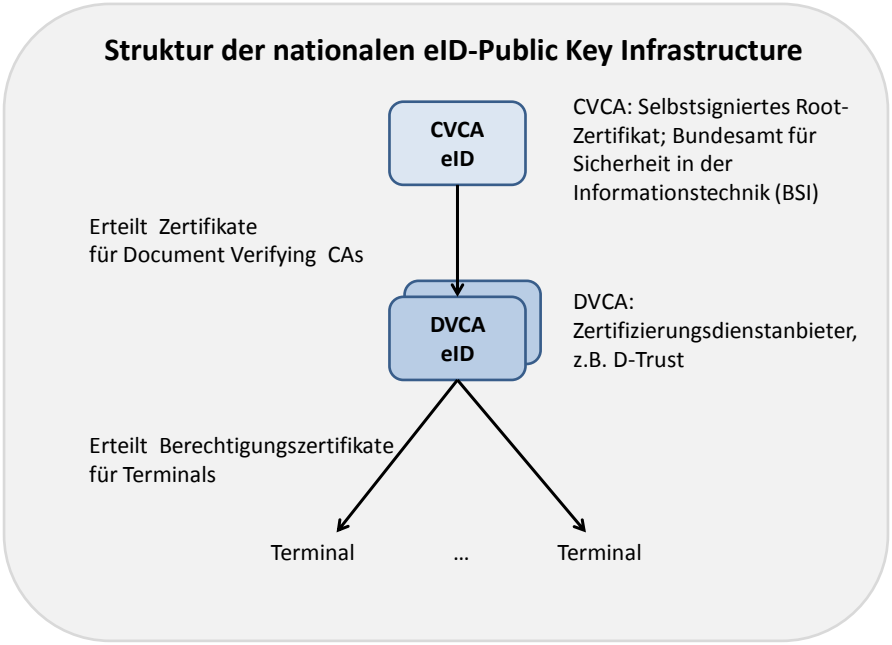

Abbildung 5: Anwendungsorientierte Grundstruktur der eID-PKI [21]

Die Country Verifying Certificate Authority (CVCA) für eID wird vom BSI betrieben. Das BSI erstellt die deutschen Wurzelzertifikate, mit deren privaten Schlüsseln die Document Verifier-Zertifikate signiert werden. Die Document Verifier erstellen die Terminal-Berechtigungszertifikate. Diese erhält ein Dienstanbieter nur, wenn er gegenüber der Zertifizierungsinstanz, der Vergabestelle für Berechtigungszertifikate (VfB) ein berechtigtes Interesse nachweisen kann [21]. Mit der Berechtigung erhält der Dienstanbieter eine Leseberechtigung für bestimmte Datenfelder des nPA. Die Berechtigungszertifikate haben in der Regel aus Sicherheitsgründen nur eine kurze Laufzeit.

Die einzelnen Komponenten kommunizieren dabei über unterschiedliche Protokolle, die in den Technischen Richtlinien TR-03127, TR-03128 und TR-03130 festgelegt sind, miteinander. Die Kommunikationskanäle werden in der folgenden Übersicht dargestellt und entsprechend der Nummerierung wie folgt beschrieben:

Übersicht der Kommunikationskanäle

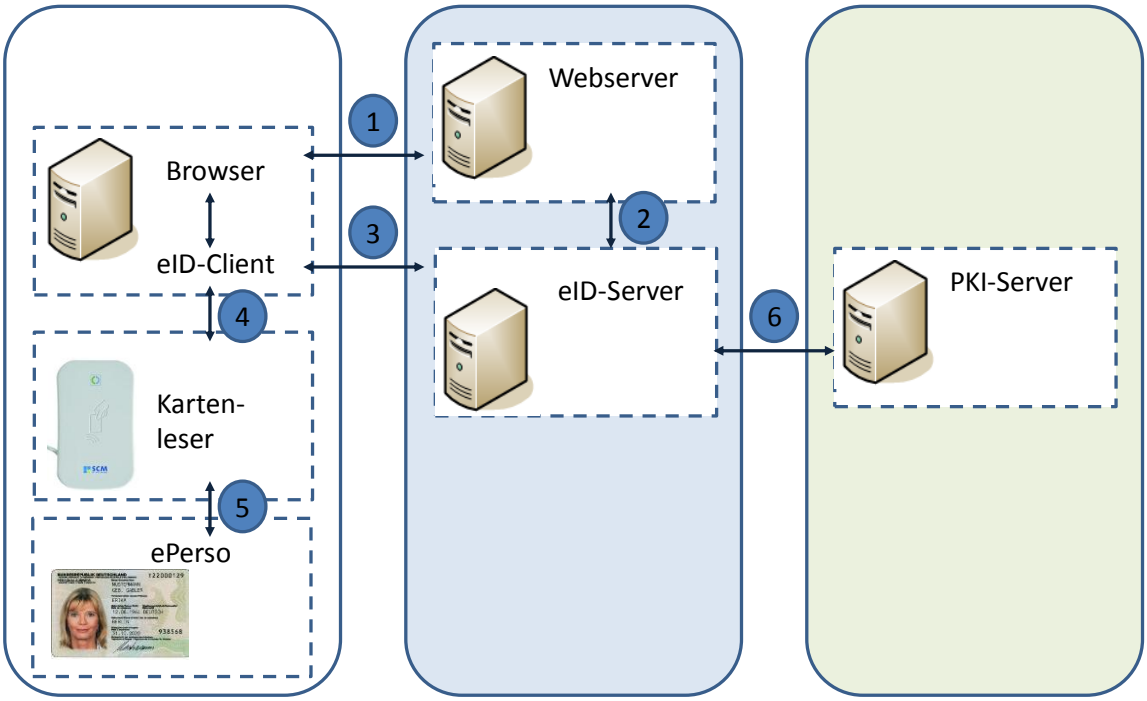

Abbildung 6: Übersicht der Kommunikationskanäle [2] [21] [23]

Zu 1: Die Kommunikation geht vom Client aus. Dieser baut eine Verbindung zum Web-Server auf. Der Web-Server beantwortet die Anfrage. Diese Kommunikation mit Nutzern wird mit SSL-Zertifikaten verschlüsselt. Im Kommunikationskanal Web-Server zu Web-Browser sowie Start der AusweisApp durch das Browser Plugin wird auf Basis des http-Protokolls der Secure-Socket-Layer (SSL) laut RFC 6101 verwendet: HyperText Transfer Protocol Secure (HTTPS) [32].

Zu 2: An dieser Stelle kann gemäß TR 3130 zwischen zwei Protokollen gewählt werden [23]: Es kann entweder Simple Object Access Protocol (SOAP) mit Pre-shared-Key oder die Security Assertion Markup Language 2.0 (SAML 2.0) mit Pre-shared-Key eingesetzt werden. Das SOAP-Protokoll ist ein XML-basiertes (eXtensible Markup Language) Kommunikations-Protokoll. Das Protokoll dient dem Austausch von strukturierten Informationen in verteilten Systemen [33]. Das SAML-Protokoll ist ein Standard des OASIS-Konsortiums und spezifiziert laut SAML 2.0 ein XML-Framework, für den sicheren Austausch von Authentifizierungen und Autorisierungen zwischen Sicherheitsdomänen [34]. SAML Assertions sind Aussagen, anhand derer ein Dienstanbieter (Service-Provider) mittels eID-Service Identitätsdaten eines eID-Nutzers

erhält. Der SAML-Token beinhaltet die Informationen vom Ausweis und wird dem Dienstanbieter zur Weiternutzung zur Verfügung gestellt. Die SAML-Kernspezifikation beschreibt den Aufbau der Assertion und der Protokollnachrichten. SAML Protokollnachrichten werden in HTTP-redirect-Nachrichten versendet. Die SAML Bindings spezifizieren, wie die verschiedenen SAML Protokollnachrichten auf darunter liegende Transportprotokolle abgebildet werden.

Zu 3: Im Kommunikationskanal zwischen dem eID-Client und dem eID-Server kommt auf Basis des http-Protokolls die Transport-Layer-Security (TLS), unter Verwendung von Pre shared Key-Verschlüsselungen zum Einsatz. Auf Basis der Pre-shared Key Ciphersuites for TLS (RFC 4279) erfolgt die Authentifizierung. Damit können die zusammengehörenden Prozesse zwischen den Komponenten (Kanalverschränkung der Kommunikation von Web-Browser mit Web-Server und von eID-Client mit eID-Server) identifiziert werden [35]. Innerhalb des eID-Funktionsablaufs ist es notwendig, dass der eID-Client Services für den eID-Server anbieten kann. Dies wird auf Basis des PAOS-Protokolls durchgeführt. Entsprechend dem Standard „Reverse HTTP Binding for SOAP“ definiert Reverse SOAP Binding einen mehrphasigen Nachrichtenaustausch. Der SOAP-Header und dessen Anbindung an HTTP sind so spezifiziert, dass der Client unter Verwendung von SOAP Services anbieten kann. Es handelt sich hierbei um ein „Reverse HTTP Binding for SOAP“, der es einem http-Client (AusweisApp) ermöglicht auf SOAP Requests des eID-Servers zu antworten. Daher sind die Buchstaben der Abkürzung auch in umgekehrter Reihenfolge genannt [36].

Zu 4 und 5: Das Application Protocol Data Units (APDU) ist eine Protokoll-Dateneinheit der Anwendungsschicht zwischen eID-Anwendung und dem Chip des neuen Personalausweises nach dem ISO 7816-4 Standard [31]. Es ist dadurch gekennzeichnet, dass jedes Kartenkommando Angaben zu Chaining, Kanalnummer, Bezeichnung der Kommandos sowie Kodierung der Länge der gelieferten Daten und der zu erwartenden Antwortdaten enthält.

Zu 6: Im Kommunikationskanal zwischen dem eID-Server und dem PKI-Server werden Web-Services - auf Basis des HTTP-Protokolls - unter Verwendung von z.B. der Transport-Layer-Security (TLS) eingesetzt [37].

4.1.3. Sicherheitsmerkmale der eID-Funktion

In der Kommunikation zwischen den Komponenten werden laut der Technischen Richtlinie TR-03110 des BSI nachfolgende Sicherheitsprotokolle verwendet [22]:

Das **Password Authenticated Connection Protocol (PACE)** sichert die Verbindung von Kartenleser bzw. eID-Client zu nPA. Gleichzeitig wird überprüft, ob der Benutzer die auf dem Chip hinterlegte PIN eingegeben hat. Das PACE-Protokoll dient dem gleichzeitigen Nachweis, dass sich Chip und Kartenleser bzw. eID-Client im Besitz des gleichen Passwortes befinden. Die PIN-Prüfung erfolgt ohne Übertragung der PIN. Es ist ein kryptografisches Protokoll, das im Kern aus einem doppelten Diffie-Hellmann Schlüsselaustauschprotokoll besteht. Jeder Kommunikationspartner vereinbart dezentral ein Schlüsselpaar bestehend aus einem geheimen Schlüssel und einem öffentlich bekannten Schlüssel. Der eID-Client erhält die Kurvenparameter (Elliptic Curve Diffie-Hellmann (ECDH), Elliptic Curve Digital Signature Algorithmus (ECDSA)) vom Chip des neuen Personalausweises und liefert die Referenzen zur PIN und die Zertifikatkette. Daraus werden durch eID-Client und nPA die gemeinsamen Schlüssel mittels Gleitpunktableitung aus der Kurve berechnet. Das Ergebnis nach Durchführung von PACE ist der Aufbau eines verschlüsselten und integritätsgesicherten Kommunikationskanals zwischen Personalausweis und Kartenleser bzw. eID-Client.

Das **Extended Access Control (EAC)** bietet eine erweiterte Zugriffskontrolle auf die Daten des nPA. EAC umfasst die Subprotokolle Chip Authentication (CA) und Terminal Authentication (TA) in der Verbindung von eID-Service zu nPA. Das Ergebnis nach Durchführung ist, dass nur berechtigte und authentische Terminals Zugriff über einen verschlüsselten Kanal auf die Daten des nPA erhalten.

Die **Terminal-Authentifizierung (TA)** dient der Authentisierung des Dienstanbieter-Terminals gegenüber dem nPA zum Auslesen sensibler Daten und dient dem Nachweis der Zugriffsrechte eines Dienstanbieters. Der eID-Service sendet die Zertifikatskette (CVCA Link Certificates, DV Certificate, Terminal Certificate) [22]. Danach kommt das asymmetrische Challenge-Response-Protokoll zur Anwendung. Challenge: nPA prüft die Zertifikate, extrahiert den öffentlichen Schlüssel und sendet eine Zufallszahl an den eID-Service; Response: Die vom eID-Service signierte Zufallszahl wird an den nPA zurückgesendet und vom Chip validiert.

Die **Chip-Authentication (CA)** des Personalausweises gegenüber dem eID-Service dient dem Nachweis der Echtheit des Chips, durch Einsatz des auf dem Chip gespeicherten privaten Chip-Authentisierungsschlüssels [22]. Weiter dient die Chipauthentifizierung dem Aufbau eines sicheren Kanals zwischen eID-Service und nPA.

Entsprechend der TR-03127 werden die Zugriffsrechte des Terminals an die in der Chip-Authentisierung ausgehandelten Sitzungsschlüssel gebunden, d.h. die Rechte des Terminals können nur innerhalb des durch die Chip-Authentisierung aufgebauten verschlüsselten Kanals ausgeübt werden. Die Verifizierung der Signatur ist die passive Authentisierung des nPA. Die Datengruppen der nicht-hoheitlichen eID-Anwendung werden nicht signiert [2].

4.1.4. Sequenzdiagramm zur eID-Funktion

Der Ablauf im nachfolgenden Sequenzdiagramm wurde auf Basis der Technischen Richtlinien TR-03127, TR-03128 und TR-03130 erstellt [2] [21] [23].

Von besonderer Bedeutung für das Szenario „Nutzer will im Rahmen der Web-Anwendung den elektronischen Identitätsnachweis nutzen“ sind das Zusammenspiel von Browser-Plugin und EID-Client sowie die Nutzerinteraktion zur PIN-Eingabe. Diese Funktionen werden in den nachfolgenden Kapiteln näher betrachtet.

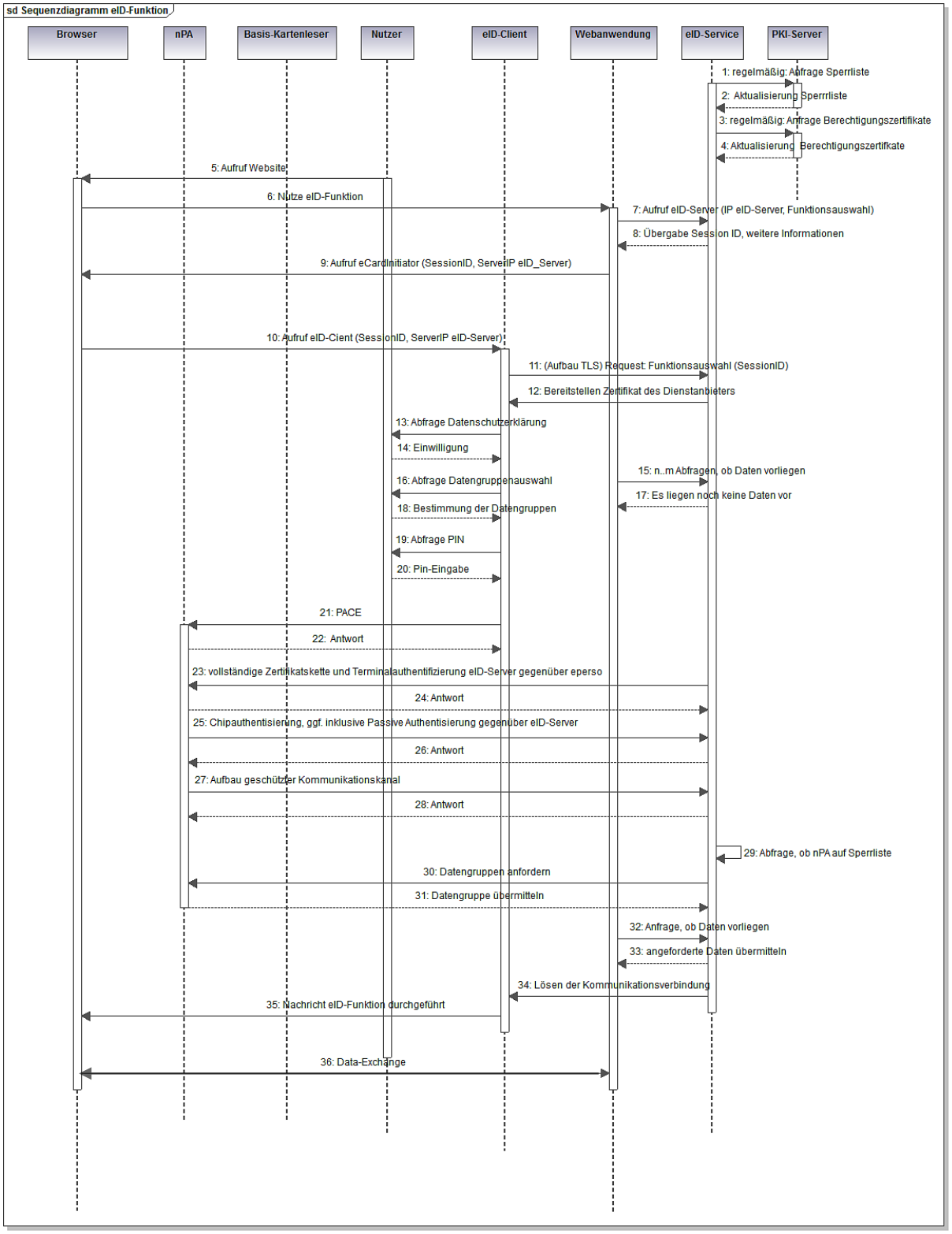

Abbildung 7: Ablauf der eID-Funktion [2] [21] [23]

4.2. Schutzziele und Schutzbedarfsermittlung

Die personenbezogenen Daten des nPA müssen geschützt werden. Die Grundlage für diese Forderung ergibt sich aus dem § 1 Bundesdatenschutzgesetz (BDSG), das den Einzelnen davor schützt, dass er durch den Umgang mit seinen personenbezogenen Daten in seinem Persönlichkeitsrecht beeinträchtigt wird [38]. Alle personenbezogenen Daten, wenn sie persönliche (z.B. Name und Vorname) oder sachliche Verhältnisse (z.B. E-Mail-Adresse, IP-Adresse) einer natürlichen Person beschreiben unterliegen laut § 4 in Verbindung mit § 13 BDSG dem Verbotsprinzip mit Erlaubnisvorbehalt. Damit ist die Erhebung, Verarbeitung und Nutzung von personenbezogenen Daten im Prinzip verboten. Sie ist nur dann erlaubt, wenn das Gesetz die Datenverarbeitung erlaubt oder die betroffene Person ausdrücklich die Zustimmung zur Erhebung, Verarbeitung und Nutzung gegeben hat.

In den folgenden Abschnitten werden auf Basis der gesetzlichen Grundlage und den in Abschnitt 3.1 genannten Grundwerten der Informationssicherheit: Vertraulichkeit, Integrität und Verfügbarkeit die Schutzziele und der Schutzbedarf analysiert.

4.2.1. Schutzziele

Aus der Anforderung Vertraulichkeit folgt, dass keine unautorisierte Informationsgewinnung erfolgen darf. Es muss durch Maßnahmen und Kontrollen sichergestellt sein, dass nur Berechtigte die Information erlangen. Die berechtigten Subjekte müssen sich authentifizieren. Eckert versteht unter Authentizität die Echtheit und Glaubwürdigkeit des Objektes bzw. Subjektes, die anhand einer eindeutigen Identität und charakteristischen Eigenschaften überprüfbar ist [39]. Die Authentisierung ist nicht nur Voraussetzung für den Grundwert Vertraulichkeit, sondern auch für die Gewährleistung des Grundwerts Integrität. Die Integrität soll gewährleisten, dass es Subjekten nicht möglich ist, die zu schützenden Daten unautorisiert und unbemerkt zu manipulieren.

Die Verfügbarkeit eines Systems ist gegeben, wenn authentifizierte und autorisierte Subjekte in der Wahrnehmung ihrer Berechtigungen nicht unautorisiert beeinträchtigt werden.

Zusammenfassend lassen sich folgende Schutzziele ableiten:

Vertraulichkeit:

Personenbezogene Daten des nPA sind vertraulich und dürfen nicht unberechtigt zur Kenntnis genommen oder weitergegeben werden.

Integrität:

Die Korrektheit, der aus dem nPA ausgelesenen Daten, muss sichergestellt sein.

Authentizität:

Die Authentizität von Personen und Komponenten, die auf die personenbezogenen Daten des nPA zugreifen wollen, muss verifiziert werden.

Verfügbarkeit:

Die Prozesse zum elektronischen Identitätsnachweis stehen zur Verfügung.

4.2.2. Schutzbedarfsanalyse

Das Ziel der Schutzbedarfsanalyse besteht darin, anhand der möglichen Schäden den Schutzbedarf der eID-Anwendung im Hinblick auf die oben genannten Schutzziele abzuleiten. Gemäß BSI-Standard 100-2 wird der Schutzbedarf der Anwendung in „niedrig bis mittel“, „hoch“ sowie „sehr hoch“ auf Basis der entwickelten Schadensszenarien kategorisiert. Das BSI stellt nach der Vorgehensweise entsprechend dem IT-Grundschutz folgende Schadensszenarien zur Verfügung. [40]:

1. Verstöße gegen Gesetze, Vorschriften oder Verträge
2. Beeinträchtigungen des informationellen Selbstbestimmungsrechts
3. Beeinträchtigungen der persönlichen Unversehrtheit
4. Beeinträchtigungen der Aufgabenerfüllung
5. negative Außenwirkung
6. finanzielle Auswirkungen

Die in dem gewählten Szenario zu den Grundwerten: Vertraulichkeit, Integrität und Authentizität denkbaren Schäden und deren Folgen lassen sich diesen vom BSI systematisierten Schadensszenarien zuordnen und eine Schutzkategorie („niedrig bis mittel“, „hoch“ sowie „sehr hoch“) zuweisen. Aufgrund eines Angriffes können

erhebliche, sogar für die Person existenzbedrohende Schäden entstehen, die den Schadensszenarien „negative Innen- oder Außenwirkung“ und „finanzielle Auswirkungen“ zugeordnet werden können. Ein Beispiel aus dem Schadensszenario „finanzielle Auswirkungen“ ist die Zulassung eines Kraftfahrzeugs mit einer fremden Identität. Dies kann bei einem schweren Personenunfall, mit dem auf die fremde Identität zugelassenen Fahrzeug, für den vermeintlichen Halter des Fahrzeugs aufgrund der Haftung für Unfallschäden existenzbedrohende finanzielle Auswirkungen haben. Ein Beispiel aus dem Schadensszenario „negative Innen- oder Außenwirkung“ ist der Imageverlust einer Person, dessen Identität geraubt und ausgenutzt wird. Die Veröffentlichung von Geschäften, die illegal (z.B. Ausnutzung einer geraubten Schufa-Auskunft) oder unsittlich sind (z.B. Pornographie) würden dem Image erheblichen Schaden zufügen. Der Einsatz der eID-Funktion ermöglicht neue Rechtsgeschäfte für Privatpersonen im Internet. Diese Geschäfte führen bei erfolgreichen Angriffen ggf. zu höheren Schäden als bei den bisher im Internet möglichen Geschäften. Die seitherige Authentisierung mit Benutzerkennung und Passwort war geprägt durch zahlreiche unterschiedliche Passwörter für die einzelnen Authentisierungsvorgänge. Bei Nutzung der neuen eID-Funktion gibt es für diese Funktion für alle Dienstanbieter das gleiche Passwort, die PIN. Ein Angreifer kann mit der geraubten eID-PIN die Identität des Nutzers bei allen Diensten mit eID-Funktion ausnutzen. Die Schadensszenarien können von erheblichem, ja sogar wie oben dargestellt von existenzbedrohendem Ausmaß für die Privatperson sein.

Schäden resultierend aus der unzureichenden Verfügbarkeit der eID-Prozesse können den Schadensszenarien unter 1.: Verstöße gegen Gesetz, Vorschriften oder Verträge oder den Szenarien unter 5.: Negative Außenwirkung oder unter 6.: Finanzielle Auswirkungen; zugeordnet werden. Die nicht verfügbare technische eID-Funktion kann kurzfristig durch etablierte Verfahren zur Identifikation ersetzt werden. Solche kurze Zeitverzögerungen haben tolerable Auswirkungen.

In der nachfolgenden Übersicht wird das Ergebnis der Analyse zusammengefasst:

Schutzbedarf der AusweisApp		
Grundwert	**Schutzbedarf**	**Schutzziel**
Vertraulichkeit	Hoch	Die personenbezogenen Daten des nPAs dürfen Unberechtigten nicht zur Kenntnis gelangen
Integrität	Hoch	Die Unversehrtheit der personenbezogenen Daten bei Nutzung der eID-Funktion muss gewährleistet sein
Authentizität	Hoch	Die Identitäten der Sender und Empfänger müssen vor Beginn der Kommunikation von personenbezogenen Daten eindeutig bewiesen werden
Verfügbarkeit	Niedrig	Die Prozesse zur eID-Funktion stehen zur Verfügung

Tabelle 3: Ergebnis der Schutzbedarfsanalyse zur AusweisApp (vgl. [23])

Der Schutzbedarf für die Grundwerte: Vertraulichkeit, Integrität und Authentizität ist mit hoch eingestuft. Nur die Verfügbarkeit ist mit niedrig eingestuft. Die Anwendung AusweisApp erhält den Schutzbedarf „hoch“.

4.3. Bedrohungsanalyse

Zu der Aufgabe dieser Arbeit gehört es, die aus der Anwendung der eID-Funktion hervorgehende Bedrohungslage zu untersuchen. Die Bedrohungen auf das Gesamtsystem zur Nutzung der eID-Funktion werden auf Basis der Schutzziele im Überblick dargestellt. Anschließend wird die Bedrohungslage entsprechend der Zielsetzung der Diplomarbeit: „Verhinderung von Angriffen, die das Ziel haben die Identität des Ausweisinhabers zu rauben“ identifiziert.

Die Angriffsvektoren auf das Gesamtsystem, bei Nutzung der eID-Funktion, können auf den Mensch als Nutzer oder Mitarbeiter im Rechenzentrum, auf den PC des Nutzers (Betriebssystem, Speicher, Dateisystem, Anwendungen), auf Kartenleser oder Personalausweis, auf die Rechner in den Rechenzentren, deren dazugehörigen Betriebssystemen, deren Middleware, deren Anwendungen, Speicher und

Dateisystemen inklusive der Datensicherung und deren internen und externen Netzwerke wirken. Die nachfolgende Aufstellung erhebt nicht den Anspruch auf Vollständigkeit. Ziel dieser Übersicht ist die Darstellung der typischen sowie der eID-spezifischen Gefährdungen. Grundlage für die Übersicht sind die Grundschutzkataloge G 1- 5 des BSI [18]: G 1 Höhere Gewalt , G 2 Organisatorische Mängel [41], G 3 Menschliche Fehlhandlungen, G 4 Technisches Versagen und G 5 Vorsätzliche Handlungen [42] sowie die Zusammenstellung des BSI zu typischen Gefährdungen für den Internet-PC [43], für Allgemeine Server [44] und für Heterogene Netze [45].

Schutzziel: „Die personenbezogenen Daten des nPAs dürfen Unberechtigten nicht zur Kenntnis gelangen“

Unberechtigte können auf folgende Weise Daten ausspähen. Sie können:

1. Die unzureichende Kenntnis des Benutzers über Regelungen ausnutzen G 2.2
2. Konzeptionelle Schwächen des Netzes ausnutzen G 2.45
3. Rechte unerlaubt ausüben G 2.7
4. Das Fehlverhalten des Benutzers ausnutzen G 3.1
5. Die Nichtbeachtung von Sicherheitsmaßnahmen ausnutzen G 3.3
6. Die fehlerhafte Administration des Benutzer PCs ausnutzen G 3.9
7. Konfigurations- und Bedienungsfehler ausnutzen G 3.38
8. Software-Schwachstellen oder –Fehler ausnutzen G 4.22
9. Die Kommunikation zwischen den oben genannten Komponenten abhören G 5.7
10. Systematisch Passwörter ausprobieren G 5.18
11. Trojanische Pferde einsetzen G 5.21
12. Durch Schadprogramme ausspähen G 5.23
13. Mit einem nachgebauten, gefälschten nPA, Kartenlesegerät, AusweisApp oder Browser in Kenntnis der Daten gelangen
14. Durch Social Engineering ausnutzen G 5.42
15. Auf eine der folgenden Komponenten: PC des Benutzer mit eID-Client und Browser, Web-Anwendung, eID-Service und PKI-Service (G 5.71) durch
    a. Auslesen von Dateien (Skimming)
    b. Kopieren von Dateien
    c. Wiedereinspielen von Datensicherungsbeständen
    d. Diebstahl des Datenträgers und anschließendes Auswerten
    e. Mitlesen an der Tastatur oder am Bildschirm

16. Makro-Viren einsetzen G 5.43
17. Netzmanagement-Funktionen unberechtigt ausführen G 5.67
18. Durch Spoofing ausnutzen G 5.48, g 5.78, G 5.87
19. Aktive Inhalte missbrauchen G 5.88
20. Webmail missbrauchen G 5.103
21. Durch einen Man-in-the-Middle-Angriff angreifen G 5.143

Schutzziel: „Die Unversehrtheit der personenbezogenen Daten bei Nutzung der eID-Funktion muss gewährleistet sein“

Ein Angreifer kann auf folgende Weise gefälschte Daten einspeisen:

Er kann:

1. Den Inhalt der Daten über Zugriff auf eine der oben genannten Komponenten, wie unter Schutzziel Vertraulichkeit dargestellt, ausspähen und sie daraufhin unberechtigt manipulieren
2. Informationen oder Software manipulieren G 5.2,

Neben den oben genannten Bedrohungen können auch:

1. Daten unabsichtlich manipuliert werden G 3.24

Schutzziel: „Die Identitäten der Sender und Empfänger müssen vor Beginn der Kommunikation von personenbezogenen Daten eindeutig bewiesen werden“

Ein Angreifer kann auf folgende Weise eine falsche Identität vortäuschen. Er kann:

1. Die Unzureichende Kenntnis der Beteiligten über Regelungen ausnutzen G 2.2
2. Die Fehlerhafte Administration von IT-Systemen ausnutzen G 3.9
3. Konfigurations- und Bedienungsfehler ausnutzen G 3.38
4. Software-Schwachstellen oder –Fehler ausnutzen G 4.22
5. Informationen oder Software manipulieren G 5.2,
6. Das Kryptomodul manipulieren G 5.82
7. Kryptographische Schlüssel kompromittieren G 5.83
8. Zertifikate fälschen G 5.84
9. Schlechte oder fehlende Authentifikation ausnutzen G 4.33 (Eingriff in den Austausch von Sitzungsparametern)
10. Unberechtigte Netzmanagement-Funktionen ausführen G 5.67
11. Durch Spoofing ausnutzen G 5.48, g 5.78, G 5.87
12. Aktive Inhalte missbrauchen G 5.88

13. Netz-Verbindungen hijackenG 5.89
14. Webmail missbrauchen G 5.103
15. Durch einen Man-in-the-Middle-Angriff angreifen G 5.143

Schutzziel „die eID-Prozesse stehen zur Verfügung“

Ein Angreifer kann auf folgende Weise die Verfügbarkeit einer der folgenden Komponenten: nPA, Kartenleser, PC des Benutzer mit eID-Client und Browser, Web-Anwendung, eID-Service und PKI-Service beeinträchtigen: Es können

1. Eine der Komponenten ausfallen, gestört oder blockiert werden G 4.1, G 4.31, G 4.34
2. Komponenten manipuliert oder zerstört werden G 5.1, G 5.5, G 5.6
3. Informationen oder Software manipuliert werden G 5.2,
4. Komponenten gestohlen werden G 5.4
5. Dienste verhindert werden G 5.28
6. Komponenten sabotiert werden G 5.102

Nach dem Überblick zur Bedrohungslage des Gesamtsystems sollen nun die Bedrohungen zum Kernthema dieser Arbeit analysiert werden. Eckert schreibt, dass sich Bedrohungsbäume zur systematischen Analyse der unterschiedlichen Wege zur Erreichung eines Angriffsziels eignen [46]. Der nachfolgende Bedrohungsbaum zum Angriffsziel „Rauben der Identität“ wird textuell dargestellt und soll alle für das Szenario relevante Bedrohungen systematisch erfassen. Der Diebstahl der Identität kann durch folgende Bedrohungen erfolgen:

Angriffsziel: Diebstahl der Identität

UND Subziel 1: Kenntnis der PIN erlangen

ODER Subziel 1.1: PIN aus dem nPA rauben

ODER Subziel 1.1.1: Kenntnis der PIN durch erraten/Versuch und Irrtum

Subziel 1.1.2: Rückrechnung des Schlüssels aus Kommunikation über

Funk-Schnittstelle und Speicherzugriff

Subziel 1.1.3: Zugriff auf Speicher des nPAs zum Auslesen der PIN

Subziel 1.1.4: PIN per Hardwareangriff auslesen

Subziel 1.2: PIN durch Userangriff rauben

ODER Subziel 1.2.1: Inhaber des elektronischen Personalausweises zur Preisgabe der PIN verleiten

Subziel 1.2.2: Bedrohen, erpressen des Inhabers des elektronischen Personalausweises zur Preisgabe der PIN

Subziel 1.2.3: Ausspähen der PIN (Sniffing mittels Keylogger)

Subziel 1.2.4: Durch Notiz der PIN Kenntnis erlangen

Subziel 2: Zugriff auf den nPA

ODER Subziel 2.1: Inbesitznahme des Ausweises

ODER Subziel 2.1.1: Verlorener/gefundener Ausweis

Subziel 2.1.2: Bedrohen, erpressen des Inhabers des elektronischen Personalausweises zur Übergabe des Ausweises

Subziel 2.1.3: Stehlen des Ausweises

Subziel 2.2: Nachbauen/Duplizieren eines nPA

Subziel 2.3: Zugriff über Benutzer PC

ODER Subziel 2.3.1: Angriff auf die AusweisApp selbst

Subziel 2.3.2: Angriff auf den Browser (inkl. Browser-Plugin)

Subziel 2.3.3: Angriff an der USB-Schnittstelle

Subziel 2.4: Einsatz von RFID-Malware (Manipulation des Programmcodes des nPAs)

Für den Angreifer sind die Kenntnis der PIN und der Zugriff auf den nPA notwendige Voraussetzungen für die Ausnutzung der fremden Identität. Solche Angriffe unter Nutzung des Basiskartenlesers wurden bereits demonstriert.
Bereits vor Veröffentlichung der AusweisApp hat der Chaos Computer Club CCC in der Sendung vom 22. September 2010 "Bericht aus Brüssel" im WDR vor Angriffen auf die PIN mittels Trojanern (Keylogger) gewarnt [7].

Ca. zweieinhalb Monate nach Einführung des neuen Personalausweises, am 17.01.2011, ist es Jan Schejbal gelungen, unbemerkt ein trojanisches Pferd als „falsche" AusweisApp einzuschleusen und eine gefälschte Authentifizierungs-Anwendung auszuführen. Bei dieser Demonstration hat der Anwender die gefälschte eID-Funktion nicht erkannt und die PIN eingegeben [8]. In weiteren Schritten könnte der Angreifer die erhaltene Information z.B. per E-Mail an sich senden. In diesen Fällen betrifft die Schwachstelle nicht den nPA selbst, sondern das Zusammenspiel des Basiskartenlesers mit dem PC.

Am 08.08.2011 veröffentlichte Heise Security eine weitere Demonstration von Jan Schejbal für einen erfolgreichen Angriff zur Ausnutzung der fremden Identität. In diesem Angriff nutzt der Angreifer die Möglichkeit des Browser-Plugins (OWOK-Plugin), per JavaScript einen Kanal zur Chipkarte zu öffnen. Darüber könnte der Angreifer beliebige APDUs (Application Protocol Data Units) an die Karte schicken und die Antworten des nPA lesen [9].

4.4. Risikoanalyse

Entsprechend der in Abschnitt 3.4 dargestellten Vorgehensweise erfolgt in diesem Abschnitt die Abschätzung der Eintrittswahrscheinlichkeiten und des Schadenspotenzials. Die Schadenshöhe ist bei allen Zweigen des Bedrohungsbaumes identisch, weil der mögliche Schaden unabhängig von der ausgenutzten Schwachstelle ist. Daraus folgt, dass das Risiko ausschließlich von der Eintrittswahrscheinlichkeit abhängt. Die Eintrittswahrscheinlichkeit einer Bedrohung hängt von dem geschätzten Aufwand und der Einschätzung des möglichen Nutzens ab, den der Angreifer aus einem erfolgreichen Angriff ziehen könnte. Laut BSI liegen für die meisten Szenarien keine verlässlichen statistischen Informationen, über die Häufigkeit sicherheitsrelevanter Vorgänge und die Höhe der dabei entstandenen Schäden, vor [47]. Zur Abschätzung des

Risikos werden in der Praxis Experten eingesetzt. Diese führen zum Beispiel einen Source-Code Review durch oder erstellen ein eigenes Sicherheitskonzept und gleichen dieses mit dem Sicherheitskonzept aus dem Software Engineering Prozess ab, um das Risiko zu bewerten. Diese Vorgehensweisen sind aufgrund der mangelnden Verfügbarkeit, der notwendigen Informationen, sowie aufgrund der Rahmenbedingungen dieser Diplomarbeit nicht möglich.

Zur Bewertung des Risikos im Rahmen dieser Arbeit wird das Risiko in die zwei Kategorien „gering“ und „nicht gering“ angegeben. Es erhalten alle Bedrohungen die Kategorie „nicht gering“, zu denen dem Autor bereits erfolgreiche, veröffentlichte Angriffe bekannt sind. Die relevanten Bedrohungen wurden in Abschnitt 4.3 bereits dargestellt. Alle anderen Bedrohungen erhalten die Bewertung „gering“.

Mit der gewählten Vorgehensweise werden die Eintrittswahrscheinlichkeiten für die Bedrohungen „Erpressen“ und „Bedrohen“ (Subziele: 1.2.2 und 2.1.2) und „Durch Notiz Kenntnis erlangen“ (Subziel: 1.2.4) unzureichend erfasst. In Veröffentlichungen von solchen Straftaten werden gegebenenfalls solche Details nicht veröffentlicht. Die Verbesserung der Abwehr dieser Angriffe liegt jedoch nicht im Fokus dieser Arbeit.

Abbildung 8 zeigt, dass das Risiko für den Angriff „Diebstahl der Identität“ unter Einsatz des Basis-Kartenlesers nicht gering ist. Entsprechend der Zielsetzung dieser Arbeit, sollen die drei Bedrohungen mit der Risikoeinstufung nicht gering, im Folgenden näher betrachtet werden.

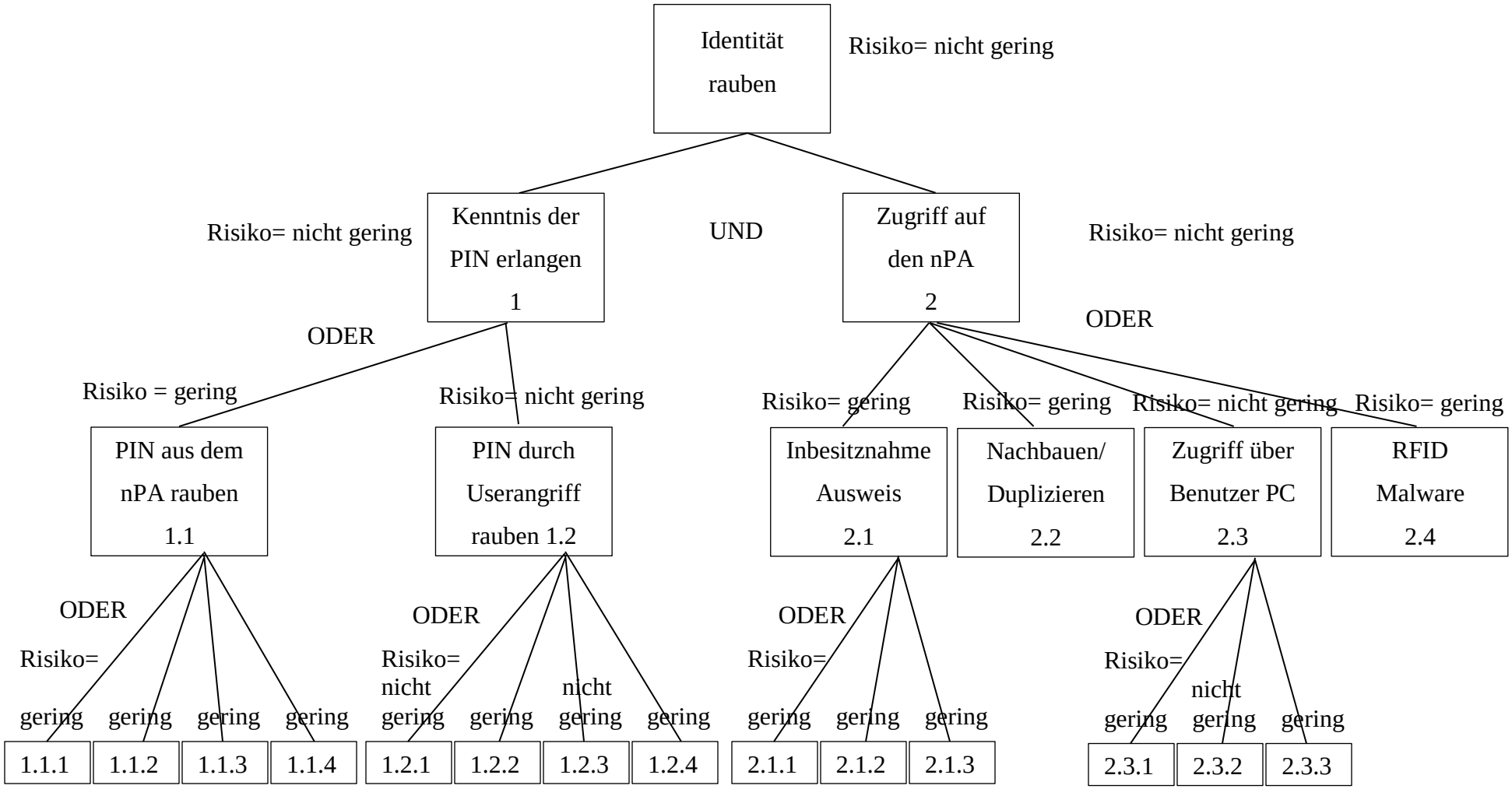

Abbildung 8: Risikoanalyse: Angriffsziel "Identität rauben"

4.5. Analyseergebnisse und Ableitung der Anforderungen

Nachfolgend werden die Schlussfolgerungen aus der Analyse dargestellt und die Anforderungen an den Entwurf und den Prototypen abgeleitet, mit denen die Nutzung der eID-Funktion sicherer werden soll.

Alle Risiken mit Einstufung nicht gering, resultieren nicht aus Schwachstellen des nPA bzw. der AusweisApp, sondern aus dem Zusammenspiel des Basiskartenlesers bzw. des Browsers inklusive dem Browser-Plugin mit dem PC. Bei den oben dargestellten Angriffen auf die eID-Funktion (siehe Abschnitt 4.3) war Voraussetzung, dass der Angreifer die vorhandenen Sicherheitsdienste des Betriebssystems umgangen, täuschen oder unautorisiert modifizieren konnte und Schadsoftware, z.B. einen Keylogger installieren konnte. Laut einer BITKOM-Umfrage aus dem Jahr 2011 haben 47% der deutschen Internet-Nutzer über 14 Jahre Erfahrungen mit Schadprogrammen gemacht. Bei 13% der Internet-Nutzer (knapp 7 Millionen Deutsche Internet-Nutzer) wurden die Zugangsdaten ausspioniert. Im Vorjahr waren es nur rund 3,7 Millionen [48]. Das BSI geht in seinem Lagebericht zur IT-Sicherheit 2011 von einer weiterhin sehr starken Zunahme von Schadprogrammen aus [16]. Daraus lässt sich schließen, dass die PIN-Eingabe über die Tastatur des PCs und dessen Weitergabe an den Kartenleser dazu führt, dass die sichere Nutzung der eID-Funktion von dem Sicherheitskonzept des Betriebssystems bzw. vom Schutzniveau des PCs direkt abhängt. Entsprechend Abschnitt 4.2.2 ist der Schutzbedarf hoch. Das Sicherheitsniveau eines typischen PCs in Deutschland ist nicht ausreichend für diesen Schutzbedarf. Es besteht ein nicht geringes Risiko, dass die Identität des Ausweisinhabers in dem gewählten Szenario geraubt und ausgenutzt werden kann.

Daraus folgt, dass die Schutzanforderungen der eID-Funktion nicht erfüllt werden können. Eine Anforderung an eine Lösung ist, dass sie das Sicherheitsniveau des typischen PCs verbessert. Damit wird das Sicherheitsniveau der AusweisApp bezüglich der oben genannten Angriffe mit nicht geringem Risiko verbessert. Neben den sicherheitsrelevanten Auswirkungen der Lösung müssen auch die weiteren Effekte der Lösungskonzepte in den Anforderungen berücksichtigt werden. So sind Auswirkungen auf die Anwenderfreundlichkeit, Performance und den Funktionsumfang in die Auswahl der

Lösungsalternative aufzunehmen. Zusammenfassend können daraus folgende Anforderungen abgeleitet werden:

1. Verbesserung des Sicherheitsniveaus der AusweisApp bezüglich der Bedrohungen „PIN durch User-Angriff rauben“ sowie „Zugriff auf den nPA durch Zugriff auf den Benutzer PC“ durch Isolierung der Funktionen zur eID-Anwendung
2. Beibehaltung des Sicherheitskonzeptes der AusweisApp
3. Die Nutzung der eID-Funktion soll anwenderfreundlich bleiben
4. Die Performance bei der Nutzung der eID-Anwendung soll gleich bleiben
5. Die vorhandenen Funktionen (z.B. Pseudonymisierungfunktion, Altersverifikation, usw.) der eID-Anwendung müssen weiterhin betrieben werden können

Bezüglich der Lösungsansätze gibt es folgende Beschränkungen: Der Quellcode der AusweisApp ist nicht verfügbar. Änderungen an der Anwendung selbst sind daher nicht möglich. Die Bedrohungen, die sich im Verhalten des Ausweisinhabers begründen (Notiz der PIN) sowie das Erpressen und Bedrohen der Person, um Zugriff auf den nPA zu erhalten, sind nicht im Fokus dieser Arbeit und werden bei der Konzepterstellung nicht weiter berücksichtigt.

5. Konzeptioneller Entwurf

Das neue Konzept soll das Risiko der Angriffsvektoren mit der Risikoeinstufung „nicht gering“ deutlich mindern. Die Lösung soll das Sicherheitsniveau des PCs bei Nutzung der AusweisApp unter Berücksichtigung der weiteren Anforderungen aus Abschnitt 4.5 erhöhen.

In diesem Kapitel werden zunächst die Empfehlungen des BSI dargestellt und Lösungsansätze diskutiert. Anschließend werden im Rahmen der Nutzwertanalyse Lösungsalternativen erarbeitet und bewertet. Der Ansatz, der die oben aufgeführten Anforderungen bestmöglich erfüllt, wird im Rahmen des konzeptionellen Entwurfs ausgearbeitet.

5.1. Diskussion zu den Lösungsalternativen

Aus der Analyse geht hervor, dass das Sicherheitsniveau des PCs bei Nutzung der eID-Funktion nicht ausreichend ist. Es gibt gesetzliche Anforderungen, die beachtet werden müssen: Laut § 23 Absatz 2 Personalausweisverordnung muss der Ausweisinhaber sicherstellen, dass bei Nutzung des elektronischen Identitätsnachweises, seine informationstechnischen Systeme mit geeigneten Abwehrmaßnahmen gegen Sicherheitslücken nach dem Stand der Technik eingesetzt werden [49]. Ausgehend von der Personalausweisverordnung empfiehlt das Bundesamt für Sicherheit in der Informationstechnik (BSI) folgende Maßnahmen zum Schutz der Nutzerumgebung [50]:

- Nutzung eines aktuellen Virenschutzes und Firewall und regelmäßige Einspielung aller Sicherheitsupdates sowie Hinweis zur Einhaltung der 10 Tipps zum Basisschutz [51]:
    1. Verwendung eines aktuellen Virenschutzprogramms und Anti-Spyware-Programms und Einsatz einer aktuellen Personal Firewall
    2. Zeitnahe Durchführung der Sicherheitsupdates für Betriebssystem und Software
    3. Unterschiedliche Benutzerkonten einrichten und nach Möglichkeit nicht als Administrator arbeiten

4. Nutzung der bei Browser integrierten Funktion zur Warnung vor als bösartig bekannten Webseiten
5. Sorgfältiger Umgang mit Zugangsdaten: Kennwörter, Benutzernamen, Zugangscodes sowie Wechsel der Passwörter in regelmäßigen Abständen
6. Beim Öffnen von E-Mail-Anhängen im Zweifelsfall beim Absender nachfragen, ob der Anhang tatsächlich von ihm stammt.
7. Bei Downloads von Webseiten nur von vertrauenswürdigen Quellen nutzen
8. Persönliche Informationen nur zurückhaltend weitergeben
9. Bei Übertragung durch Voice over IP, Wireless LAN nur verschlüsselt kommunizieren
10. Regelmäßige Erstellung von Sicherungskopien der wichtigsten Dateien auf CD oder externer Festplatte

- Den Personalausweis nur bei Verwendung der eID-Funktion auf das Lesegerät zu legen und unmittelbar nach Abschluss der Authentisierung wieder weg zu nehmen
- Nutzung von zertifizierten Karten-Lesegeräte

Wie in Abschnitt 4.5 dargestellt, nehmen Schwachstellen in den komplexer werdenden Anwendungen und darauf abzielende Schadprogramme zu. Die Reaktion darauf ist, regelmäßig den Virenscanner zu aktualisieren und die Sicherheitsupdates zeitnah auszuführen, um das Gesamtsystem nicht zu gefährden. Der typische PC enthält Anwendungen mit unterschiedlichen Sicherheitsanforderungen. Beispielsweise wird auf einem PC mit dem Browser im Internet gesurft und im nächsten Schritt Online-Banking durchgeführt. Die Mehrzahl der Anwender hat diese Konfiguration. Bei einem erfolgreichen Angriff, resultierend aus dem Besuch einer nicht vertrauenswürdigen Webseite, ist das gesamte System, so auch die eID-Funktion kompromittiert. Es könnte ein Keylogger installiert sein und die Karte würde noch auf dem Kartenleser liegen. Der Angreifer könnte nun mit der falschen Identität alle eID-Funktionen im Internet missbrauchen. Es lässt sich feststellen, dass dieser Ansatz alleine nicht geeignet ist, ein ausreichendes Sicherheitsniveau auf dem PC sicherzustellen. Zur Verbesserung des Sicherheitsniveaus wird - in Ergänzung der oben genannten Maßnahmen - folgender Ansatz die Grundlage des konzeptionellen Entwurfs: Separierung der potenziell nicht

vertrauenswürdigen Anwendungen von der AusweisApp. In den folgenden Abschnitten werden auf dieser Basis verschiedene Lösungsalternativen, die hinreichend voneinander trennbar sind erarbeitet und bewertet.

5.2. Ansatz

Zur Bewertung der Lösungsalternativen stehen verschiedene Bewertungsverfahren zur Verfügung. Nagel hat verschiedene Methoden zur Nutzenanalyse untersucht und die Nutzwertanalyse als geeignetste Methode zur Auswahl von Handlungsalternativen empfohlen [52]. Die Nutzwertanalyse erlaubt es, anhand mehrerer Anforderungskriterien und aufgrund subjektiver Wertvorstellungen eine Wahl unter verschiedenen Lösungsalternativen zu treffen. Nagel beschreibt die notwendigen Schritte, die zur Auswahl der Lösungsalternative mit dem höchsten Nutzwert führen, wie folgt [53]:

a) Ziel der Entscheidung definieren
b) Festlegen der Forderungen, die unbedingt erfüllt werden müssen
c) Aufstellen der Auswahlkriterien
d) Gewichten der Auswahlkriterien
e) Erarbeiten der Alternativen
f) Bewerten der Alternativen
g) Auswahl der besten Alternative als Entscheidung

Dieses Verfahren soll wie folgt in der Arbeit angewendet werden: Aus (b) folgt, dass Lösungsansätze, die die unbedingten Anforderungen nicht erfüllen, nicht weiter verfolgt werden. Für die praktische Durchführung werden die Anforderungen entsprechend ihrem Einfluss auf die Zielerreichung gewichtet (d). Ein unterdurchschnittlicher Prozentsatz - der Prozentsatz ist kleiner als 1/Anzahl der Auswahlkriterien*100 - gibt einen unterdurchschnittlichen Einfluss der Anforderungen auf die Zielerreichung wieder und umgekehrt. Die Summe der Gewichtungen ergibt 100%. Für die Bewertung der Alternativen (f) wird das Punktebewertungsverfahren angewendet. Für die Erfüllung einer Anforderung werden Wertzahlen von 1 bis 10 (1 = geringste Erfüllung, 10 = Anforderung wird voll erfüllt) vergeben. Die Punktewertzahlen werden mit den gewichteten Anforderungen multipliziert. Die beste Alternative zur Zielerreichung ist die Alternative mit der höchsten Summe der gewichteten Punktwertzahlen (g).

5.3. Zielsetzung der Entscheidung

Ziel der Entscheidung ist, die geeignetste Lösungsalternative unter den erarbeiteten Alternativen zu wählen, die das Sicherheitsniveau bei Nutzung der eID-Funktion in dem gewählten Szenario maximiert und die Nebeneffekte durch Anwendung des Konzeptes (vgl. Anforderungen aus Abschnitt 4.5) berücksichtigt.

5.4. Festlegen der Forderungen, die unbedingt erfüllt werden müssen

Die Anforderungen aus Abschnitt 4.5 werden daraufhin überprüft, inwieweit es dabei um Anforderungen handelt, die unbedingt erfüllt werden müssen. Die Anforderung zwei, Beibehaltung des Sicherheitskonzeptes ist unbedingt zu erfüllen, da die eID-Funktion bei Änderung des Sicherheitskonzeptes unter Umständen nicht mehr betrieben werden dürfte. Es wäre ggf. die Einhaltung der Technischen Richtlinien nicht mehr gewährleistet. Anforderung fünf, Beibehaltung des Funktionsumfangs ist unbedingt zu erfüllen, da der elektronische Identitätsnachweis unter anderem aus Gründen des Datenschutzes alle Funktionen laut Abschnitt 4.1.1 benötigt. Weitere unbedingt zu erfüllende Anforderung ist, dass die gesetzlichen Vorgaben eingehalten werden und dass die Lösung Potenzial auf Verbesserung des Sicherheitsniveaus mitbringt. Die Forderungen, die unbedingt erfüllt werden müssen lauten:

a. Alle Lösungsansätze müssen ein hinreichendes Sicherheitsniveau gewährleisten, das mindestens die Anforderungen aus § 23 Absatz 2 Personalausweisverordnung erfüllt [49]. Darüber hinaus muss die Lösungsalternative Potenzial haben das Schutzniveau zu verbessern
b. Beibehaltung Sicherheitskonzept
c. Beibehaltung Funktionsumfang

5.5. Aufstellen der Auswahlkriterien

Die Anforderungen aus Abschnitt 4.5, die mit Erfüllungsgraden zu bewerten sind, werden in die Auswahlkriterien übernommen:

1. Das resultierende Sicherheitsniveau der AusweisApp bezüglich der Bedrohungen „PIN durch User-Angriff rauben“ sowie „Zugriff auf den nPA durch Zugriff auf den Benutzer PC“ durch Isolierung der Funktionen zur eID-Anwendung verbessert sich

2. Die Anwenderfreundlichkeit bei Nutzung der eID-Funktion
3. Die Performance bei Nutzung der eID-Funktion

5.6. Gewichten der Auswahlkriterien

Die Anforderungen sollen mit unterschiedlicher Gewichtung W in die Bewertung eingehen Die Gewichtungen sind subjektive Einschätzungen des Autors nach der relativen Wichtigkeit für das Entscheidungsziel. Alle Anforderungen erhalten in Summe 100%. Die Gewichtungen der einzelnen Anforderung werden aufgrund folgender Überlegungen vergeben: Die Analyseergebnisse haben Handlungsbedarf bei der Verbesserung des Sicherheitsniveaus des PCs bei Nutzung der AusweisApp gezeigt, daher sollen die Anforderungen, die das Sicherheitsniveau verbessern 50% erhalten. Diese Verbesserung darf jedoch nicht an anderer Stelle die eID-Prozesse ungünstig beeinflussen. So könnte die Anwendung des Konzeptes Nebeneffekte haben, die in den Anforderungen zwei und drei berücksichtigt sind. Die Anwenderfreundlichkeit und die Performance werden je mit 25% gewichtet, da die Anwenderakzeptanz der Lösungsalternativen Voraussetzung ist, dass er die Anwendung auch nutzt.

5.7. Erarbeiten der Lösungsalternativen

Grundsätzlich gibt es verschiedene Möglichkeiten zur Absicherung der PIN-Eingabe bei Nutzung der AusweisApp. Für die Auswahl der Lösungsalternativen muss gelten, dass sie hinreichend trennbar von anderen Lösungsalternativen sind. Die in Abschnitt 5.1 genannte Empfehlung des BSI bildet die Grundlage für die Lösungsalternativen und stellt gleichzeitig den Ausgangspunkt dar.

Aufgrund folgender Überlegung erreicht die Empfehlung des BSI das Mindestschutzniveau für die sicherheitskritische eID-Funktion nicht. Entsprechend den Analyseergebnissen aus Abschnitt 4.5 reicht das Sicherheitsniveau des typischen PCs trotz Berücksichtigung der Empfehlungen des BSI nicht aus. Aus Abschnitt 5.1 geht hervor, dass auch mit Einhaltung der Empfehlung des BSIs bei einem erfolgreichen Angriff alle Anwendungen kompromittiert sind. Die Empfehlung des BSIs wird als Lösungsalternative aufgenommen unter der Voraussetzung, dass der Tipp 3 „Unterschiedliche Benutzerkonten einrichten und

nach Möglichkeit nicht als Administrator arbeiten“ in folgende Regel geändert wird: In der ersten Lösungsalternative sollen die Anwendungen nach ihrem Sicherheitsniveau in zwei Benutzer separiert werden.

I. **Separierung in zwei Benutzer:** Dieser Lösungsansatz ist konform mit der Empfehlung des BSI aus Abschnitt 5.1 und gibt darüber hinaus eine strengere Regel bezüglich der Separierung der Anwendungen in zwei Benutzer vor: Es werden zwei Benutzerkonten eingerichtet. Es erfolgt eine Trennung in einen Benutzer für die AusweisApp und einen zweiten Benutzer für alle anderen, potenziell nicht vertrauenswürdigen Anwendungen. Die Regel zur Separierung der Anwendungen wird am Beispiel Webbrowser erläutert: Der Browser kann wie folgt zweigeteilt und separiert werden: Es wird eine separate Gruppe von Anwendungen bzw. Funktionen mit der Bezeichnung „trusted Browser“ eingeführt, in der sich alle Anwendungen in dem Sicherheitskontext „vertrauenswürdig“ befinden. Hier bewegt sich der zweite Benutzer ausschließlich auf vertrauenswürdigen Webseiten und nutzt die eID-Funktion. Es wird eine zweite separierte Gruppe von Anwendungen bzw. Funktionen mit dem Namen „untrusted Browser“ eingeführt. Hier werden alle anderen Anwendungen von einem zweiten Benutzer betrieben. Beispielsweise kann sich der Anwender mit dem Browser in „untrusted Browser“ frei im World Wide Web bewegen und alle Webseiten aufrufen. Die Zuordnungen der weiteren Anwendungen bzw. Funktionen (Einkäufe im Internet unter Verwendung der Kreditkarte, E-Mail-Programm, usw.) hängen von den Sicherheitsanforderungen, der Vertrauenswürdigkeit des Anbieters und von der Nutzung ab und können entsprechend dem vertrauenswürdigen „trusted Browser“ oder dem nicht vertrauenswürdigen „untrusted Browser“ zugeordnet werden.

Der Grundgedanke für die zweite Lösungsalternative ist die Aufteilung der Anwendungen in zwei unterschiedliche physikalische Maschinen.

II. **Separierung in zwei PCs:** Die Anwendungen werden wie folgt zugeordnet: Im ersten PC ist „untrusted Browser“ installiert. Im zweiten PC befindet sich der „trusted Browser“ mit der eID-Funktion. In diesem Lösungsansatz werden die anderen Anwendungen „untrusted Browser“ auf eine physikalische Maschine und

die vertrauenswürdige Anwendung eID-Client inklusive „trusted Browser“ auf eine zweite physikalische Maschine installiert. Sofern der Anwender die eID-Funktion nutzen will, muss er den zweiten PC hochfahren und dort die Webanwendung des Dienstanbieters aufrufen, um die eID-Funktion zu nutzen. Natürlich darf der Anwender den „trusted Browser“ nur für vertrauenswürdige Webseiten benutzen. Ist der PC namens „untrusted Browser“ kompromittiert bleibt der PC „trusted Browser“ noch sicher. Eine Variante dieses Lösungsansatzes ist die Nutzung der AusweisApp und des „trusted Browser“ als ein „System on the stick“. In diesem Fall benötigt der Anwender keinen zweiten physischen PC. Der Anwender hat einen USB-Stick, auf dem sich ein komplettes Betriebssystem inklusive den vertrauenswürdigen Anwendungen „trusted Browser“ installiert sind. Sofern der Anwender die eID-Funktion nutzen möchte, muss er sein auf der Festplatte befindliche Betriebssystem, „untrusted Browser“ herunterfahren und von seinem USB-Stick den „trusted Browser“ mit Betriebssystem neu booten. Nun kann die vertrauenswürdige Anwendung, die AusweisApp in „trusted Browser“ ausgeführt werden.

Der Grundgedanke für die dritte Lösungsalternative ist die Aufteilung der Anwendungen in mehrere virtuelle Maschinen.

III. **Isolierung der Anwendungen in 3 virtuelle Maschinen:** Mit dem Einsatz von Virtualisierungsprogrammen können mehrere Domänen bzw. virtuelle Maschinen gleichzeitig auf einem physischen Computer betrieben werden. Jede Domäne mit der entsprechenden Anwendung, z.B. „trusted Browser“ hat sein eigenes Gast-Betriebssystem. Für die Gast-Betriebssysteme sieht es so aus, als würde die virtuelle Hardware real existieren. Die Zuteilung der Anwendungen zu diesen Domänen erfolgt entsprechend den Sicherheitsanforderungen. Zusätzlich zu der bereits vorgenommenen Trennung von „untrusted Browser“ und „trusted Browser“ soll „trusted Browser“ weiter aufgeteilt werden. Es werden die Funktionen der AusweisApp entsprechend den Schutzanforderungen in zwei Sicherheitsdomänen geteilt und mit einem Kommunikationskanal zur Aufrechterhaltung der eID-Funktion verbunden. Der Anwender startet „untrusted Browser“ als eine virtuelle

Maschine und kann dort potenziell nicht vertrauenswürdige Anwendungen nutzen. Bei Bedarf startet er zur Nutzung der Webanwendungen von Dienstleistern mit eID-Funktion die zweite virtuelle Maschine „trusted Browser“. Sofern er die eID-Funktion tatsächlich benötigt wird, führt der Anwender diese in der dritten virtuellen Maschine aus.

5.8. Bewertung der Lösungsalternativen

Für die praktische Durchführung der Bewertung der Alternativen soll das Punktebewertungsverfahren angewendet werden. Für die Erfüllung einer Anforderung werden Wertzahlen von 1 bis 10. Eine Punktewertzahl von 0 bedeutet bzgl. der ersten Anforderung, dass sich das erreichte Sicherheitsniveau nicht verbessert. Die Punktewertzahl 10 wird bei einer deutlichen Verbesserung des Sicherheitsniveaus vergeben. Bzgl. der Anforderungen zur Performance und zur Anwenderfreundlichkeit wird eine Punktewertzahl von 10 vergeben, wenn der Status Quo vor Anwendung des Lösungskonzeptes erhalten werden kann. Die Punktewertzahl 0 wird bei deutlicher Verschlechterung der Performance oder der Anwenderfreundlichkeit vergeben.

Erste Lösungsalternative: Separierung in zwei Benutzer;

Die **erste Anforderung,** bezüglich der Verbesserung des Sicherheitsniveaus, wird mit diesem Lösungsansatz nur teilweise erfüllt. Die Punktewertzahl wird mit 2 festgelegt. Die Anwendungen sind nach ihren Sicherheitsanforderungen separiert. Die eID-Funktion ist inklusive des „trusted Browsers“ in einem Benutzer von den nicht vertrauenswürdigen Anwendungen in einen zweiten Benutzer separiert und beide Benutzer sind gleichzeitig angemeldet. Ein Angriff auf die PIN mittels eines Benutzer-Trojaners, der nur über Benutzerrechte verfügt, ist durch die Separierung in zweiten Benutzer erschwert. Der Angreifer könnte die eingegebene PIN des Benutzers 2 so nicht mitlesen. Jedoch würde ein Angriff mit einem System-Trojaner, der über System-Administrationsrechte verfügt, die PIN-Eingabe von einem zweiten Benutzer mitlesen können. Windows und Linux-Betriebssysteme verlangen bei Aktionen, die System-Administratorenrechte benötigen, dass der Anwender den Zugriff auf die Systemadministration durch eine Benutzer-Interaktion bestätigt. Viele Anwender beantworten solche Anfrage ohne weitere Prüfung

mit ja. Der Systemtrojaner bekäme vollen Zugriff auf das System und könnte das gesamte System kompromittieren.

Die **Anforderung 2,** Beibehaltung der Anwenderfreundlichkeit, erhält aufgrund folgender Überlegungen die Punktewertzahl 7. Der Anwender muss einen zweiten Benutzer einrichten und diesen starten, bevor er die eID-Funktion im Rahmen einer Webanwendung nutzen kann. Er muss sich an die Regel halten, die Anwendungen nach ihrem Sicherheitsniveau mit zwei Benutzer zu bedienen.

Die **dritte Anforderung,** bezüglich der Performance, wird mit 10 angegeben. Die Verteilung der Anwendung auf zwei Benutzer lässt keine negativen Auswirkungen auf die Performance erwarten.

Zweite Lösungsalternative: Separierung in zwei PCs;

Der Erfüllungsgrad der **Anforderung 1,** Verbesserung des Sicherheitsniveaus, wird mit 5 festgelegt. Die Bewertung erfolgt auf Basis der folgenden Überlegungen: In „Untrusted Browser“ sind alle anderen Anwendungen, außer der AusweisApp separiert. Das bedeutet, dass erfolgreiche Angriffe auf den „untrusted Browser“ die Anwendungen „trusted Browser“ und die eID-Funktion nicht kompromittieren. Durch die Isolierung von „untrusted Browser“ sind nicht alle möglichen Angriffe erschwert.

Erfolgreiche Angriffe auf „trusted Browser“ können auch in dieser Lösungsvariante die eID-Funktion kompromittieren, denn im Webbrowser Mozilla Firefox ist im Jahr 2010 laut dem Bundesamt für Sicherheit in der Informationstechnik (BSI) nach dem Standard CVE[2] eine hohe Anzahl von Schwachstellen mit der Möglichkeit der Ausführung von Schadcode aufgetreten [16]. Browser-Funktionen wie zum Beispiel JavaScript, Java oder Flash erfordern, dass ein fremder Code auf dem Rechner der Besucher ausgeführt wird. Daraus entstehen weitere Sicherheitslücken. Das BSI beschreibt in seinem Lagebericht zur IT-

[2] „Common Vulnerabilities and Exposures“ (CVE) ist ein Standard zur Einteilung von Schwachstellen in IT-Systemen

Sicherheit eine Angriffsklasse, die ohne Nutzerinteraktion das System kompromittieren kann. Ein Beispiel sind die sogenannten „drive-by-exploits“, die bereits beim Betrachten einer Webseite ohne weitere Nutzerinteraktion Schwachstellen im Browser, in Browser-Plugins oder im Betriebssystem ausnutzen, um Schadsoftware wie Trojanische Pferde unbemerkt auf dem PC zu installieren. Gemäß einer Warnung des Bundesamtes besteht auch beim Besuch von als vertrauenswürdig anzusehenden Webseiten, die Gefahr einer Infektion des PCs und zwar über speziell manipulierte Werbebanner [16]. Daraus lässt sich schließen, dass die Aufteilung der Anwendungen in einen Computer für nicht vertrauenswürdige Anwendungen und einen zweiten Computer mit Verbindung in das Internet für die ausschließliche Nutzung von vertrauenswürdigen Anwendungen teilweise die erste Anforderung erfüllt. Dies bewirkt eine Verbesserung des Sicherheitsniveaus, durch die Separierung von „untrusted Browser“. Laut der oben genannten Warnung des Bundesamtes besteht auch bei vertrauenswürdigen Webseiten die Gefahr eines erfolgreichen Angriffes.

Die Erfüllung der **zweiten Anforderung**, dass die Nutzung der eID-Funktion anwenderfreundlich bleibt wird mit der Punktewertzahl 6 festgelegt. Der Anwender muss einen zweiten PC zur Verfügung haben, diesen einrichten und warten. Er muss sich an die Regel halten, die Anwendungen nach ihrem Sicherheitsniveau mit zwei PCs zu bedienen. In der Lösungsvariante „System on the stick“ muss er das komplette System herunterfahren, um anschließend vom Stick das System wieder hochzufahren.

Anforderung 3: Die Punktewertzahl zur Performance der Lösungsalternative erhält den Wert 6. Es werden keine negativen Auswirkungen auf die Performance der eID-Funktion erwartet. Jedoch muss zur Ausführung der Funktion zusätzlich ein zweiter PC gestartet werden und der Ausweisinhaber muss sich als neuer Benutzer anmelden. In der Lösungsvariante „System on the stick“ muss der Anwender das komplette System herunterfahren, um anschließend vom Stick das System „trusted Browser“ wieder hochzufahren.

Dritte Lösungsalternative: Isolierung der Anwendungen in 3 virtuelle Maschinen;
Der Erfüllungsgrad des Lösungsansatzes drei bezüglich der **ersten Anforderung,** „Verbesserung des Sicherheitsniveaus“ wird mit 10 festgelegt. Der Grad der Separierung ist bei diesem Lösungsansatz am höchsten und einem potenziellen Angreifer sind nach Anwendung des Lösungskonzeptes alle mit dem Risiko nicht gering bewerteten Angriffsvektoren erschwert.

In diesem Lösungsansatz wird die Virtualisierung als Konzept zur Teilung der physikalischen Hardware in mehrere logische Einheiten verwendet. Dies ermöglicht auf einem einzigen Computer mehrere virtuelle Maschinen unterzubringen, von denen jede potenziell unter einem anderen Betriebssystem läuft. Die Virtualisierungsprogramme trennen in Sicherheitsdomänen und ordnen diesen Domänen die Hardware-Ressourcen zu. Die Isolierung der eID-Funktion in einer Domäne und die Aufteilung der zwei Browser in jeweils weitere isolierte Domänen hat zur Konsequenz, dass sämtliche Angriffe erschwert sind und z.B. ein sogenannter Drive-by-exploit einer infizierten Website die eID-Funktion nicht gefährden würde. Gemäß dem Bericht zur Lage der IT-Sicherheit in Deutschland 2011 bieten Viren-Schutzprogramme keine ausreichende Sicherheit gegen Angriffe durch Drive-by-exploits. Virtualisierungstechniken können jedoch vor solchen Angriffen schützen [16]. Daraus lässt sich schließen, dass die erste Anforderung durch Lösungsansatz drei voll erfüllt wird.

Dieser Lösungsansatz erfüllt die **zweite Anforderung**, die AusweisApp bleibt nur bedingt anwenderfreundlich. Sie erhält die Punktewertzahl 7. Die Usability der eID-Funktion selbst ändert sich durch diesen Lösungsansatz nicht. Es entsteht jedoch zusätzliche Komplexität: Der Anwender muss drei virtuelle Maschinen starten. Er muss sich an die Regeln der getrennten Anwendungen halten.

Die Erfüllung der **Anforderung 3** erhält die Punktwertzahl 7. Bei der Bewertung der Performance wird von drei virtuellen Maschinen für die eID-Funktion, „trusted Browser“ sowie „untrusted Browser“ ausgegangen. Es können in Abhängigkeit von der Umsetzung negative Auswirkungen auf die Performance auftreten. Beispielsweise könnte bedingt durch die zusätzlich notwendige Segmentierung des Speichers oder durch die zusätzlichen

Kontext-Switchs zwischen den virtuellen Maschinen die Performance zusätzlich belastet werden.

5.9. Auswahl der besten Lösungsalternative als Entscheidung

Die gewichteten Punktewertzahlen (gewichtete Anforderung W multipliziert mit dem Erfüllungsgrad der Anforderung P) der einzelnen Anforderung können der folgenden Tabelle entnommen werden:

Ziel der Entscheidung:			Wahl der Lösungsalternative, die das Sicherheitsniveau bei Nutzung der eID-Funktion in dem gewählten Szenario durch Anwendung des Konzeptes unter Berücksichtigung der Nebeneffekte des Konzeptes (vgl. Anforderungen aus Abschnitt 4.5) maximiert.					
Unbedingte Anforderungen:			a. Hinreichendes Sicherheitsniveau und Potenzial das Schutzniveau zu verbessern b. Beibehaltung des Sicherheitskonzeptes c. Beibehaltung Funktionsumfang					
Nr.	**Anforderungen**	**Gewichtung**	**Lösungsansatz I**		**Lösungsansatz II**		**Lösungsansatz III**	
		W	P	W x P	P	W x P	P	W x P
1	Verbesserung des Sicherheitsniveau der AusweisApp bezüglich der Bedrohungen „PIN durch User-Angriff rauben“ sowie „Zugriff auf den nPA durch Zugriff auf den Benutzer PC“ durch Isolierung der Funktionen zur eID-Anwendung	50 %	2	1	6	3	10	5

2	Die Nutzung der eID-Funktion soll anwenderfreundlich bleiben	25%	7	1,75	6	1,5	7	1,75
3	Die Performance bei der Nutzung der eID-Anwendung soll gleich bleiben	25%	10	2,5	6	1,5	7	1,75
Erfüllungsgrad der Anforderungen		100%		5,25		6		8,5
Entscheidung								X

Tabelle 4: Nutzwertanalyse zur Entscheidung der Alternativen (vgl. [53])

Der Lösungsansatz III: **Isolierung der Anwendungen in drei virtuellen Maschinen,** „untrusted“ Browser; „trusted Browser“ und eID-Funktion ist aufgrund der Summe der gewichteten Erfüllungsgrade von 8,5 zur Erfüllung der Anforderungen am geeignetsten und dient daher als Grundlage für das Sollkonzept

5.10. Ausarbeitung

In diesem Abschnitt wird auf Basis des oben ausgewählten Lösungsansatzes: Isolierung der eID-Funktion, des „trusted Browsers“ und des „untrusted Browsers“ in 3 virtuelle Maschinen, der konzeptionelle Entwurf zur sichereren Nutzung der eID-Funktion ausgearbeitet.

5.10.1. Grundlagen Virtualisierung

Wesentlicher Ansatzpunkt der gewählten Lösung ist die Separierung der Komponenten der eID-Funktion auf dem Benutzer PC mit unterschiedlichen Sicherheitsanforderungen durch die Virtualisierung. Der Begriff Virtualisierung wird in der Informatik in vielen unterschiedlichen Anwendungsfällen in verschiedenen Ausprägungen eingesetzt. Hardware-Virtualisierung ist eine Abstraktion von physikalischer Hardware zu virtuellen Hardwarekomponenten, die in gleicher Weise genutzt werden können, wie ihr physikalisches Gegenstück. Auf virtueller Hardware können Betriebssysteme und Software in derselben Form wie auf physikalischer Hardware betrieben werden [54]. Virtualisierungsprogramme verfügen über einen Virtual Machines Monitor, der virtuelle Hardware, also einen Rechner bestehend aus CPU, RAM, Festplatte Netzwerkkarte, usw. simuliert. Der Ansatz eines virtual Machines Monitor hat gegenüber dem Ansatz mit einem

monolithischen Kernel den Vorteil, dass der Virtual Machines Monitor mit kleinen Programmen von wenigen tausend Codezeilen, die im Kernmodus direkt auf der Hardware laufen, auskommt. Je weniger Codezeilen, desto geringer die Wahrscheinlichkeit einen Fehler zu finden, der durch einen Angreifer ausgenutzt werden kann. Zhang et al. stellen in ihrer Studie den Unterschied wie folgt dar:

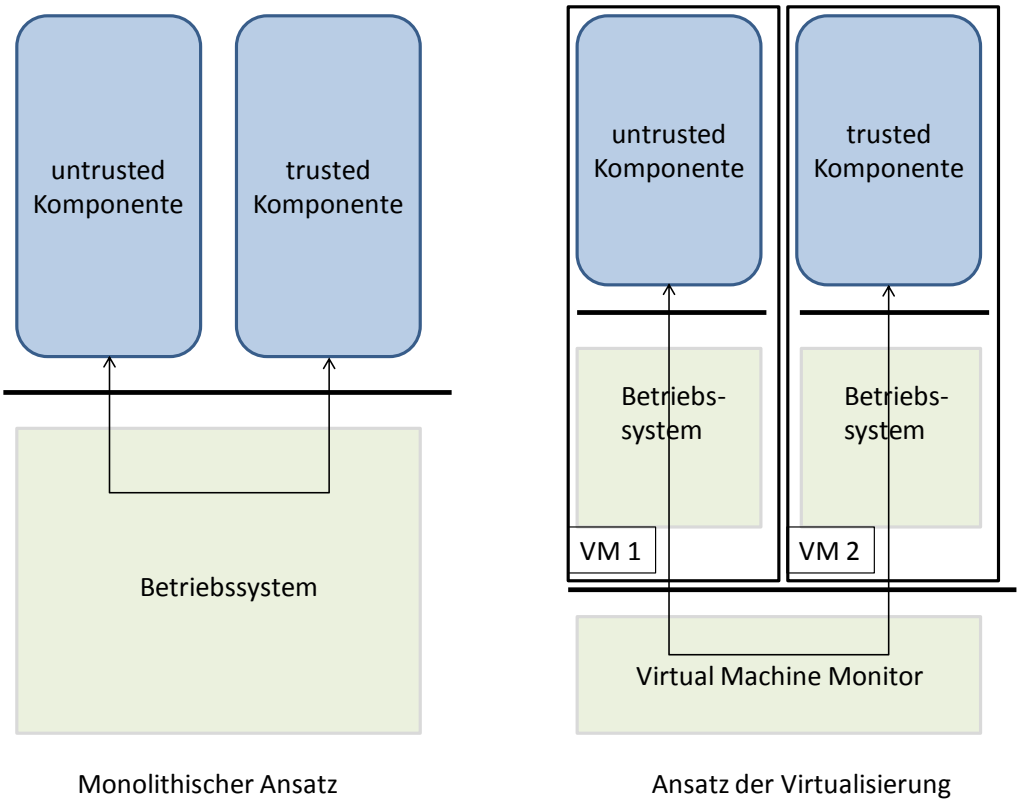

Abbildung 9: Monolithischer Ansatz und Ansatz der Virtualisierung [55]

Laut Zhang et al. hat es ein Angreifer im Vergleich zum Monolithischen Ansatz in der virtualisierten Umgebung schwerer von der untrusted Komponente aus, die trusted Komponente zu kompromittieren [55].

Tanenbaum geht in seinem Buch „moderne Betriebssysteme“ auf zwei Konzepte zur Virtualisierung näher ein. Diese sind der Typ-1-Hypervisor und die Paravirtualisierung. Der Typ-1-Hypervisor läuft direkt auf der Hardware, Voraussetzung hierfür sind CPUs mit Virtualisierungstechnologie. Das Gast-Betriebssystem in der virtuellen Maschine läuft im

Benutzermodus. Wenn ein sensitiver Befehl ausgeführt werden soll, findet ein Sprung in den Kern statt und der Hypervisor führt den Befehl aus [56]. Bei der Paravirtualisierung wird das Gast-Betriebssystem dahingehend modifiziert, das anstelle der Ausführung sensitiver Befehle der Hypervisor aufgerufen wird. Die Funktionsweise des Typ-1-Hypervisors und der Paravirtualisierung wird in folgender Übersicht dargestellt:

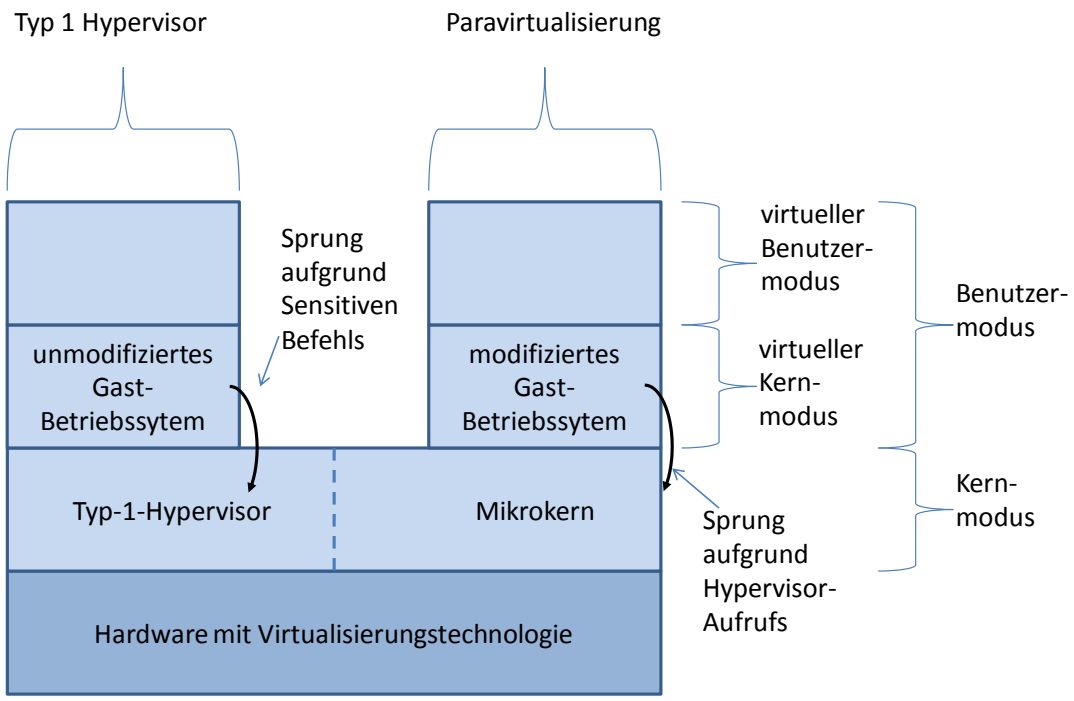

Abbildung 10: Hypervisor mit Virtualisierung und Paravirtualisierung [56]

Die gestrichelte Linie auf der Ebene des Hypervisors soll zeigen, dass es im Fall des Typ-1-Hypervisors dieser die sensitiven Befehle interpretiert. Im Fall des Mikrokerns führt dieser lediglich die Hypervisor-Aufrufe aus. Hier müssen die Befehle nicht mehr emuliert werden.

Moderne CPUs z.B. von Intel, enthalten hardware-seitig Funktionen zur Vereinfachung von Virtualisierungstechniken. Intel nennt diese Technik in Abhängigkeit der Architektur des Prozessors VT-x bzw. VT-i [57]. Intel VT-d ist eine Infrastruktur zur Virtualisierung von I/O-Geräten. Intel-VT-d erhöht die Sicherheit durch Begrenzung des Direct Memory Access (DMA) für die einzelne virtuelle Maschine. Dies wird durch DMA-Remapping

Hardware Logik erreicht. Hier wird die Speicheradresse der eingehenden DMA-Anfrage zu der korrekten physikalischen Speicheradresse mit Prüfung der Berechtigung des Zugriffs auf diesen Speicherraum ermittelt. Dadurch wird der physikalische Speicherraum jeder Sicherheitsdomäne von der Umwelt isoliert [58].

5.10.2. Separierung von „eID“, „trusted Browser“, „untrusted Browser“

Der gewählte Lösungsansatz III sieht die Isolierung von „untrusted Browser“, des „trusted Browsers“ und der eID-Funktion in drei verschiedene virtuelle Maschinen vor. Bezüglich des „untrusted Browsers“ bedeutet das, dass eine separate Domäne mit Betriebssystem, Internetzugang und einem Browser sowie den nicht vertrauenswürdigen Anwendungen in der ersten virtuellen Maschine installiert wird. Die AusweisApp benötigt zur Ausführung des elektronischen Identitätsnachweises sowohl den Browser, als auch den eID-Client. Im nächsten Schritt müssen alle sicherheitsrelevanten Funktionen der AusweisApp in der zweiten Domäne isoliert werden. Anschließend muss „trusted Browser“ inklusive Browser-Plugin getrennt von der eID-Funktion der AusweisApp in die dritte virtuelle Maschine isoliert werden. Die Zuordnung der Funktionen zu den virtuellen Maschinen eID und „trusted Browser“ soll anhand des Sequenzdiagramms aus Abschnitt 4.1.4 erfolgen.

Auf dieser Basis werden alle Funktionen, die unmittelbar der eID-Funktion zuzuordnen sind und damit sicherheitsrelevant sind in die Domäne mit der eID-Funktion isoliert:

a. Funktionen zur Interaktion mit dem Nutzer: Abfrage/Einwilligung Datenschutzerklärung, Abfrage Datengruppenauswahl/Bestimmung der Datengruppen und Abfrage PIN/Eingabe PIN
b. Funktionen zur Interaktion mit dem nPA: PACE, Terminalauthentifizierung, Chip-Authentifizierung, Aufbau geschützter Kommunikationskanal und Datengruppen übermitteln
c. Funktionen zum Verbindungssaufbau und –abbau: Aufruf eID-Client, Aufbau TLS, Nachricht eID-Funktion durchgeführt bzw. lösen der Kommunikationsverbindung

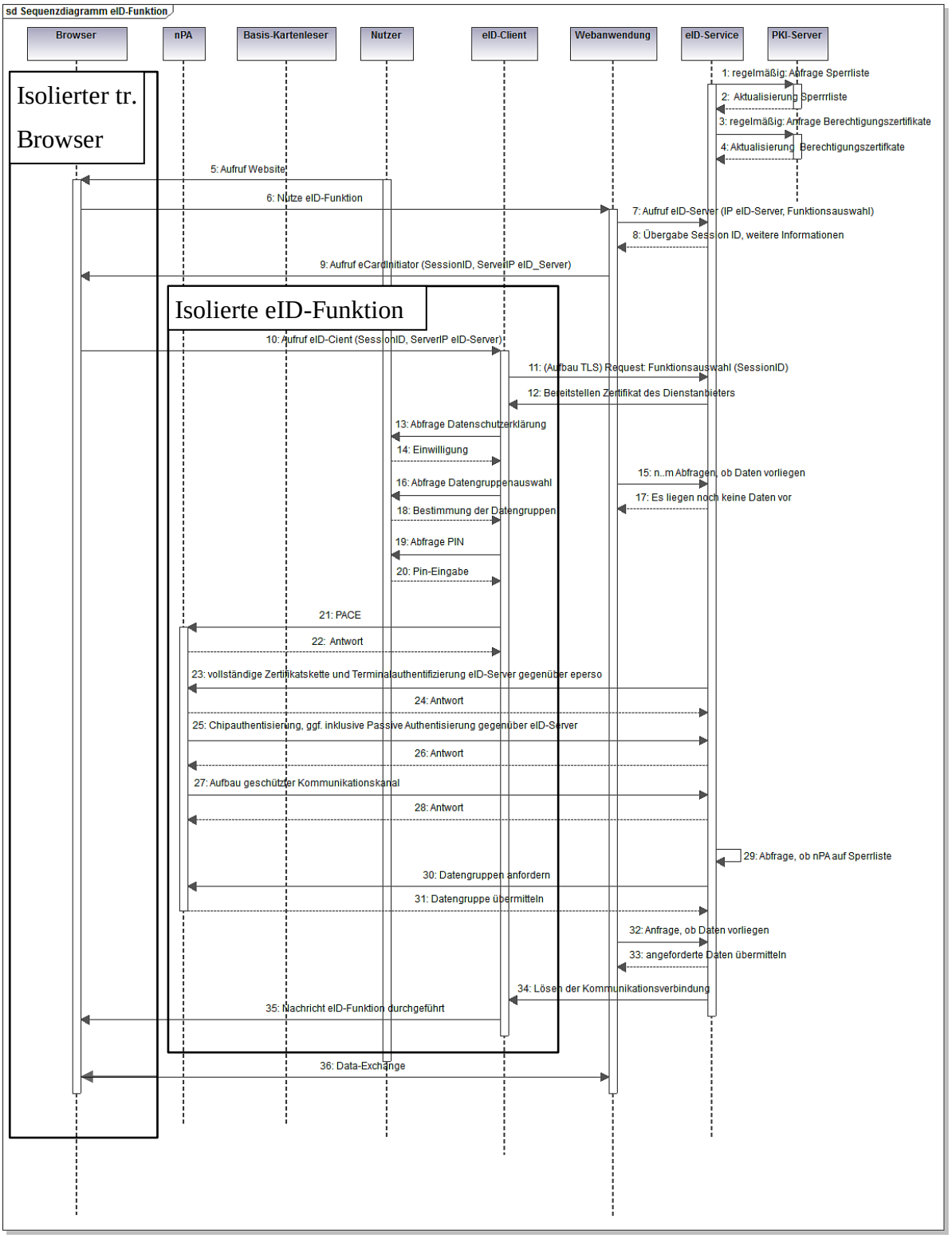

Abbildung 11: Sequenzdiagramm mit isolierten Anwendungen [2] [21] [23]

Alle Funktionen, die dem Browser inklusive Browser-Plugin zuzuordnen sind in die Domäne „trusted Browser“ isoliert:

a. Die Funktionen zur Interaktion mit der Webanwendung: Aufruf der Website, Nutze eID-Funktion und Data Exchange
b. Die Funktionen zur Interaktion mit der eID-Funktion: Aufruf eID-Client, Nachricht eID-Funktion durchgeführt

In Abbildung 11 sind die Funktionen, die in virtuelle Maschinen zu isolieren sind, mit einem Rahmen gekennzeichnet.

5.10.3. Kommunikation zwischen den virtuellen Maschinen

Die vollständige Isolierung von „trusted Browser“, von der virtuellen Maschine mit der eID-Funktion, ist nicht möglich. Wie in Funktion 10 „Aufruf eID-Client“ in Abbildung 11 dargestellt, ruft der eCardInitiator den eID-Client auf. Dabei werden Daten, z.B. SessionID, ServerIP eID-Server, an den eID-Client übergeben. Diese Kommunikation ist zur Ausführung der eID-Funktion notwendig. Daraus folgt, dass eine Kommunikation zwischen den beiden virtuellen Maschinen eingerichtet werden muss.

Grundsätzlich kann der Verbindungsaufbau für die notwendige Kommunikation aus beiden virtuellen Maschinen erfolgen. Würde der Verbindungsaufbau aus der virtuellen Maschine „trusted Browser“ erfolgen, könnte ein Schadprogramm über das Internet eingeschleust werden, die Verbindung zur AusweisApp aufbauen und diese kompromittieren. Aus diesem Grund sollte der Verbindungsaufbau von der virtuellen Maschine, die nicht mit dem Internet verbunden ist, an die weniger vertrauenswürdige virtuelle Maschine, die über einen Web-Browser mit der „Außenwelt“ verbunden ist, erfolgen. Der Kommunikationskanal soll nur für die Nutzungsdauer der eID-Funktion geöffnet sein.

6. Lösungskonzept anhand Qubes OS

Das Projekt Qubes OS, durchgeführt von Invisible Things Lab, hat zum Ziel, ein sicheres Betriebssystem für Desktop- und Laptop-Computer zu entwickeln. Ein wichtiger Baustein zur Verbesserung des Sicherheitsniveaus in diesem Betriebssystem ist die Isolierung von Anwendungen und Systemkomponenten mit unterschiedlichen Sicherheitslevels [59]. Qubes OS verwendet somit auch den im Rahmen dieser Arbeit favorisierten Lösungsansatz. Das Betriebssystem setzt auf der Linux-Distribution Fedora 14 auf.

In diesem Kapitel wird das bestehende Konzept mit dem Betriebssystem Qubes OS umgesetzt. Hierzu werden die Sicherheitsmerkmale von Qubes beschrieben und anschließend auf das bestehende Konzept abgebildet.

6.1. Sicherheitsmerkmale von Qubes OS

Qubes OS ist ein open source Betriebssystem, das Sicherheit durch Isolierung bietet. Es nutzt Virtualisierungstechnologie, um verschiedene Sicherheitsdomänen voneinander zu isolieren. Gemäß Rutkowska und Wojtczuk ist ein unberechtigter Zugriff von einer Sicherheitsdomäne auf die Andere nur mit hohem Aufwand durchführbar. Wird eine Sicherheitsdomäne mit Schadprogrammen infiziert, sind die anderen Sicherheitsdomänen nicht kompromittiert. Die Sicherheitsdomänen werden isoliert durch den Hypervisor Xen, die Virtualisierungstechnologie VT-d, die Trusted Execution Technology (TXT) und weitere Maßnahmen. Der Benutzer kann bestimmen, wie viele leichte- bzw. „wegwerf“-virtuelle Maschinen (light VMs) oder virtuelle Maschinen mit den entsprechenden Anwendungen des Benutzers (AppVMs) implementiert werden. Der Anwender kann z.B. Sicherheitsdomänen für persönliche Angelegenheiten, für Einkäufe im Internet oder für die Durchführung von Bankgeschäften installieren. Die Anwendungen in den Sicherheitsdomänen können genutzt werden, als würden sie auf dem lokalen Rechner laufen [60].

In Abbildung 12 wird die Architektur des Betriebssystems im Überblick dargestellt.

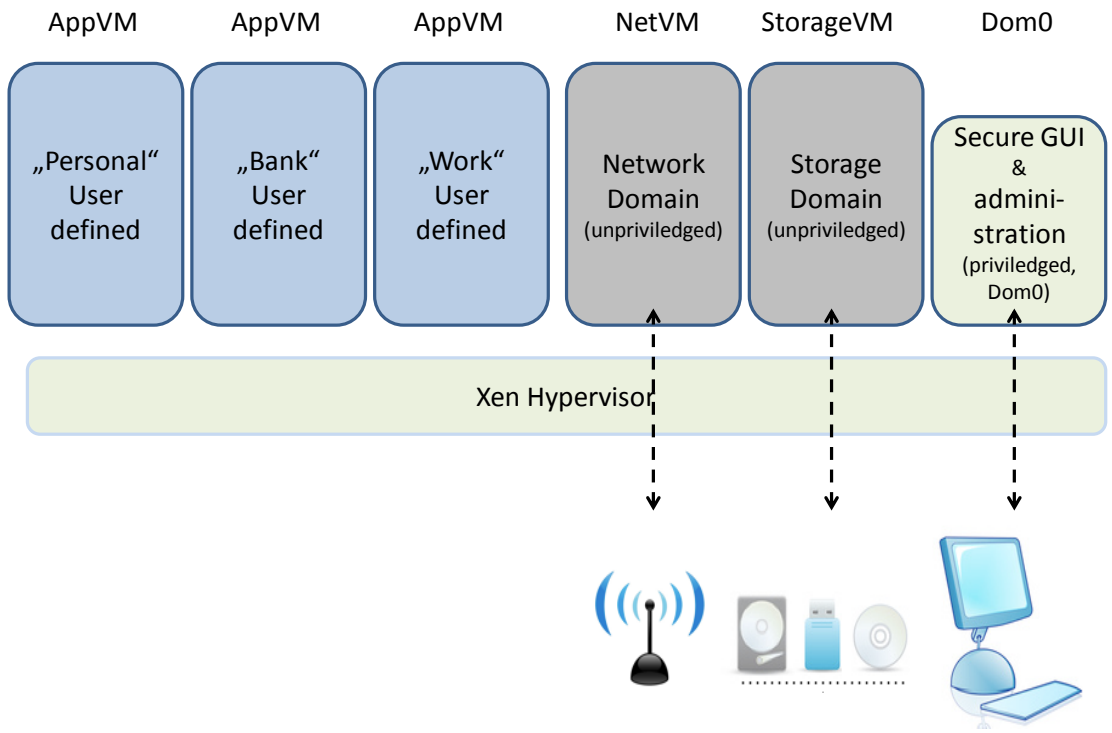

Abbildung 12: Architektur von Qubes OS im Überblick [60]

Das System basiert auf dem Hypervisor Xen und nutzt dessen Funktionen zur Paravirtualisierung. Dom0 ist die privilegierte virtuelle Maschine. Sie dient der Interaktion mit dem Hypervisor und startet und verwaltet die weiteren Sicherheitsdomänen. Ein erfolgreicher Angriff auf Dom0 oder den Hypervisor könnte das gesamte System kompromittieren. Der Anteil von sicherheitskritischem Code in Dom0 und im Hypervisor gilt als Trusted Computing Base. Qubes OS hat sich das Ziel gesetzt, diesen sicherheitskritischen Code auf ein Minimum zu reduzieren. Eine Umsetzung der Reduktion ist die Schaffung von System-Komponenten, wie zum Beispiel die unprivilegierte Network-domain für das Netzwerkmanagement bzw. die unprivilegierte Storage-Domain für z.B. Treiber als Sandboxen [60]. Diese Technik gewährleistet einen Speicherschutz über softwarebasierte Kontrollen. Ein Code-Modul kann nur noch auf Adressen eines festgelegten Bereiches (Sandbox) zugreifen [61]. Die Folge der Isolierung von System-Komponenten als Sandboxen ist, dass der C-Code des Netzwerkmanagements oder z.B. der Treiber nicht mehr in Dom0 ist. Laut Projekt liegt die Größe der aktuellen Trusted

Computing Base in der Größenordnung von Hunderttausend Zeilen C-Code [62]. Im Vergleich dazu enthalten Betriebssysteme, wie Windows, Linux oder Mac ca. 5 Millionen Codezeilen [63].

Dom0 enthält auch die GUI-Separation und hat direkten Zugriff auf die Grafikkarte und die Eingabegeräte, wie Tastatur und Maus. Das GUI-Subsystem besteht in Dom0 aus dem X Server, dem Windowsmanager sowie für jede VM aus den Appviewer. Jede AppVM hat entsprechend der Architekturbeschreibung jeweils einen Windowsmanager und einen reduzierten X Server. Die Stub-Anwendung Appviewer in Dom0 interagiert im Rahmen der Inter-Domain-Kommunikation mit dem Windowsmanager in der AppVM. Die GUI-Separation ermöglicht die gleichzeitige Anzeige der Fenster der einzelnen virtuellen Maschinen auf einem Bildschirm, unter Beibehaltung der Isolationseigenschaften. Die Fenster jeder virtuellen Maschine sind farblich gekennzeichnet, sodass der Anwender jederzeit erkennen kann, auf welcher virtuellen Maschine er gerade arbeitet.

Die AppVMs werden mit dem copy-on-write-Verfahren von einem zentralen Template beim Hochfahren der AppVM kopiert. Alle Dateien, die zum root-Dateisystem gehören, werden den AppVMs nur lesend zur Verfügung gestellt. Wird eine AppVM mit einem Schadprogramm im root-Dateisystem infiziert, wird beim nächsten Neustart der AppVM vom nicht infizierten Template neu geladen und das Schadprogramm ist damit eliminiert. Software-Updates für AppVMs werden zentral in zugehörigen TemplateVM durchgeführt.

6.2. Isolierung von „eID“, „trusted Browser“ und „untrusted Browser“

Entsprechend Abschnitt 5.10.2 werden drei virtuelle Maschinen nach ihren Sicherheitsanforderungen konzipiert. Die sicherheitsrelevanten eID-Funktionen sind in der neuen virtuellen Maschine namens „eID“ installiert. In der zweiten virtuellen Maschine namens „trusted Browser“ befindet sich der Browser mit dem eCard-Client-Initiator und dem Browser-Plugin. Alle weiteren Anwendungen sind in einer weiteren Maschine „untrusted Browser“ installiert. Bei Bedarf können weitere virtuelle Maschinen, zum Beispiel für das Online-Banking, usw. installiert werden. Abbildung 13 zeigt den Systemaufbau und die Zuordnung der Anwendungen/Funktionen im Überblick.

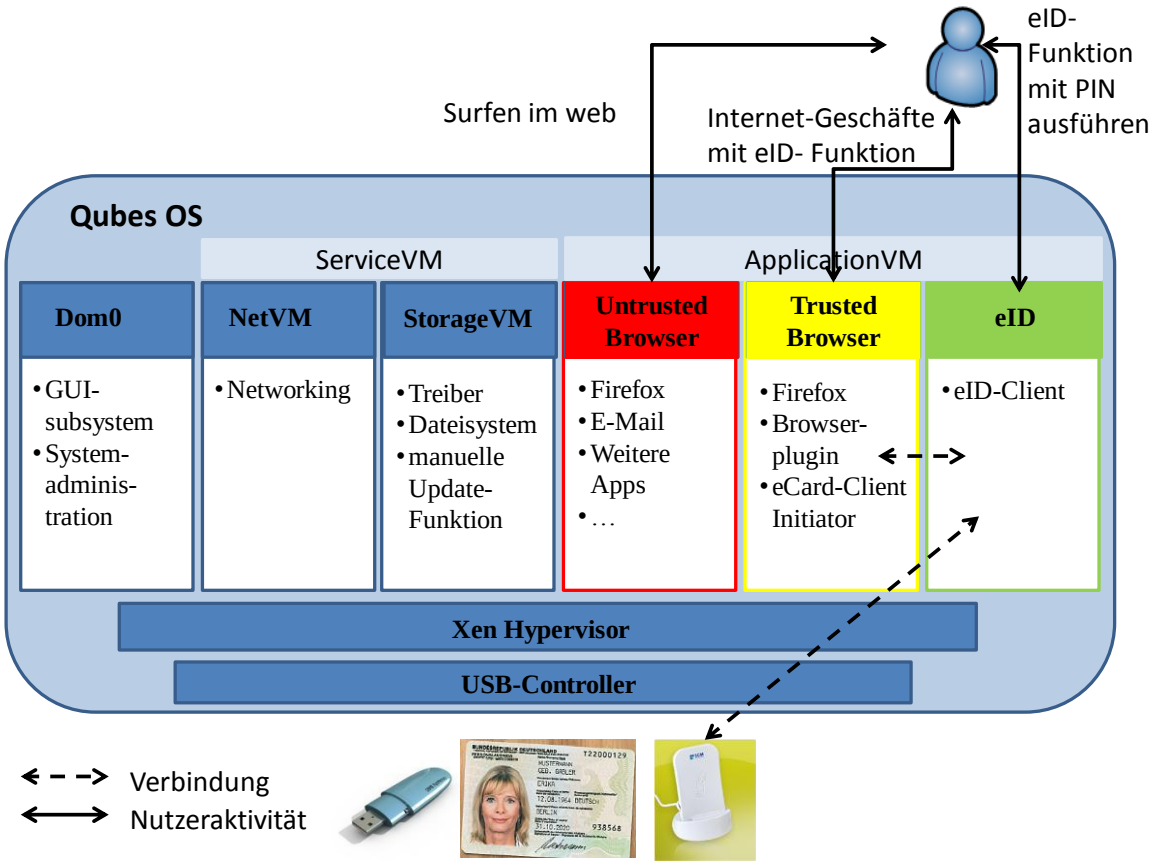

Abbildung 13: Sollkonzept mit Qubes OS

Eine „eID“ wird als virtuelle Maschine mit eigenem Template implementiert. Das Template liefert das root-Dateisystem. Die „eID“ hat nur Leserechte auf der TemplateVM. Die Implementierung von „eID“ mit eigener TemplateVM hat den Vorteil, dass die Anwendungen in „eID“ individuell zusammengestellt werden können bzw., dass „eID“ nur an individuell auswählbaren Updates teilnimmt. Bei einem erfolgreichen Angriff auf „eID“ ist diese nach Neustart, bezüglich dem root-Dateisystem, wieder im ursprünglichen Zustand. Ein weiterer Vorteil ist, dass die trusted computing base für „eID“ viel kleiner ist, weil sich keine weitere Anwendung im Sicherheitskontext der „eID“ befindet.

Für den Zugriff der „eID“ auf den Kartenleser, ist die Installation des Treibers des Basis-Kartenlesers (zum Beispiel: SCL011) notwendig. Um den Basis-Kartenleser in „eID“ verfügbar zu machen, muss dieser am USB-Eingang mit der PVUSB-Technik (ParaVirtualized USB) an „eID“ durchgereicht werden. Diese Technik erzeugt einen

virtuellen Host Controller in „eID“ und verbindet den Kartenleser mit ihr. Die Kommunikation erfolgt über URB (USB Request Block structure) [64]. Zur Ausführung der eID-Funktion muss „eID“ mit dem eID-Server kommunizieren und erhält daher einen Zugang zum Netzwerk.

Die zweite virtuelle Maschine „trusted Browser“ mit dem Browser inklusive dem eCardInitiator wird aufgrund der oben ausgeführten Überlegungen, ebenso mit eigener TemplateVM installiert.

6.3. Die Kommunikation zwischen „eID“ und „trusted Browser“

Entsprechend Abschnitt 5.10.3 muss eine Kommunikationsverbindung von der IP-Adresse 127.0.0.1 (Localhost) Port 18080 der virtuellen Maschine „trusted Browser“ zum Localhost, listening Port 18080 der Domain „eID“, eingerichtet werden. Die Inter-Prozess-Kommunikation zwischen zwei Prozessen auf einer Maschine muss zur Inter-Domain-Kommunikation zwischen zwei virtuellen Maschinen erweitert werden. Hierzu bietet Qubes OS folgende Ansätze [65]:

1. Network I/O
2. Vchan
3. Qubes RPC

Zu Network I/O:

Aus einer virtuellen Maschine kann die Kommunikation zu einer anderen Domäne mit einer virtuellen Netzwerk-Schnittstelle hergestellt werden. In Xen ist jedes virtuelle Interface in der virtuellen Maschine mit dem zugehörigen backend-Interface in der Dom0 verbunden [66]. Die bestehenden virtuellen Netzwerkschnittstellen, der beiden virtuellen Maschinen, können zu einem Netzwerk verbunden werden. Die Inter-Prozess-Kommunikation kann über eine Port-Weiterleitung in diesem virtuellen Netzwerk durchgeführt werden.

Zu Vchan:

Vchan wird in Qubes OS z.B. für die Grafikweiterleitung vom Appviewer benutzt (siehe Abschnitt 6.1). Vchan ist ein asynchroner Benachrichtigungsmechanismus über Event-Kanäle. Die Inter-Domain-Kommunikation zwischen Xen-Domänen wird durch geteilten Speicher hergestellt. Wenn ein Paar von Endpunkten - z.B. Ports - miteinander verbunden sind, kann eine „send-Operation“ ein Ereignis in der Zieldomäne auslösen [67]. Auf diese Weise lässt sich eine Port-Weiterleitung realisieren.

Zu Qubes-RPC:

Qubes-RPC ist auf der Basis von Qrexec realisiert. Der Aufruf entfernter Prozeduren erfolgt über qrexec-agent sowie qrexec-Daemon, die an den unix-sockets hören. Die Kommunikation zwischen agent und Daemon erfolgt über Vchan.

Zum Zeitpunkt der Konzeptionsphase war Vchan unzureichend dokumentiert und Qubes-RPC war noch nicht verfügbar. Die Inter-Domain-Kommunikation wird daher als network I/O, unter Nutzung der Port-Weiterleitung, über die virtuellen Netzwerkschnittstellen hergestellt.

Secure Shell (SSH) bietet die Möglichkeit in Netzwerken Daten von einem Port auf dem entfernten Host an einen anzugebenen Port auf dem lokalen Client verschlüsselt weiterzuleiten. SSH ist ein sicheres Protokoll der Transportebene [68]. SSH war ursprünglich ein sicheres Login-Protokoll, um eine Kommandozeile auf einem entfernten Rechner auszuführen. Das SSH-Tunneling mit dem SSH-Protokoll beinhaltet heute darüber hinaus Funktionen zur symmetrischen Verschlüsselung und zur Public Key-Authentifizierung des Clients und des Servers. Des Weiteren kann das Netzwerkprotokoll zwei Ports durch Sicherung eines Kanals miteinander verbinden. Bei Öffnung der Ports wird ein listening am entfernten Host etabliert. Die eingehenden Datenströme auf dem entfernten Host werden gekapselt, quasi in einen Umschlag getan und durch den Kommunikationskanal transportiert. Der Umschlag besteht aus Informationen (u.a. Adressen, Prüfsummen), die benötigt werden, um die Pakete zu befördern [69]. Dieses SSH-Tunneling wird für den Kommunikationskanal angewendet, wobei die

Verschlüsselung zu den bereits bestehenden Verschlüsselungen in der eID-Funktion redundant ist.

Entsprechend Abschnitt 5.10.3 wird der Verbindungsaufbau, aus der vertrauenswürdigeren virtuellen Maschine, „eID“ erfolgen. Daraus folgt, dass die virtuelle Maschine “trusted Browser“ als Server fungiert. Die Zugriffskontrolle erfolgt über die Zuordnung von „trusted Browser“ bei der Erzeugung der virtuellen Netzwerke und über das SSH-Protokoll. Im Rahmen des SSH-Protokolls erfolgt die Authentifizierung des Servers gegenüber dem Client mit einem RSA, DSA Schlüsselpaar. Die Schlüssel müssen dem Server vor dem Start des SSHd zur Verfügung gestellt werden, damit beim Authentifizierungsvorgang das passende Schlüsselpaar zur Verfügung steht. Der Client sendet beim Verbindungsaufbau eine mit dem öffentlichen Schlüssel verschlüsselte Challenge. Bei richtiger Antwort des Servers hat dieser bewiesen, dass er den richtigen Schlüssel hat. Der Client authentifiziert sich per Public-Key-Authentifizierung. Diese ermöglicht, dass sich der Client ohne Benutzerinteraktion auf dem SSH-Server, „trusted Browser“, einloggen kann [70].

Die zwei virtuellen Maschinen „eID“ und „trusted Browser“ können durch eine Direktverbindung oder mittels Bridge miteinander verbunden werden. Bei einer Direktverbindung wird die Netzwerkschnittstelle direkt zwischen Sender und Empfänger-Domäne erzeugt. Bei einer Bridge werden die Netzwerkschnittstellen zwischen der Bridge und den jeweiligen Domänen erzeugt. Eine Bridge kann mehrere Netzwerksegmente in ein logisches Netzwerk verbinden und dabei das Ziel der Datenpakete prüfen und ggf. Datenpakete fallen lassen. Der Einsatz einer Bridge hat den Vorteil, dass von der „eID“ neben „trusted Browser“ zu weiteren virtuellen Maschinen ein Kommunikationskanal aufgebaut werden kann. Daher wird die Verbindung zwischen den virtuellen Maschinen über eine „Backend Bridge“ hergestellt. Abbildung 14 zeigt die Einbindung der Backend Bridge zusammen mit der FirewallVM und der netVM in das virtuelle Netzwerk.
Jede AppVM, für die networking zugelassen wurde, hat ein eigenes virtuelles Netzwerk. Die FirewallVM blockiert jede Kommunikation zwischen den virtuellen Maschinen. Dom0 enthält aus Sicherheitsüberlegungen keine Netzwerkverbindung [71]. Die Bridge wird aus Dom0 erzeugt und eine IP-Adresse zugewiesen. Den virtuellen Maschinen „trusted

Browser“ und „eID“ werden jeweils durch paarweises erzeugen der ethx/vifx.0-devices über die Backend-Bridge Netzwerkschnittstellen hinzugefügt.

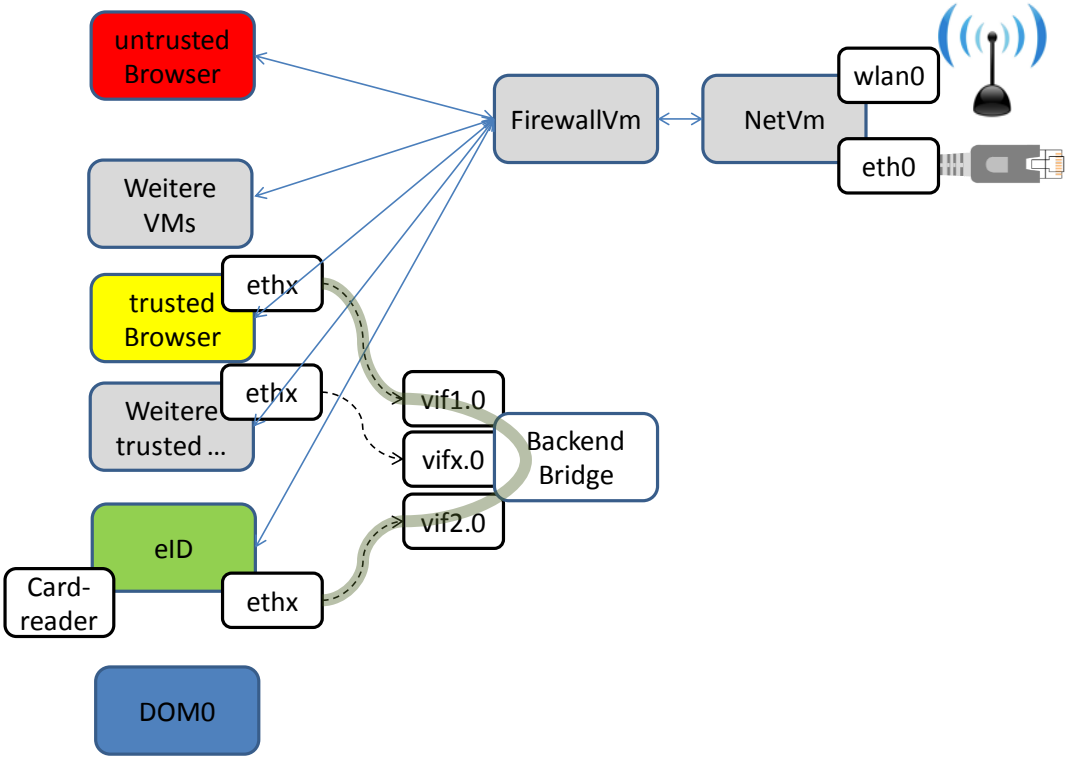

Abbildung 14: Kommunikationskanal zwischen „eID“ und „trusted Browser“

Das Sequenzdiagramm in Abbildung 15 zeigt den Funktionsablauf mit Fokussierung auf die Steuerung des Kommunikationskanals. Der Anwender will die eID-Funktion nutzen und startet, sofern noch nicht geschehen, die „eID“ und „trusted Browser“ (Sequenzdiagramm: Funktionen 1-4, 7 und 8). Für den Anschluss des Kartenlesers wird mit einem xm-Kommando ein virtueller USB-Host-Controller in „eID“ erzeugt und der Kartenleser „eID“ zugänglich gemacht (Sequenzdiagramm: Funktionen 5-6). Mit einem weiteren xm-Kommando wird aus Dom0 eine virtuelle Netzwerkkarte auf der Backend Bridge erzeugt und „trusted Browser“ zugänglich gemacht und aktiviert.

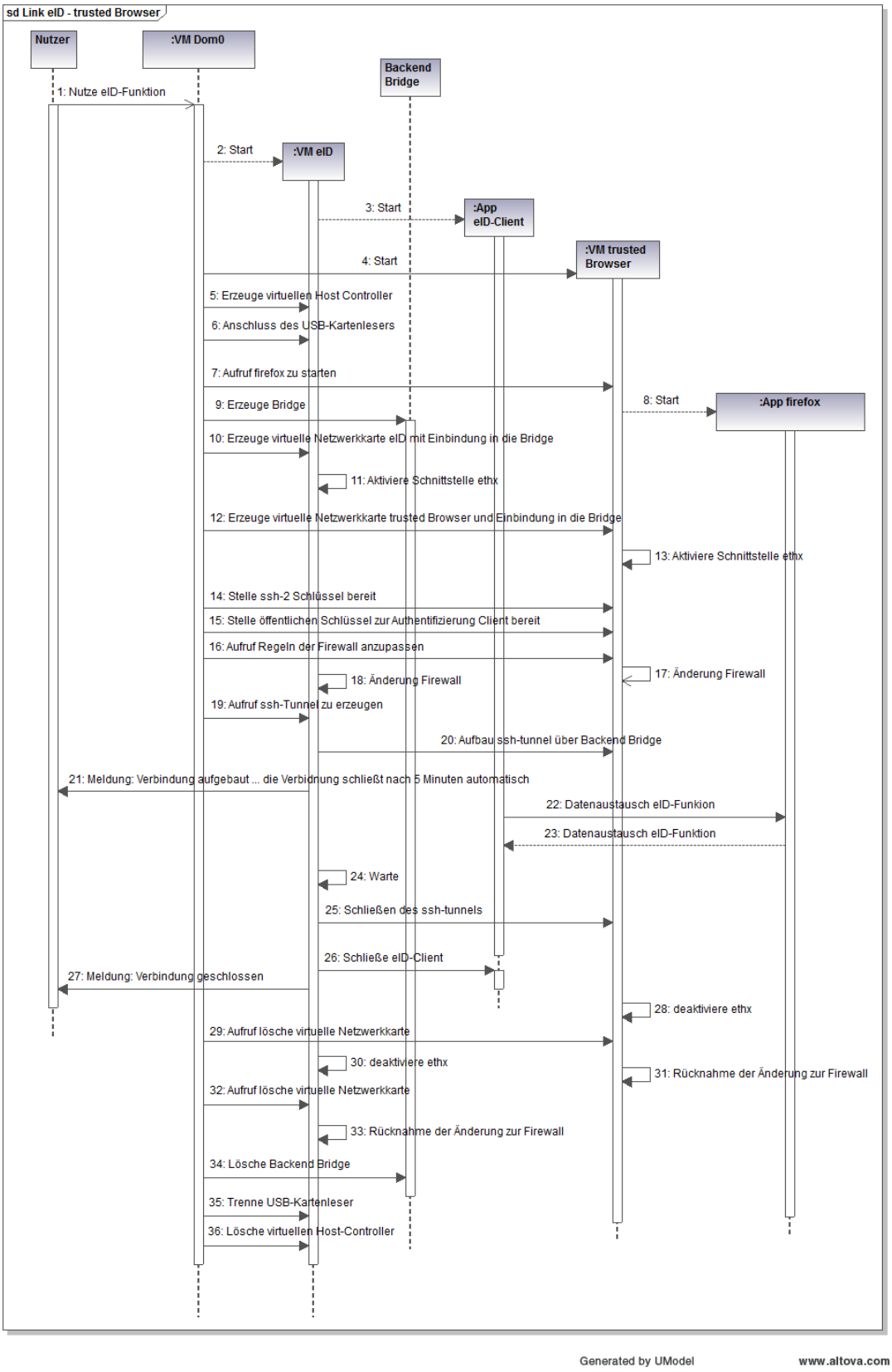

Abbildung 15: Kommunikation zwischen „eID“ und „trusted Browser“

Neben dem Namen und der Identifikationsnummer der virtuellen Maschine „trusted Browser“ muss die Bridge und das Backend im o.g. Kommando angegeben werden. Gleichermaßen wird in „eID“ eine virtuelle Netzwerkkarte erzeugt und aktiviert (Sequenzdiagramm: Funktionen 9-13). Für die gegenseitige Authentifizierung werden die notwendigen Schlüssel in den beiden Domänen bereitgestellt und die Firewall in „trusted Browser“ und „eID“ angepasst (Sequenzdiagramm: Funktionen 14-18). Die Verbindung aus „eID“ wird mit einem SSH-Tunnel aufgebaut (Sequenzdiagramm: Funktionen 19-20). Nach Schließen des SSH-Tunnels werden die paarweise erzeugten ethx-vifx.0-devices und anschließend die Bridge gelöscht. Zuletzt wird der Zugriff auf den Kartenleser und der virtuelle USB-Host-Controller gelöscht (Sequenzdiagramm: Funktionen 25-36).

Eine Implementierung kann auf Basis der Standardbefehle eines Bash-Scripts erfolgen. Zum Komfort des Anwenders kann der Start über das Kde-Menü erfolgen.

7. Umsetzung im Prototyp

Das im vorigen Kapitel erarbeitete Lösungskonzept - wird im Rahmen der Diplomarbeit - in eine prototypische Implementierung umgesetzt. Hierzu werden in den folgenden Abschnitten die notwendigen Anpassungen des Lösungskonzeptes aus Kapital 6, die Vorbereitung des Systems und danach die Implementierung des Prototyps beschrieben.

7.1. Umsetzungskonzept Prototyp

Die Version Qubes OS Beta 1 verwendet den Xen Hypervisor in der Version 3.4.5. In dieser Xen-Version ist das in Abschnitt 6.2 beschriebene PVUSB nicht verfügbar. Laut Xen wiki ist das Durchreichen des am USB-Anschluss befindlichen Basis-Kartenlesers an „eID“ bei Nutzung der Paravirtualisierung erst mit XEN Version 4.0 möglich [72]. Dadurch kann der Basis-Kartenleser nicht an „eID“ durchgereicht werden und ist nur in Dom0 verfügbar. Daraus folgt, dass der Treiber des Kartenlesers und damit auch die eID-Funktion in Dom0 liegen müssen. Diese Änderung macht eine weitere Anpassung notwendig. Der eID-Client muss während der eID-Funktion mit dem eID-Server kommunizieren, daher muss in Dom0 eine Verbindung zum Internet eingerichtet werden. Das Beta 2 Release mit Xen 4.1 und weiteren Optimierungen wurde am 19.09.2011 veröffentlicht [73]. Aufgrund des bestehenden Terminplanes dieser Diplomarbeit, erfolgt die prototypische Implementierung auf Basis des Qubes Beta 1 Releases vom 11.04.2011. Eine Anpassung des Umsetzungskonzepts zum Prototyp an das ursprüngliche Lösungskonzept ist ohne Schwierigkeiten möglich.

Die AusweisApp setzt auf der 686er Architektur mit 32 bit System auf. Qubes OS setzt auf der Linux-Distribution Fedora 14 und einer X86_64-Architektur auf [74]. Qubes OS verfügt über einen 64-bit Browser. Die AusweisApp benötigt einen 32-bit-Browser. Aufgrund dieser Abhängigkeit zur AusweisApp (eCard-client-Inititiator) muss Firefox in der Version 3.0 oder 4.0 installiert werden. Die AusweisApp wird zurzeit vom Bundesministerium des Innern nur als Debian-Paket, Ubuntu- oder OpenSuse-Paket zur

Verfügung gestellt[3]. Die Installation-Scripts der AusweisApp können nicht verwendet werden, da die Dateistruktur von Qubes OS von dem in Debian üblichen Dateisystem abweicht. Die Installationsroutine und die Aktualisierungsfunktion können daher nicht genutzt werden.

Da es sich um einen Prototypen handelt und keine weiteren „Trusted VMs“ benötigt werden, wird zwischen „eID“ und „trusted Browser“ anstatt einer Bridge, eine direkte Verbindung eingerichtet. Die Dauer der Verbindung wird auf 5 Minuten begrenzt. Auf der CD im Materialanhang befindet sich eine Installationsanleitung, die die durchzuführenden Arbeiten zur Installation des Systems ausführlich beschreibt. Abbildung 16 zeigt das Umsetzungskonzept im Überblick, nach Berücksichtigung der notwendigen Anpassungen:

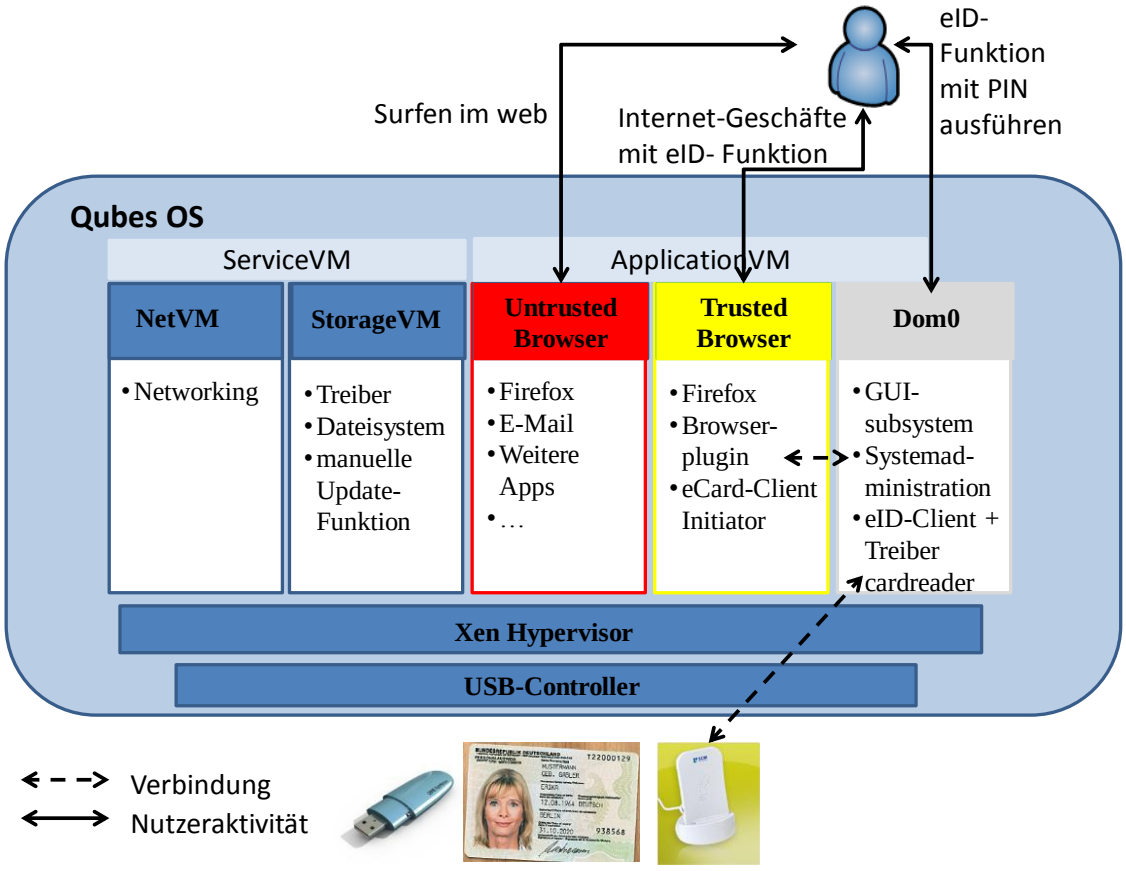

Abbildung 16: Umsetzungskonzept Prototyp

[3] https://www.ausweisapp.bund.de/pweb/filedownload/download_pre.do

7.2. Vorbereiten des Systems

Das Betriebssystem Qubes OS wird von der DVD im Materialanhang installiert[4]. Bei der Installation werden die drei in Abbildung 16 gezeigten ApplicationVMs eingerichtet. Für die Kommunikation mit dem Basis-Kartenleser wird der Smart-Card-Daemon pcsc-lite[5] verwendet. Der Treiber scl011_2.06_linux_32bit in Dom0 unterstützt den Basis-Kartenleser von SCM Microsystems. Der pcsc-Daemon erkennt den Kartenleser und kann nun mit dem RFID-Chip kommunizieren.

Die AusweisApp in der Version AusweisApp_010300_i686.deb wird aus dem Internet[6] heruntergeladen und auf einem PC mit Debian-Betriebssystem aktualisiert. Die aktualisierten Dateien der AusweisApp werden auf den PC mit dem Betriebssystem Qubes OS in Dom0 kopiert. Die nicht ausführbare Installationsroutine wird durch die manuellen Zuordnungen der Rechte für die auszuführenden Dateien und die manuelle Erstellung der symbolischen Links ersetzt. Die Details zur Implementierung befinden sich in der Installationsanleitung auf der CD im Materialanhang.

In die AppVM „trusted Browser“ mit der Farbe Gelb wird der Browser *firefox-3.0.tar.bz2* installiert[7]. Im Firefox wird die Erweiterung *eCardClientExt_ffxx_Lin32.xpi* und das Browser-Plugin *vnd.ecard-client* eingerichtet.

7.3. Implementierung Prototyp

Entsprechend Kapitel 6, soll die Implementierung über Standardbefehle eines Bash-Scripts erfolgen. Im folgenden Abschnitt werden die Realisierungen der Scripte dargestellt. Anschließend werden die Integration der Scripte in das Kde-Menü und der Ablauf der EID-Funktion im Prototyp dargestellt.

---

[4] http://qubes-os.org/Home.html

[5] http://pcsclite.alioth.debian.org

[6] https://www.ausweisapp.bund.de/pweb/filedownload/download_pre.do

[7] *ftp://ftp.mozilla.org/pub/firefox/releases/3.0/linux -i686/de/firefox-3.0tar.bz2*

7.3.1. Realisierung der Scripte

Zuerst wird das Shell-Script */usr/bin/domscriptstart* zur Steuerung des Kommunikationskanals zwischen den beiden virtuellen Maschinen dargestellt.

Netzwerkzugang Dom0:

Laut der Architekturbeschreibung von Qubes OS enthält Dom0 aus Sicherheitsüberlegungen keine Netzwerkverbindung [60]. Zur Ausführung der eID-Funktion muss der eID-Client jedoch mit dem eID-Server kommunizieren. Mit dem Qubes-eigenen Script *qvm-dom0-networking-via-netvm up* wird Dom0 das Netzwerk zugänglich gemacht. Manche Operationen in diesem Script benötigen root-Rechte. Um diese Rechte einzuräumen wird im Prototyp das *setgid* verwendet.

Netzwerkkonnektivität:

Der weitere Ablauf des Scripts ist wie folgt: Zuerst müssen die virtuellen Netzwerke der Domänen miteinander verbunden werden. Es wird eine Bridge erzeugt. Mit dem Befehl *xm network-attach* wird eine virtuelle Netzwerkkarte auf der virtuellen Maschine „trusted Browser“ erzeugt. Zuvor ist die Ermittlung der Nummer von „trusted Browser“ erforderlich. Es wird eine direkte Verbindung zwischen Dom0 und „trusted Browser“ hergestellt, indem die virtuelle Netzwerkkarte deaktiviert und die Bridge gelöscht werden und anschließend die Netzwerkschnittstelle mit der IP-Adresse 10.99.0.1 in Dom0 eingerichtet wird. Durch Ausführen des Qubes-eigenen Scripts *qvm-run* können aus Dom0 in „trusted Browser“ Kommandos ausgeführt werden. Mit dem Befehl *qvm-run –u root personal ´ifconfig eth1 10.99.0.2 netmask 255.255.0.0 up´* wird die neue Netzwerkschnittstelle in „trusted Browser“ konfiguriert und aktiviert.

Gegenseitige Authentifizierung:

Nachdem das logische Netzwerk eingerichtet ist, folgen die Kommandos zur gegenseitigen Authentifizierung und die Regelanpassungen der Firewall. Zur Bereitstellung des passenden Schlüsselpaares zur Server-Authentifizierung wird per echo-Befehl, der private Schlüssel in eine Datei auf dem Client kopiert und per *qvm-run*-Befehl auf dem Server abgelegt. Diese Befehle sind entsprechend für alle SSH2-Schlüssel durchzuführen. Zur Authentifizierung des Clients gegenüber dem Server verwenden wir eine public-Key-

Authentication. Die Bereitstellung der Schlüssel erfolgt nach dem oben gezeigten Verfahren.

Firewall:
Die Einstellungen der Firewall werden wie folgt angepasst: Es wird eine Regel an erster Stelle der Regelkette INPUT eingeführt, die alle von der IP-Adresse 10.99.0.1 in „trusted Browser“ eingehenden Pakete akzeptiert. Die Erweiterung der Regelkette INPUT erfolgt mit dem Befehl: *iptables –I INPUT 1 –s 10.99.0.1 –j ACCEPT.*

Tunneling:
Schließlich wird der SSH-Tunnel mit dem Befehl *ssh 10.99.0.2 –l root –v –R 127.0.0.1:18080:127.0.0.1:18080 sleep 300* aufgebaut. Die Verbindung wird mit Zeitablauf von 300 Sekunden automatisch wieder geschlossen.

Verbindungsabbau:
Es werden die Netzwerkschnittstellen in „trusted Browser“ und in Dom0 deaktiviert und die virtuelle Netzwerkkarte ethx/vifx.0 mit dem Befehl *xm network-detach* wieder entfernt. Anschließend wird die Bridge gelöscht und mit Aufruf des Scripts *qvm-dom0-networking-via-netvm down* der Zugang von Dom0 zum Netzwerk deaktiviert.

Mit dem Befehl *flock* kann verhindert werden, dass das Script während eines Laufes ein zweites Mal gestartet werden kann.

Das zweite Script */usr/bin/domscripttrustedbrowser* startet mit einem qvm-run-Befehl zuerst die virtuelle Maschine „trusted Browser“ und anschließend den Firefox.
Abschließend erhalten die Scripte mit dem *chmod*- und *chown*-Befehlen die notwendigen Berechtigungen.

7.3.2. Integration der Scripte in das Kde-Menü

Die Eintragungen zur Bedienung der eID-Funktion befinden sich im Navigationsbaum unter der Bezeichnung *other:*

eID-Funktion: Das Start-Script */opt/olsc/Ausweisapp/bc.sh* der AusweisApp ist im Rahmen der Vorbereitung bereits in das Kde-Menü eingetragen. Mit dem Menü-editor wird der Name in *AusweisApp Start eID* angepasst.

„trusted Browser“: Die Einbindung der ausführbaren Startdatei */usr/bin/domscripttrustedbrowser* in das Kde-Menü erfolgt durch die Datei */usr/share/applications/ausweisapp-2-start-trusted-Browser.desktop*. Mit dieser Datei werden die Startoptionen festgelegt.

Inter-Domain-Kommunikation: Ebenso erfolgt die Einbindung der ausführbaren Startdatei */usr/bin/domscriptstart* durch die Datei */usr/share/applications/ausweisapp-3-Kommunikationstarte-eID-trusted-Browser.desktop* in das Kde-Menü.

7.3.3. Ablauf der eID-Funktion im Prototyp

Die Anwendung der eID-Funktion über das Kde-Menü erfolgt durch Ausführung der folgenden Menüpunkte im Hauptmenü *Other*:

1. *AusweisApp Start trusted Browser*

   Aufruf einer Webseite eines Dienstanbieters und Entscheidung des Anwenders die eID-Funktion zu nutzen.
2. *AusweisApp Start eID*
3. *AusweisApp Start Link eID-trusted Browser*

   Der Anwender prüft das Berechtigungszertifikat und führt die eID-Funktion unter Eingabe der PIN aus. Der Kommunikationskanal schließt automatisch nach fünf Minuten. Bei Bedarf kann die Verbindung mehrmals genutzt werden.

8. Evaluierung

Anhand des Prototyps wird in diesem Kapitel überprüft, inwieweit das Lösungskonzept die in den Abschnitten 5.4 und 5.5 definierten Anforderungen erfüllt. Zuerst werden die gewichteten Auswahlkriterien und anschließend die Anforderungen, die unbedingt zu erfüllen sind, evaluiert. Anschließend wird erneut eine Risikoanalyse durchgeführt und der Sicherheitsgewinn bewertet. Das Kapitel schließt mit der Darstellung, was im Rahmen der Diplomarbeit nicht betrachtet werden konnte.

8.1. Anforderung 1: Verbesserung des Sicherheitsniveaus

In Abschnitt 4.4 wurde im Rahmen der Risikoanalyse Abbildung 8 folgende Angriffsvektoren mit der Risikoeinstufung „nicht gering“ identifiziert:

1: Kenntnis der PIN erlangen

- 1.2: PIN durch Userangriff rauben
  - 1.2.1: Inhaber des elektronischen Personalausweises zur Preisgabe der PIN verleiten
  - 1.2.3: Ausspähen der PIN (Sniffing mittels Keylogger)

2: Zugriff auf den nPA

- 2.3: Zugriff über den Benutzer PC
  - 2.3.2: Angriff auf den Browser (inkl. Browser-Plugin)

In den nächsten beiden Abschnitten wird zuerst der Angriffszweig „Ein Angreifer kann durch einen Userangriff die Kenntnis der PIN erlangen“ dargestellt.

8.1.1. Angriffsvektor: Ausspähen der PIN

Das Angriffsszenario: Der Angreifer infiziert den PC mit einem Schadprogramm und schreibt alle Eingaben mit. Die PIN wird bei Nutzung der eID-Funktion eingegeben. Die Log-Datei wird per E-Mail mit weiteren Informationen an den Angreifer gesendet. Dieser Angriffsvektor könnte sich gegen Dom0 oder den Hypervisor richten. Entsprechend den in Abschnitt 6.1 beschriebenen Eigenschaften von Dom0 und Hypervisor, wird ein erfolgreicher Angriff auf die Trusted Computing Base mit gering eingestuft. Im Rahmen der Evaluierung dieses Angriffsvektors wird dieser Fall durch einen Keylogger in „trusted

Browser“ simuliert. Der Evaluierung liegt die Annahme zu Grunde, dass über das Surfen vertrauenswürdiger Websites „trusted Browser“ mit einem Keylogger infiziert wurde. **Durchführung Evaluierung:** Für die Durchführung wird der pykeylogger-1.2.1[8] verwendet. Die Evaluierung erfolgt wie in dem Abschnitt 2.3 beschriebenem Szenario der Altersverifikation bei einer Online-Videothek auf einem Test- und Demonstrationssystem für den neuen Personalausweis. Die Durchführung beginnt mit dem Start des „trusted Browsers“ und dem eID-Client über das Kde-Menü. In „trusted Browser“ wird der Keylogger gestartet. Abbildung 17 zeigt, dass auf der Website des Testsystems zunächst die Zugangsdaten zur Authentifizierung eingegeben werden müssen.

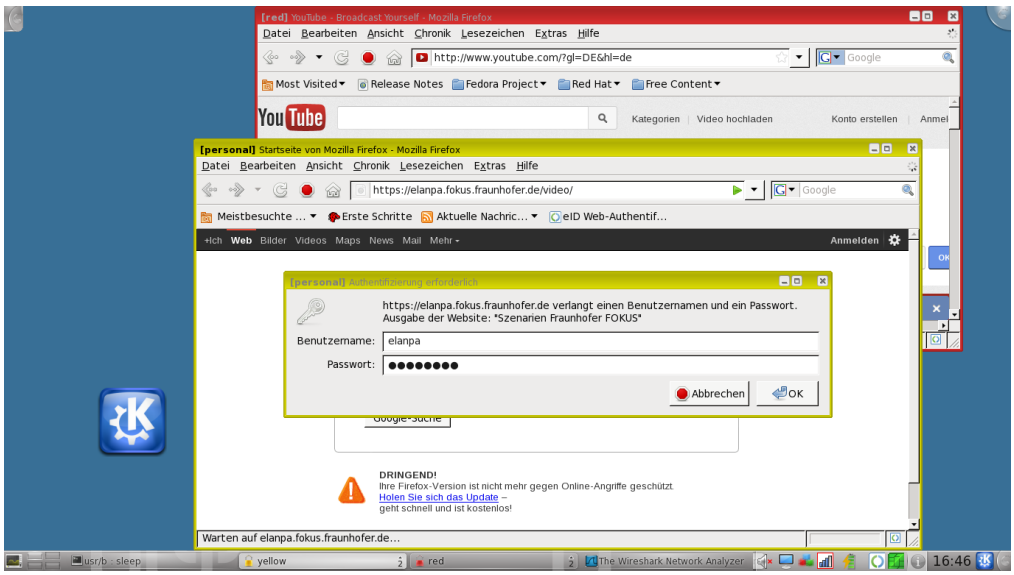

Abbildung 17: Authentifizierung Testsystem

Dann muss zur Freigabe der Filme ab 18 Jahren die Volljährigkeit des Nutzers über die eID-Funktion nachgewiesen werden. Zur Nutzung der eID-Funktion wird der Kommunikationskanal zwischen Dom0 und „trusted Browser“ aufgebaut. Nun wird auf dem Testsystem der elektronische Identitätsnachweis wie folgt ausgeführt. Es erscheint zunächst das Berechtigungszertifikat des Dienstanbieters:

[8] http://downloads.sf.net/project/pykeylogger/pykeylogger/1.2.1/pykeylogger-1.2.1_src.zip.

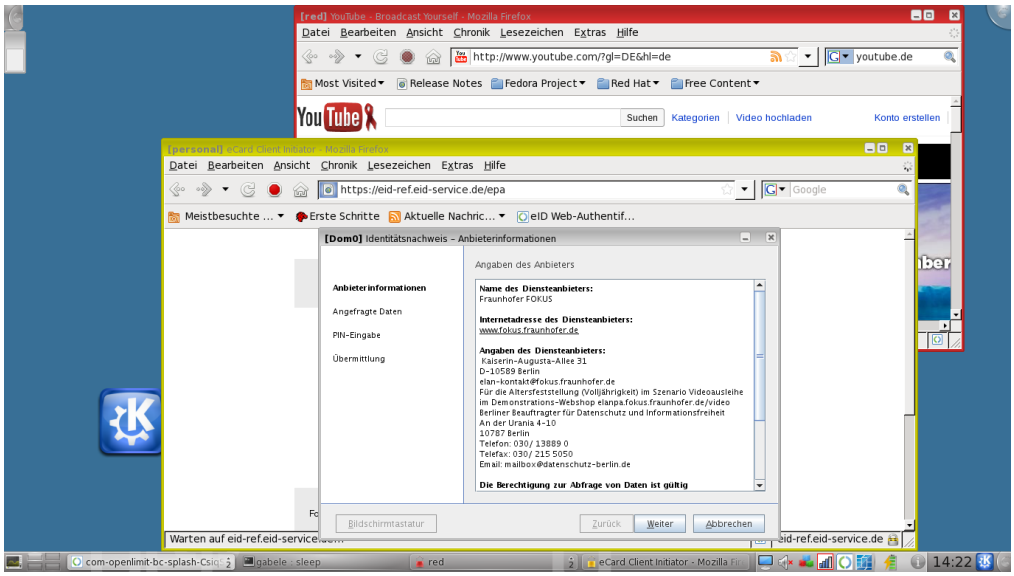

Abbildung 18: Berechtigungszertifikat des Dienstanbieters

Nach Auswahl und Bestätigung der Datengruppen erfolgt die Aufforderung zur Eingabe der PIN. Die Farbe des Eingabefensters ist grau und signalisiert dem Nutzer, dass er in Dom0 arbeitet. Nach der PIN-Eingabe werden die Daten übermittelt.

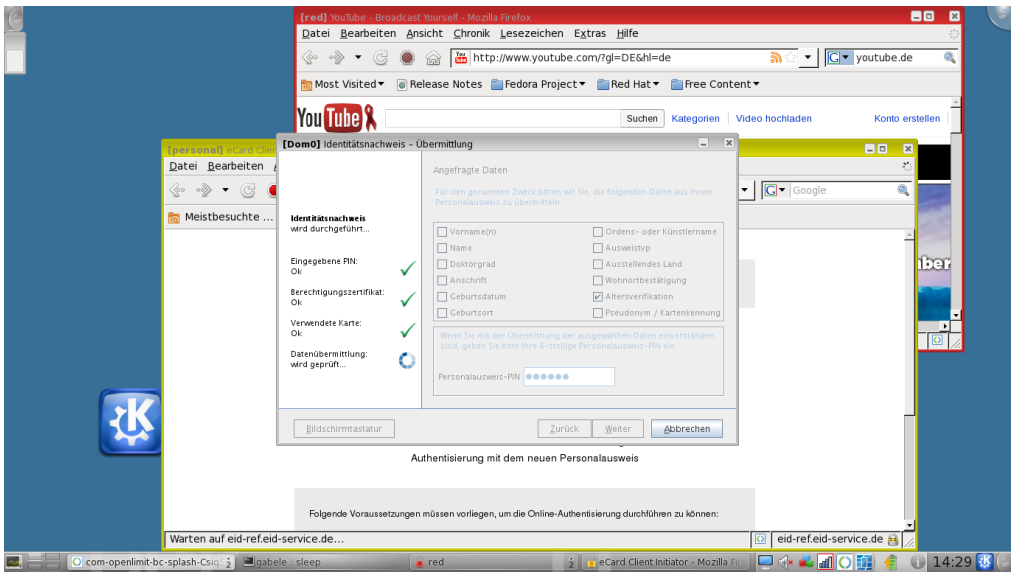

Abbildung 19: PIN-Eingabe und Datenübermittlung

Die Volljährigkeit wurde gegenüber der Online-Videothek mit der eID-Funktion nachgewiesen und der Anwender erhält Zugriff auf die Filme mit FSK 18.

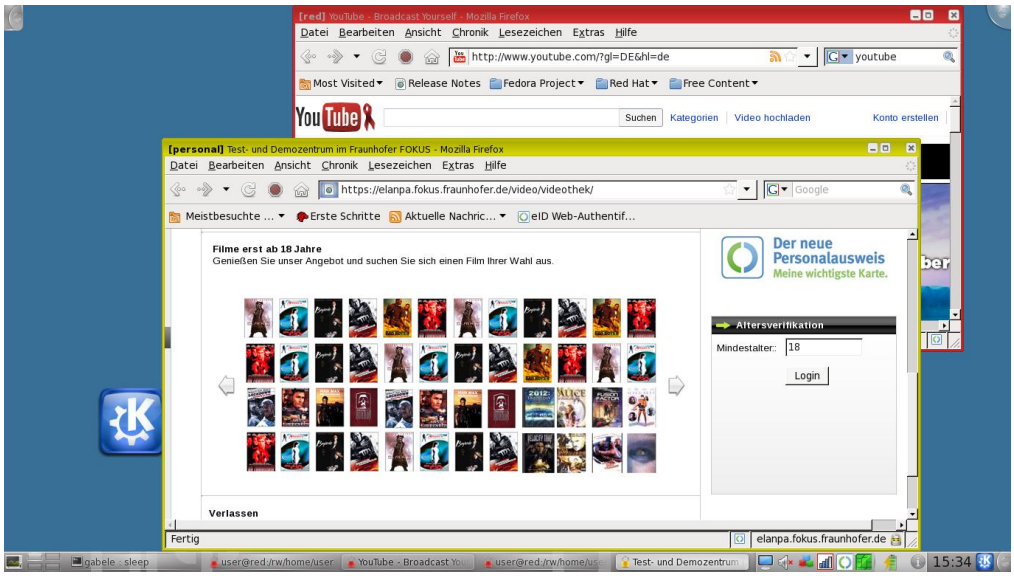

Abbildung 20: Zugriff auf FSK 18-Filme nach Altersverifikation

Evaluierungsergebnis: Die folgende Auswertung der Log-Dateien des Keyloggers in Tabelle 5 zeigen, dass alle Tastatureingaben in „trusted Browser“ mitgeschrieben wurden. Es wurde die Authentifizierung am Testsystem (vgl. Abbildung 17: Authentifizierung Testsystem) protokolliert. Die PIN-Eingabe zur eID-Funktion, die in Dom0 ausgeführt wurde, konnte der Keylogger - wie in der folgenden Tabelle dargestellt- nicht mitschreiben.

```
20111115|2043|Dialog|0x01200917|user|Authentifizierung
erforderlich|elanpa[KeyName:Tab]Cf42!Mo3
```

Tabelle 5: Log-Datei Prototyp (geschütztes System)

Die Clicks konnte der Keylogger nicht mit Sreenshots protokollieren. Ein Zugriff auf die Screenshots bzw. auf die Click-Screenshots ist aufgrund der in Abschnitt 6.1 dargestellten GUI-Separation in Qubes OS nicht möglich. Die Click- und Screenshot-Log-Dateien sind leer. Obwohl die Fenster auf einem Bildschirm dargestellt werden, ist ein Screenshot aus „trusted Browser“ nicht erzeugbar.

Ein weiterer Test, bei dem in „untrusted Browser“ der Keylogger installiert wird, zeigt, dass auch hier die Eingaben in einer anderen Sicherheitsdomäne, zum Beispiel in „trusted Browser“, nicht mitgeschrieben werden können. Aus diesen Tests lässt sich schließen, dass

in diesem Szenario ein Keylogger in einer Sicherheitsdomäne nicht in der Lage ist, die PIN-Eingabe in anderen Sicherheitsdomänen zu sniffen.

Evaluierung auf ungeschütztem Vergleichssystem: Der nachfolgende Log in Tabelle 6 wurde bei Ausführung der eID-Funktion auf einem ungeschützten Debian-System erzeugt.

Authentifizierung Testsystem

20111202|1809|Dialog|0x02000c77|gabele|Authentifizierung erforderlich|elanpa[KeyName:Tab]Cf42!Mo3[KeyName:Return]

Eingabe der 6-stelligen PIN der eID-Funktion

20111202|1810|Focus-Proxy-Window|0x04200117|gabele|FocusProxy|123987

Tabelle 6: Log-Datei ungeschütztes System

Neben dem Passwort Cf42!Mo3 für die Testplattform konnte auch die sechsstellige PIN 123987 zur eID-Funktion protokolliert werden.

8.1.2. Angriffsvektor: Inhaber zur Preisgabe der PIN verleiten

Das Angriffsszenario: Der Angreifer infiziert den PC mit einem Schadprogramm und erzeugt, z.B. mittels Java-Script, ein gefälschtes Eingabefenster und verleitet den Ausweisinhaber seine PIN in das gefälschte Eingabefenster einzugeben. Diese wird mitgeschrieben und z.B. per E-Mail mit weiteren Informationen an den Angreifer gesendet. Dieser Angriffsvektor könnte sich gegen Dom0 oder den Hypervisor richten. Entsprechend den in Abschnitt 6.1 beschriebenen Eigenschaften von Dom0 und Hypervisor wird ein erfolgreicher Angriff auf die Trusted Computing Base mit gering eingestuft. Im Rahmen der Evaluierung wird dieser Fall über ein rudimentäres Eingabefenster simuliert.

Durchführung Evaluierung: Bei einem Angriff über den Browser erscheint das Eingabefenster - wie aus Abbildung 21 ersichtlich - mit einem gelben Rahmen.

Evaluierungsergebnis: Die Farbe Gelb des Eingabefensters signalisiert dem Nutzer, dass er sich nicht in der Dom0 befindet und es sich dabei um ein gefälschtes Eingabefenster zur

AusweisApp handeln muss. Alle Fenster, die aus Dom0 erzeugt werden, haben, wie in Abbildung 22 dargestellt, die Rahmenfarbe Grau. Daraus lässt sich schließen, dass das

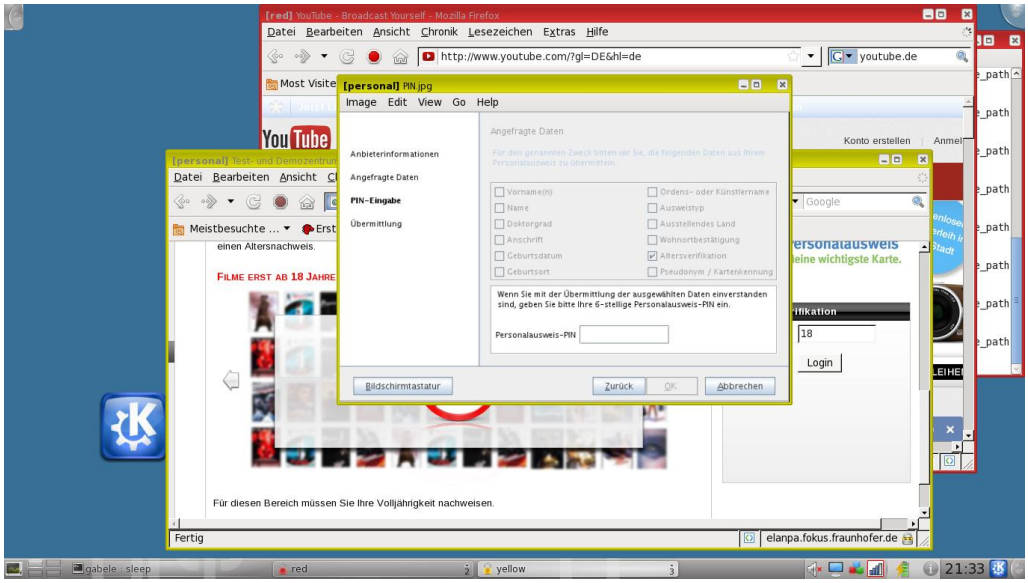

Abbildung 21: Gefälschtes Fenster für die PIN-Eingabe

Signalisieren, mittels der Fensterfarbe der virtuellen Maschine auf der gerade gearbeitet wird, den Anwender in die Lage versetzt diesen Angriffsvektor abzuwehren. Bei Anwendung des Lösungskonzeptes entsprechend Kapitel 6, könnte der Anwender zwischen den Farben blau, grün oder schwarz als Fensterfarbe für die eID-Funktion wählen.

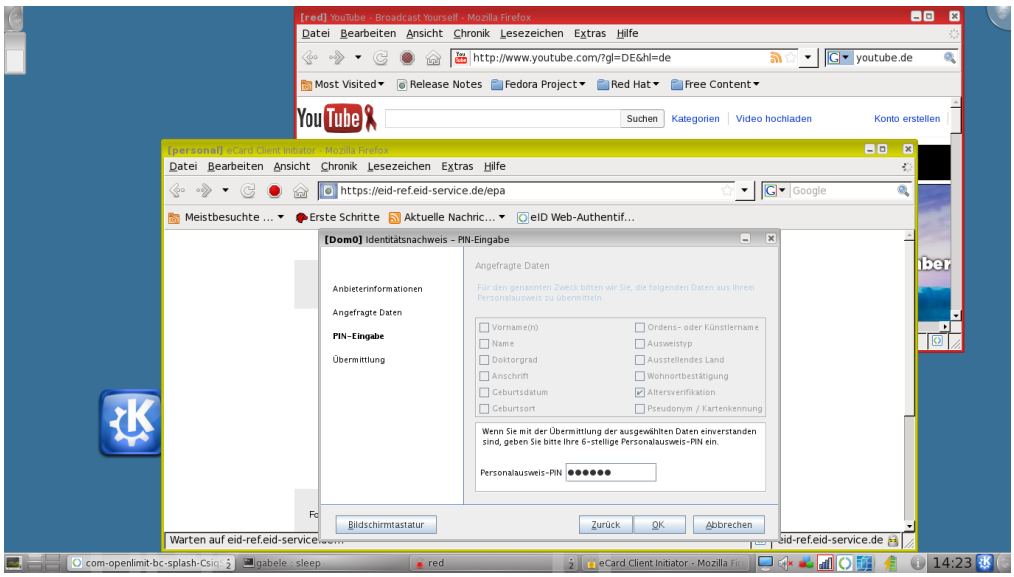

Abbildung 22: PIN-Eingabe

Evaluierung auf ungeschütztem Vergleichssystem: Auf den folgenden zwei Abbildungen ist die PIN-Eingabe mit einer nicht geschützten AusweisApp auf einem Debian-Betriebssystem dargestellt. Der Nutzer der eID-Funktion kann bei hochwertigem Nachbau des Eingabefensters kaum erkennen, welches gefälscht bzw. echt ist. Zuerst das echte Eingabefenster auf dem ungeschützten System:

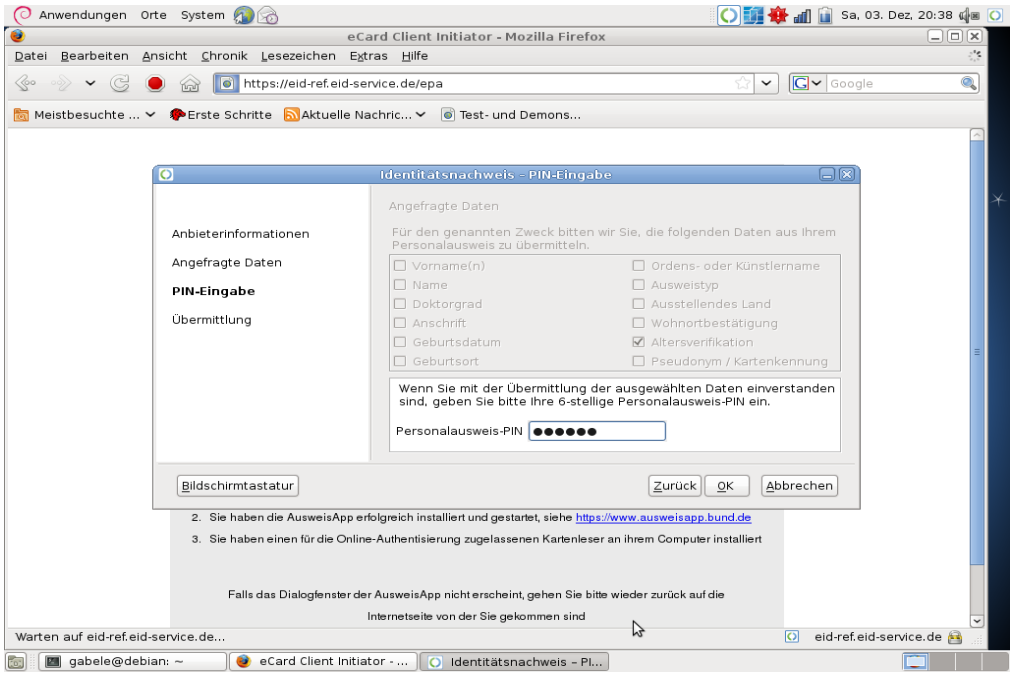

Abbildung 23: PIN-Eingabe zur eID-Funktion im ungeschützten System

Im Vergleich dazu das gefälschte Eingabefenster auf einem ungeschützten System:

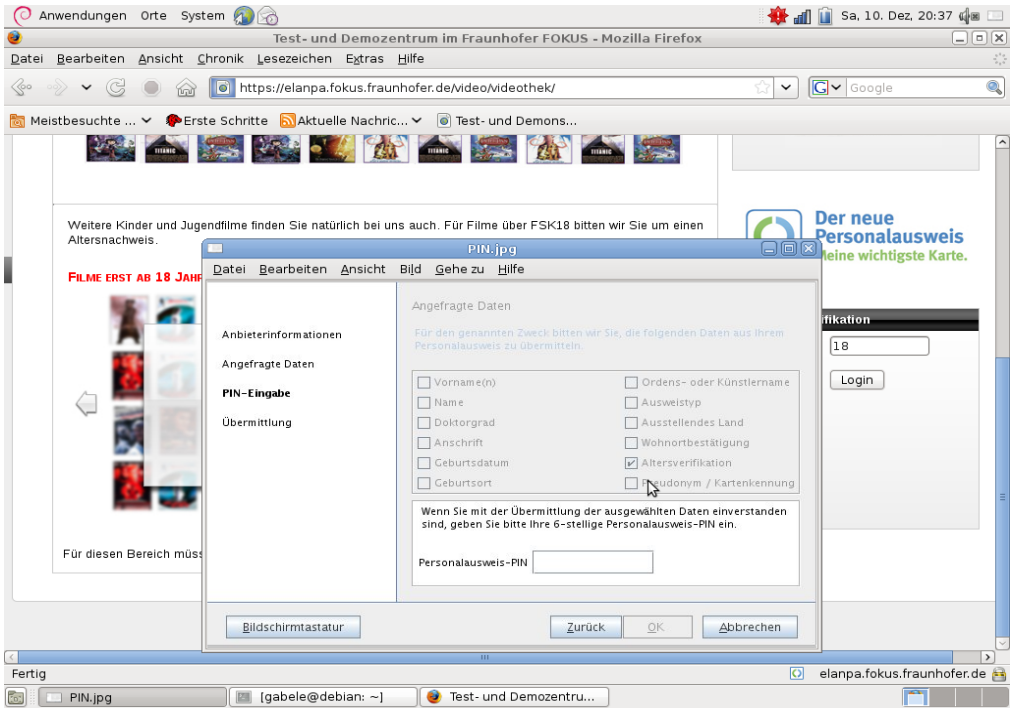

Abbildung 24: PIN-Eingabe im gefälschten Eingabefenster im ungeschützten System

8.1.3. Angriffsvektor: Angriff auf den Browser (inkl. Browser-Plugin)

In diesem Abschnitt wird folgender Angriffsvektor dargestellt: „Ein Angreifer kann durch einen Angriff auf den Browser (inklusive Browser-Plugin) Zugriff auf den PC erlangen und erreicht damit Zugriff auf den neuen Personalausweis“.

Das Angriffsszenario: Der Angreifer infiziert den PC über den Browser mit einem Schadprogramm. Er könnte über eine Schwachstelle z.B. im Browser-Plugin, per JavaScript einen Kanal zur Chipkarte öffnen. Darüber könnte er beliebige APDUs (Application Protocol Data Units) an die Karte schicken und die Antworten lesen. Im Rahmen der Evaluierung dieses Angriffsvektors wird dieser Fall durch Überprüfung des Zugriffes auf den Kartenleser evaluiert.

Durchführung Evaluierung: Mit dem Befehl lsusb wird versucht auf den Personalausweis zuzugreifen. Ein Zugriff auf den Basis-Kartenleser ist aus „trusted Browser“ nicht möglich.

Die Ausgabe des Befehls lsusb ist leer.

```
[user@personal]# lsusb
[user@personal]#
```

Tabelle 7: Ausgabe des Befehls lsusb in „trusted Browser“

Die Ausgabe des Befehls lsusb in Dom0 zeigt den Zugriff auf den Kartenleser „SCM Microsystems, Inc. am Bus 005 mit der Geräte ID 04e6:5292.

```
[gabele@dom0 ~]# lsusb
…
Bus 005 Device 002: ID 04e6:5292 SCM Microsystems, Inc.
…
[gabele@dom0 ~]#
```

Tabelle 8: Ausgabe des Befehls lsusb in "eID"

Wenn kein Zugriff auf den Kartenleser besteht, ist ein Angriff auf die Karte erschwert. Die Eintrittswahrscheinlichkeit eines erfolgreichen Angriffes über „trusted Browser“ mit dem Ziel Zugriff auf den nPA wird dadurch verringert, dass der Anwender sich in „trusted Browser“ nur auf vertrauenswürdigen Webseiten bewegt.

Denkbar wäre die Ausnutzung des Kommunikationskanals zwischen den beiden virtuellen Maschinen. Es kann jedoch nur aus Dom0 diese Verbindung aufgebaut werden. Die Dauer des geöffneten Kommunikationskanals ist begrenzt, auf die tatsächlich benötigte Zeitdauer zur Ausführung der eID-Funktion. Der Anwender wird über den Aufbau der Verbindung und die erfolgreiche Beendigung der Verbindung informiert. Ein erfolgreicher Angriff mit einem spezialisierten Schadprogramm, über den Browser, mit dem Ziel die Identität des Ausweis-Inhabers zu rauben wird deutlich erschwert. In Dom0 ist kein Browser installiert und damit ist ein Angriff über diesen Weg in Dom0 ausgeschlossen.

Evaluierungsergebnis: Daraus lässt sich schließen, dass ohne Zugriff auf den Basis-Kartenleser ein erfolgreicher Angriff über den Browser inklusive dem Browser-Plugin mit dem Ziel des Zugriffs auf den nPA, durch die Anwendung des Konzeptes, erschwert ist.

8.2. Anforderung 2: Anwenderfreundlichkeit

Die Bedienung der eID-Funktion hat sich durch die Anwendung des Lösungskonzeptes - wie der Ablauf in Abschnitt 8.1.1 zeigt - geändert. Bevor die eID-Funktion ausgeführt werden kann, müssen zwei virtuelle Maschinen gestartet werden. Der Anwender muss nun zusätzlich ein Script ausführen, bevor er die eID-Funktion nutzen kann. Der Nutzer der eID-Funktion sollte auf die Farbe des Eingabefensters achten, in das er seine PIN eingibt. Nach Einschätzung des Autors sind diese Änderungen in der Anwendung der eID-Funktion akzeptabel.

8.3. Anforderung 3: Performance

Änderungen in der Performance der eID-Funktion selbst, waren nach Anwendung des Lösungskonzeptes mit Qubes OS subjektiv nicht feststellbar. Basis für den Vergleich war die Nutzung der eID-Funktion auf dem Debian-Betriebssystem. Bevor die eID-Funktion ausgeführt werden kann, müssen zwei virtuelle Maschinen und das Script zur Kommunikation gestartet werden, was zusätzlich Zeit benötigt. Der Start einer virtuellen Maschine dauert ca. 20 Sekunden. Der Start des Scripts zur Kommunikation benötigt unter 10 Sekunden. Nach Einschätzung des Autors ist die Performance gut.

8.4. Anforderungen, die erfüllt werden müssen

8.4.1. Hinreichendes Sicherheitsniveau

Entsprechend den Ergebnissen aus Abschnitt 8.1 hat die prototypische Implementierung gezeigt, dass ein ausreichender Mindestschutz gewährleistet und das Sicherheitsniveau verbessert werden konnte.

8.4.2. Beibehaltung Sicherheitskonzept

Mit der Anwendung des Lösungskonzeptes wurde die Kommunikation zwischen den beiden Sicherheitsdomänen über ein netzwerkbasiertes SSH-Tunneling durchgeführt. Die Evaluierung der Beibehaltung des Sicherheitskonzeptes der eID-Funktion, erfolgt durch den Vergleich der Datenströme an Localhost in Dom0 bei Anwendung der eID-Funktion mit und ohne Trennung der AusweisApp. Es wird überprüft, ob sich die Datenströme an Localhost in Dom0 mit und ohne Isolierung eID und „trusted Browser“ verändern. Zum

Zweiten werden die Datenströme, von Browser-Plugin mit der eID-Funktion, auf Localhost in einer Maschine zu den Datenströmen im Kommunikationskanal gemäß Lösungskonzept zwischen „trusted Browser“ und eID verglichen. Das Capture mittels Wireshark zeigt in Tabelle 9 die Datenströme an Localhost in Dom0, während der Nutzung der eID-Funktion mit dem SSH-Tunnel.

No.	*Time*	*Source*	*Destination*	*Protocol*	*Info*
1	*0.000000*	*127.0.0.1*	*127.0.0.1*	*TCP*	*58374 > 18080 [SYN]Len=0*
2	*0.000031*	*127.0.0.1*	*127.0.0.1*	*TCP*	*18080 > 58374 [SYN,ACK] Len=0*
3	*0.000056*	*127.0.0.1*	*127.0.0.1*	*TCP*	*58374 > 18080 [ACK]Len=0*
4	*0.000730*	*127.0.0.1*	*127.0.0.1*	*TCP*	*58374 > 18080 [PSH,ACK] Len=7302*
5	*0.000747*	*127.0.0.1*	*127.0.0.1*	*TCP*	*18080 > 58374 [ACK] Len=0*
6	*4.183.739*	*127.0.0.1*	*127.0.0.1*	*TCP*	*18080 > 58374 [PSH, ACK] Len=1624*
7	*4.183.785*	*127.0.0.1*	*127.0.0.1*	*TCP*	*18080 > 58374 [FIN, ACK] Len=0*
8	*4.183.827*	*127.0.0.1*	*127.0.0.1*	*TCP*	*58374 > 18080 [ACK] Len=0*
9	*4.184.462*	*127.0.0.1*	*127.0.0.1*	*TCP*	*58374 > 18080 [FIN, ACK] Len=0*
10	*4.184.501*	*127.0.0.1*	*127.0.0.1*	*TCP*	*18080 > 58374 [ACK] Len=0*

Tabelle 9: Capture der Kommunikation an Localhost in Dom0 mit SSH-Tunnel

Tabelle 10 zeigt zum Vergleich die Datenströme an Localhost in Dom0, wenn die AusweisApp nicht getrennt ist.

No.	Time	Source	Destination	Protocol	Info
1	0.000000	127.0.0.1	127.0.0.1	TCP	39794 > 18080 [SYN] Len=0
2	0.000016	127.0.0.1	127.0.0.1	TCP	18080 > 39794 [SYN, ACK] Len=0
3	0.000028	127.0.0.1	127.0.0.1	TCP	39794 > 18080 [ACK] Len=0
4	0.000235	127.0.0.1	127.0.0.1	TCP	39794 > 18080 [PSH, ACK] Len=7302
5	0.000245	127.0.0.1	127.0.0.1	TCP	18080 > 39794 [ACK] Len=0
6	3.946.089	127.0.0.1	127.0.0.1	TCP	18080 > 39794 [PSH, ACK] Len=1624
7	3.946.127	127.0.0.1	127.0.0.1	TCP	18080 > 39794 [FIN, ACK] Len=0
8	3.946.498	127.0.0.1	127.0.0.1	TCP	39794 > 18080 [ACK] Len=0
9	3.947.122	127.0.0.1	127.0.0.1	TCP	39794 > 18080 [FIN, ACK] Len=0
10	3.947.159	127.0.0.1	127.0.0.1	TCP	18080 > 39794 [ACK] Len=0

Tabelle 10: Capture der Kommunikation an Localhost in Dom0 ohne SSH-Tunnel

Ergebnis der Evaluierung ist, dass die Anzahl und die Länge der Datenströme vor und nach Anwendung des Lösungskonzeptes identisch sind.

Tabelle 11 zeigt einen Ausschnitt aus dem Austausch der verschlüsselten Datenpakete im Kommunikationskanal zwischen Dom0 und „trusted Browser“ in der virtuellen Netzwerkschnittstelle, während Nutzung der eID-Funktion:

No.	Time	Source	Destination	Protocol	Info
1	0.000000	10.99.0.2	10.99.0.1	SSH	Encrypted response packet len=96
2	0.000026	10.99.0.1	10.99.0.2	TCP	51424 > ssh [ACK]Len=0
3	0.000396	10.99.0.1	10.99.0.2	SSH	Encrypted request packet len=48
4	0.000863	10.99.0.2	10.99.0.1	SSH	Encrypted response packet len=5792
5	0.000874	10.99.0.1	10.99.0.2	TCP	51424 > ssh [ACK] Len=0
6	0.000955	10.99.0.2	10.99.0.1	SSH	Encrypted response packet len=1448
7	0.000962	10.99.0.2	10.99.0.1	SSH	Encrypted response packet len=104
8	0.000966	10.99.0.1	10.99.0.2	TCP	51424 > ssh [ACK] Len=0
9	4.752112	10.99.0.1	10.99.0.2	SSH	Encrypted request packet len=1448
10	4.752142	10.99.0.1	10.99.0.2	SSH	Encrypted request packet len=216
11	4.752193	10.99.0.1	10.99.0.2	SSH	Encrypted request packet len=32
12	4.752491	10.99.0.2	10.99.0.1	TCP	ssh > 51424 [ACK] Len=0
13	4.753841	10.99.0.2	10.99.0.1	SSH	Encrypted response packet len=64
14	4.755266	10.99.0.1	10.99.0.2	SSH	Encrypted request packet len=32
15	4.793546	10.99.0.2	10.99.0.1	TCP	ssh > 51424 [ACK] Len=0

Tabelle 11: Capture der Kommunikation im SSH-Tunnel

Der Kommunikationskanal im Lösungskonzept basiert auf dem SSH-Tunneling. Daher sind die 10 TCP-Pakete aus dem Capture in Tabelle 9 bzw. 10 in Tabelle 11 in SSHv2-Pakete umgewandelt. Der Datenstrom Nummer 4 mit der Funktion „Aufruf eID-Client“ (Abbildung 11 in Abschnitt 5.10.2, Funktion 10) aus dem Capture in Tabelle 9 bzw. 10 entsprechen den Paketen 4,6,7 aus dem Capture in Tabelle 11. Der Datenstrom Nummer 6 mit der Funktion „Nachricht eID-Funktion durchgeführt“ (Abbildung 11 in Abschnitt 5.10.2, Funktion 35) aus dem Capture in Tabelle 9 bzw. 10 entsprechen den Paketen 9-11 aus dem Capture in Tabelle 11. Eine exakte Zuordnung durch Abgleich der Länge der

Datenpakete ist aufgrund der zusätzlichen Verschlüsselung nicht möglich.
Zusammenfassend lässt sich feststellen, dass mit Anwendung des Lösungskonzeptes das Sicherheitskonzept der eID-Funktion beibehalten wurde.

8.4.3. Beibehaltung Funktionsumfang

Mit den durchgeführten Tests konnte gezeigt werden, dass die in Abschnitt 4.1.1 dargestellten Funktionen der AusweisApp weiterhin betrieben werden können. Die Software-Aktualisierungsfunktion zur eID-Funktion im Prototyp steht nicht zur Verfügung, da die AusweisApp noch nicht für die Linux-Distribution Fedora vorliegt. Wenn die AusweisApp für Fedora zur Verfügung stehen würde, hätte der Anwender die Möglichkeit jedes Mal nach Neustart die AusweisApp zu aktualisieren. Alternativ könnte der Anwender die AusweisApp auf der TemplateVM aktualisieren

8.5. Bewertung des Sicherheitsgewinns

Die Evaluierung der Anforderungen - mit der Demonstrationsimplementierung - kann wie folgt zusammengefasst werden: Durch Anwendung des Lösungskonzeptes können die Angriffsvektoren

- Inhaber zur Preisgabe der PIN verleiten
- Ausspähen der PIN
- Über den Browser des Benutzer-PCs Zugriff auf den Personalausweis erlangen

mit folgenden Eigenschaften der Implementierung abgewehrt werden:

- Der Anwender kann immer erkennen, welche Fenster aktiv sind, in welchem er sich gerade befindet und welcher Sicherheitsdomäne die aktive Anwendung zugeordnet ist
- Es kann kein Fenster aus einer nicht privilegierten virtuellen Maschine in einer privilegierten Maschine initiiert werden
- Anwendungen aus anderen virtuellen Maschinen können nicht in privilegierten Maschinen Snapshots erzeugen, sniffen und sie können keine Events in Anwendungen der privilegierten Maschine einfügen

Restrisiken bestehen bezüglich eines erfolgreichen Angriffes, durch die Ausnutzung des zeitlich begrenzt geöffneten Kommunikationskanals zwischen der „eID“ und „trusted Browser“. Ein spezialisiertes Schadprogramm könnte über eine vertrauenswürdige Website eingeschleust werden und mit Öffnen der Verbindung die virtuelle Maschine mit der eID-Funktion kompromittieren. Des Weiteren bestehen Restrisiken, bezüglich des Fehlverhaltens des Nutzers. Er könnte zum Beispiel die eID-Fensterfarbe (in unserem Szenario grau) nicht beachten und versehentlich in ein gefälschtes eID-Fenster die PIN-eingeben. Es bestehen weitere Restrisiken bzgl. eines erfolgreichen Angriffs auf Dom0 bzw. den Hypervisor.

Das Umsetzungskonzept des Prototyps weicht in den folgenden Punkten vom Lösungskonzept aus Kapitel 6 ab: Das Lösungskonzept sieht die Installation der eID-Funktion in eine ApplicationVM vor. Dom0 benötigt keinen Netzwerkzugang. Daraus ergibt sich ein weiterer Sicherheitsgewinn des Lösungskonzeptes gegenüber dem Umsetzungskonzept. Daraus kann geschlossen werden, dass die Ergebnisse der Evaluierung anhand des Prototyps auf das Lösungskonzept übertragbar sind.

Der Ansatz Sicherheit durch Isolation hat das Risiko eines erfolgreichen Angriffs mit dem Ziel „Identität rauben“ unter Nutzung des Basis-Kartenlesers von „nicht gering“ auf „gering“ vermindert. Sie zeigt, dass Restrisiken bestehen und der Grad des Sicherheitszuwachses unmittelbar von dem Verhalten des privaten Nutzers abhängt. Das zusammengefasste Ergebnis, der erneuten Risikoanalyse auf Basis des Prototyps, mit dem Angriffsziel „Identität rauben“ ist der Abbildung 25 zu entnehmen

8.6. Im Rahmen der Arbeit wurde nicht betrachtet

Interne Angreifer, wie Familienmitglieder oder Angestellte/Mitarbeiter wurden in dem gewählten Szenario nicht betrachtet. Zur Abschätzung des Risikos konnten im Rahmen der Diplomarbeit keine Experten zur Abschätzung des Risikos hinzugezogen werden. Penetrationstests konnten ebenso aufgrund der Rahmenbedingungen einer Diplomarbeit nicht durchgeführt werden. Des Weiteren wurde die Marktreife der im Lösungskonzept verwendeten Software nicht betrachtet.

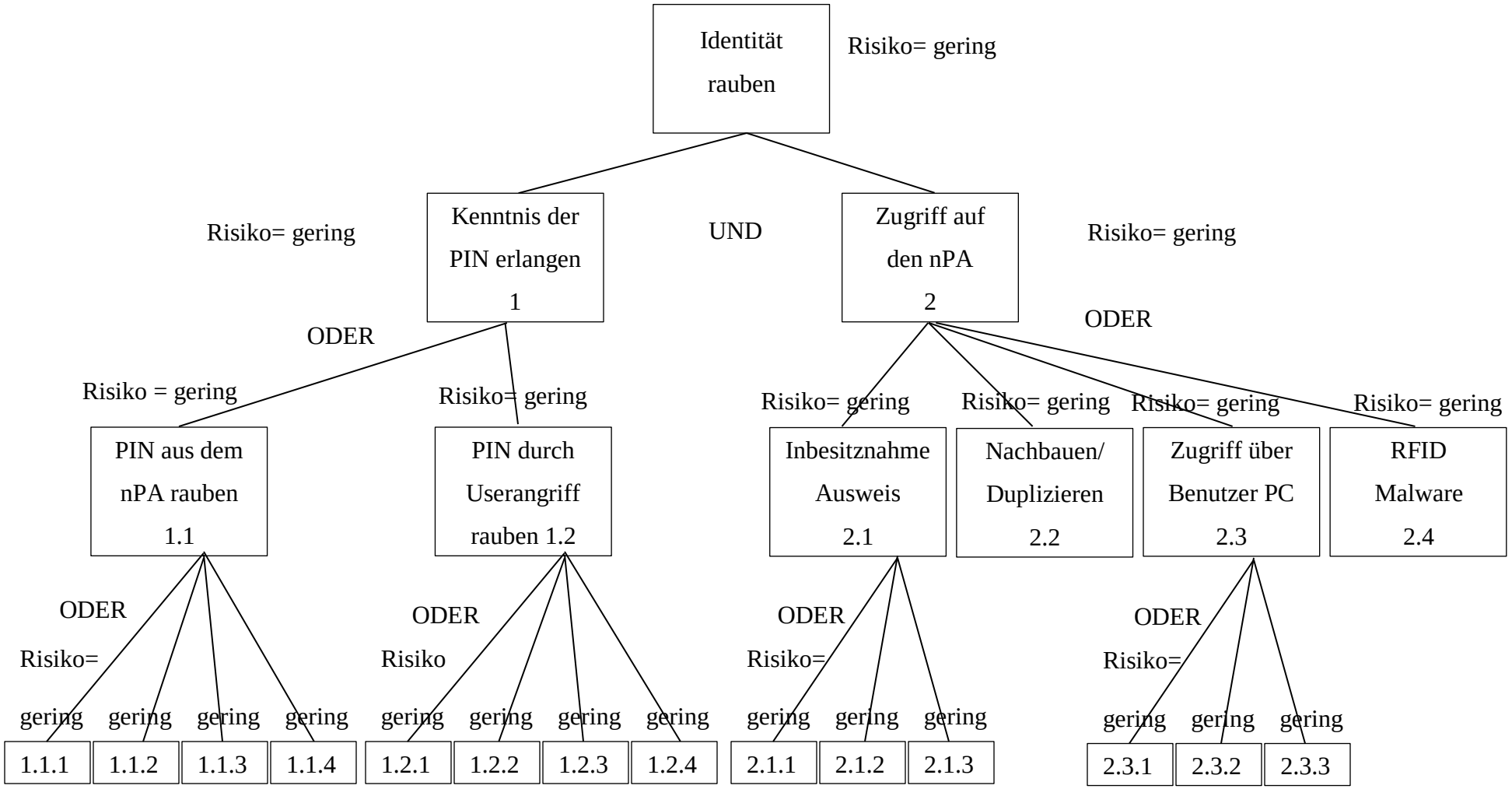

Abbildung 25: Risikoanalyse Prototyp: Angriffsziel "Identität rauben"

9. Fazit

Die sichere Nutzung der eID-Funktion mit dem neuen Personalausweis, unter Einsatz des weit verbreiteten Basis-Kartenlesers, hängt unmittelbar vom Sicherheitsniveau des PCs ab. Die durchgeführte Sicherheitsanalyse zeigt, dass mit den neuen Nutzungsmöglichkeiten der eID-Funktion ein hohes Schutzniveau notwendig ist. Es besteht ein nicht geringes Risiko, dass über einen Angriff die PIN des Personalausweises ausgespäht wird oder der Ausweisinhaber zur Preisgabe der PIN verleitet wird und der Angreifer Zugriff über den Browser auf den Personalausweis erlangt. Somit kann der Angreifer die fremde Identität ausnutzen. Zur Verbesserung des Sicherheitsniveaus der AusweisApp wird der Ansatz der Separierung angewendet. In den erarbeiteten Handlungsalternativen wird die AusweisApp in einen eigenen Benutzer, bzw. in einen eigenen physischen PC, bzw. in eigene virtuelle Maschinen separiert. Aus den verschiedenen Lösungsalternativen wird das Konzept der Isolierung der Anwendungen in virtuelle Maschinen gewählt. Alle Anwendungen, außer der AusweisApp, werden in einer virtuellen Maschine isoliert. Zusätzlich werden die Funktionen der AusweisApp entsprechend den Schutzanforderungen in zwei virtuelle Maschinen geteilt und mit einem Kommunikationskanal zur Aufrechterhaltung der eID-Funktion verbunden. Dieses Konzept wird auf das open source Betriebssystem Qubes OS abgebildet. Qubes OS bietet die Möglichkeit, durch Virtualisierung Sicherheitsdomänen zu schaffen. Daneben kann das Betriebssystem durch die GUI-Separation, unter Beibehaltung der Isolationseigenschaften, die Fenster der Sicherheitsdomänen auf einem Bildschirm anzeigen. Qubes OS verfügt über die Eigenschaft, dass bei einem erfolgreichen Angriff auf eine Sicherheitsdomäne, die anderen Sicherheitsdomänen noch sicher sind. Die Evaluierung anhand eines Prototyps zeigt, dass sich durch Anwendung des Lösungskonzeptes, das Risiko der Angriffsvektoren deutlich vermindert. Es lässt sich zusammenfassend feststellen, dass der gewählte Ansatz das Sicherheitsniveau des PCs bezüglich der eID-Funktion verbessert und dadurch der hohe Schutzbedarf dieser Funktion erfüllt werden kann. Restrisiken, aufgrund von möglichem Fehlverhalten des Nutzers, bleiben bestehen.

Die Ergebnisse dieser Arbeit lassen sich auf weitere sicherheitskritische Anwendungen übertragen, die eine Smartcard mit Basis-Kartenleser und PIN-Eingabe nutzen.

Beispielsweise können Weiterentwicklungen von Anwendungen unter Nutzung der elektronischen Gesundheitskarte oder von Signaturkarten Ansätze bieten, um das Sicherheitsniveau dieser Anwendungen weiter zu erhöhen.

Das im Rahmen dieser Arbeit vorgeschlagene Konzept beinhaltet das Signalisieren des Sicherheitsniveaus einer Domäne durch die Farbe des Fensterrahmens. Daraus ergibt sich als Anknüpfungspunkt für diese Arbeit, die Integration des beschriebenen Signalisierens zur Darstellung des Sicherheitsniveaus von Anwendungen. Dabei kann unter anderem betrachtet werden, welche Sicherheitsniveaus, welche Farben repräsentieren sollen oder, wie viele Farben maximal unterschiedliche Sicherheitsniveaus differenzieren können. In diesem Zusammenhang ist auch die Betrachtung interessant, wie der Anwender motiviert werden kann, diese Fensterfarben zu beachten.

Ein weiterer Ansatz für eine Weiterentwicklung ergibt sich aus der Kombination der zwei Entwicklungen Qubes OS und AusweisApp. Nutzer der eID-Funktion mit Basis-Kartenleser würden davon profitieren, wenn eine paketierte AusweisApp für Qubes OS bereitgestellt werden würde.

10. Literaturverzeichnis

[1]. **Bundesministerium des Innern.** *Der neue Personalausweis.* Berlin : Fachverlag Jüngling-gbb, 2010. S. 4-6.

[2]. **Bundesamt für Sicherheit in der Informationstechnik.** Technische Richtlinie TR-03127. *Architektur elektronischer Personalausweis und elektronischer Aufenthaltstitel.* [Online] 27. Mai 2011. [Zitat vom: 30. Oktober 2011.] https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03127/BSI-TR-03127_pdf.pdf?__blob=publicationFile.

[3]. **FOKUS.** Frauenhofer-Institut für Offene Kommunikation. *Onlineanwendungen.* [Online] 26. Oktober 2011. [Zitat vom: 17. Dezember 2011.] http://www.ccepa.de/onlineanwendungen.

[4]. **Bundesamt für Sicherheit in der Informationstechnik.** Technische Richtlinie TR-03119. *Anforderung an Chipkartenleser mit nPA Unterstützung.* [Online] Mai 2011. [Zitat vom: 30. Oktober 2011.] https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03119/BSI-TR-03119_V1_pdf.pdf?__blob=publicationFile.

[5]. **Fachhochschule Gelsenkirchen, Institut für Internet-Sicherheit.** Restrisiken. *Studie - Restrisiken beim Einsatz der AusweisApp.* [Online] Oktober 2010. [Zitat vom: 08. Dezember 2011.] http://www.bmi.bund.de/SharedDocs/Downloads/DE/Themen/Sicherheit/PaesseAusweise/restrisiken.pdf?__blob=publicationFile.

[6]. **Computer Bild Digital GmbH.** *Kartenlesegerät in COMPUTER BILD.* [Online] 02. Dezember 2010. [Zitat vom: 16. Dezember 2011.] http://www.computerbild.de/artikel/cb-Special-Der-neue-Personalausweis-Kartenleser-im-Heft-5806626.html.

[7]. **Westdeutscher Rundfunk Köln.** Sendungsbeiträge. *Personalausweis.* [Online] 13. September 2010. [Zitat vom: 27. Januar 2011.] http://www.wdr.de/tv/markt/sendungsbeitraege/2010/0913/01_personalausweis_pr.jsp.

[8]. **Heise Security.** Security Meldung. *Phishing-Demo-zum-ePerso-Update.* [Online] 17. Januar 2011. [Zitat vom: 24. Januar 2011.] http://www.heise.de/security/meldung/Phishing-Demo-zum-ePerso-Update-1170481.html.

[9]. **Heise Security.** Security Meldung. *Weitere Sicherheitslücke beim elektronischen Personalausweis.* [Online] 08. August 2011. [Zitat vom: 12. August 2011.] http://www.heise.de/security/meldung/Weitere-Sicherheitsluecke-beim-elektronischen-Personalausweis-1319432.html.

[10]. **Eckert, Claudia.** *IT-Sicherheit.* München : Oldenbourg Wissenschaftsverlag GmbH, 2009. S. 167.

[11]. **Bundesamt für Sicherheit in der Informationstechnik.** *Schichtenmodell und Modellierung.* [Online] 2008. [Zitat vom: 27. Mai 2011.] https://www.bsi.bund.de/ContentBSI/grundschutz/kataloge/allgemein/modellierung/02001.html.

[12]. **Eckert, Claudia.** *IT-Sicherheit.* München : Oldenbourg Wissenschaftsverlag GmbH, 2009. S. 4f.

[13]. **Bundesamt für Sicherheit in der Informationstechnik.** *Webkurs IT-Grundschutz.* [Online] 2008. [Zitat vom: 26. Januar 2011.] https://www.bsi.bund.de/DE/Themen/weitereThemen/WebkursITGrundschutz/Schutzbedarfsfeststellung/Schutzbedarfskategorien/Schutzziele/schutzziele_node.html.

[14]. **Eckert, Claudia.** *IT-Sicherheit.* München : Oldenbourg Wissenschaftsverlag GmbH, 2009. S. 17f.

[15]. **Fraunhofer-Institut für Sichere Informationstechnologie, MEZ, TZI.** *RFID-Studie 2007.* [Online] April 2007. [Zitat vom: 23. Juli 2011.] http://sit.sit.fraunhofer.de/studies/de/studie-rfid-2007-de.pdf.

[16]. **Bundesamt für Sicherheit in der Informationstechnik.** *Lagebericht 2011.* [Online] 2011. [Zitat vom: 22. Juli 2011.] https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/Lageberichte/Lagebericht2011_nbf.pdf?__blob=publicationFile.

[17]. **Eckert, Claudia.** *IT-Sicherheit.* München : Oldenbourg Wissenschaftsverlag GmbH, 2009. S. 14f.

[18]. **Bundesamt für Sicherheit in der Informationstechnik.** *IT-Grundschutzkataloge.* [Online] 2008. [Zitat vom: 26. Januar 2011.] https://www.bsi.bund.de/cln_183/DE/Themen/weitereThemen/ITGrundschutzKataloge/itgrundschutzkataloge_node.html.

[19]. **Bundesministerium der Justiz.** Personalausweisgesetz. *Gesetz über Personalausweise und den elektronischen Identitätsnachweis.* [Online] 18. Juni 2009. [Zitat vom: 30. Oktober 2011.] http://www.gesetze-im-internet.de/pauswg/BJNR134610009.html#BJNR134610009BJNG000100000.

[20]. **Bundesdruckerei GmbH.** *pocketguide eID-Service 2011.* [Online] August 2011. [Zitat vom: 29. Oktober 2011.] http://www.bundesdruckerei.de/de/service/service_downloads/untern_eID-service_de.pdf.

[21]. **Bundesamt für Sicherheit in der Informationstechnik.** Technische Richtlinie TR-03128. *EAC-PKI´n für den elektronischen Personalausweis.* [Online] 08. Oktober 2010. [Zitat vom: 29. Oktober 2011.] https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03128/BSI_TR-03128.pdf?__blob=publicationFile.

[22]. **Bundesamt für Sicherheit in der Informationstechnik.** Technical Guideline TR-03110. *Advanced Security Mechanisms for Machine Reaable Travel Documents.* [Online] 14. Oktober 2010. [Zitat vom: 04. November 2011.] https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/TechGuidelines/TR03110/TR-03110_v205_pdf.pdf;jsessionid=1E5F72F4250A1050B06ED1034F88B3B6.2_cid239?__blob=publicationFile.

[23]. **Bundesamt für Sicherheit in der Informationstechnik.** Technische Richtlinie TR-03130. *eID-Server.* [Online] 2010. [Zitat vom: 21. Oktober 2011.] https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03130/TR-03130_TR-eID-Server_V1_4_pdf.pdf;jsessionid=AF15C54996B4ADE7A957C6E223D0758C.2_cid231?__blob=publicationFile.

[24]. **ISO .** ISO-14443-1 Identification cards - Contactless integrated circuit cards - Proximity cards. *Part 1: Physical characteristics.* s.l. : ISO/IEC, 2008.

[25]. **Bundesamt für Sicherheit in der Informationstechnik.** Risiken und Chancen des Einsatzes von RFID-Systemen. [Online] 2005. [Zitat vom: 30. Oktober 2011.] https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/ElekAusweise/RFID/RIKCHA_barrierefrei_pdf.pdf?__blob=publicationFile.

[26]. **Bundesamt für Sicherheit in der Informationstechnik.** Technische Richtlinie TR-03111. *Elliptische-Kurven-Kryptographie (ECC).* [Online] 2009. [Zitat vom: 04. November 2011.] https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03111/BSI-TR-03111_pdf.pdf?__blob=publicationFile.

[27]. **Bundesamt für Sicherheit in der Informationstechnik.** *Der neue Personalausweis.* [Online] 30. Oktober 2011. [Zitat vom: 30. Oktober 2011.] https://www.bsi.bund.de/DE/Themen/ElektronischeAusweise/Personalausweis/AusweisApp/AusweisApp_node.html.

[28]. **Bundesamt für Sicherheit in der Informationstechnik.** Technical Guideline TR-03112-7. *eCard-API-Framework - Protocols.* [Online] 23. Mai 2011. [Zitat vom: 10. November 2011.] https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03112/API/api1_teil7_pdf.pdf;jsessionid=F69BEF8F1C1CD177D291720A13EC1900.2_cid251?__blob=publicationFile.

[29]. **Bundesamt für Sicherheit in der Informationstechnik.** Technical Guideline TR-03112-1. *eCard-API-Framework.* [Online] 23. Mai 2011. [Zitat vom: 17. November 2011.] https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03112/API/api1_teil1_pdf.pdf;jsessionid=F6E028D1CEE83A967AD2166D3E109DD9.2_cid248?__blob=publicationFile.

[30]. **ISO .** ISO 24727-4 Identification cards - Integrated circuit card programming interfaces. *Part 4: Application programming interfaces (API) administration.* s.l. : ISO/IEC, 2008.

[31]. **ISO .** ISO 7816-4 Identification cards - Integrated circuit cards. *Part 4: Organization, security and commands for interchange.* s.l. : ISO/IEC, 2005.

[32]. **Internet Engineering Task Force (IETF).** RFC 6101. *The Secure Socket Layer (SSL) Protocol Version 3.0.* [Online] Auguust 2011. [Zitat vom: 30. Oktober 2011.] http://tools.ietf.org/html/6101.

[33]. **W3C World Wide Web Consortium.** *SOAP Specifications.* [Online] 27. 04 2007. [Zitat vom: 12. Juni 2011.] http://www.w3.org/TR/soap/.

[34]. **OASIS Security Services TC.** Security Assertion Markup Language. *SAML V2.0 Technical Overview.* [Online] 03 2008. [Zitat vom: 05. Novmeber 2011.] http://www.oasis-open.org/committees/download.php/27819/sstc-saml-tech-overview-2.0-cd-02.pdf.

[35]. **Internet Engineering Task Force (IETF).** RFC 4279. *Pre-shared Ciphersuites for Transport Layer Security (TLS).* [Online] Dezember 2005. [Zitat vom: 30. Oktober 2011.] http://tools.ietf.org/html/rfc4279.

[36]. **Liberty Alliance Project.** liberty-paos-v2.0. *Reverse HTTP Binding for SOAP Specification.* [Online] 2006. [Zitat vom: 30. Oktober 2011.] http://projectliberty.org/liberty/content/download/909/6303/file/liberty-paos-v2.0.pdf.

[37]. **Bundesamt für Sicherheit in der Informationstechnik.** Technical Guideline TR-03129. *PKI´s for Machine Readable Travel Documents.* [Online] 09. November 2009. [Zitat vom: 05. November 2011.] https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/TechGuidelines/TG03129/BSI_TG_03129.pdf;jsessionid=8556860FAE2FB9D102332A49E597293F.2_cid156?__blob=publicationFile.

[38]. **Helfrich, Marcus Prof. Dr. und Geis, Ivo Dr.** *Datenschutzrecht.* München : Deutscher Taschenbuch Verlag, 2011. 3423057726.

[39]. **Eckert, Claudia.** *IT-Sicherheit.* München : Oldenbourg Wissenschaftsverlag GmbH, 2009. S. 6.

[40]. **Bundesamt für Sicherheit in der Informationstechnik.** *IT-Grundschutz-Vorgehensweise BSI-Standard 100-2.* [Online] 2008. [Zitat vom: 13. August 2011.] https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/ITGrundschutzstandards/standard_1002.pdf?__blob=publicationFile.

[41]. **Bundesamt für Sicherheit in der Informationstechnik.** Grundschutzkatalog G 2. *Organisatorische Mängel.* [Online] 2009. [Zitat vom: 14. November 2011.] https://www.bsi.bund.de/DE/Themen/weitereThemen/ITGrundschutzKataloge/Inhalt/

GefaehrdungskataloGe/G2OrganisatorischeMaengel/g2organisatorischemaengel_node.html.

[42]. **Bundesamt für Sicherheit in der Informationstechnik.** Grundschutzkatalog G 5. *Vorsätzliche Handlungen.* [Online] [Zitat vom: 14. November 2011.] https://www.bsi.bund.de/DE/Themen/weitereThemen/ITGrundschutzKataloge/Inhalt/GefaehrdungskataloGe/G5VorsaetzlicheHandlungen/g5vorsaetzlichehandlungen_node.html.

[43]. **Bundesamt für Sicherheit in der Informationstechnik.** Baustein 3.208. *Internet-PC.* [Online] 2009. [Zitat vom: 14. November 2011.] https://www.bsi.bund.de/ContentBSI/grundschutz/kataloge/baust/b03/b03208.html.

[44]. **Bundesamt für Sicherheit in der Informationstechnik.** Baustein 3.101. *Allgemeiner Server.* [Online] 2009. [Zitat vom: 14. November 2011.] https://www.bsi.bund.de/ContentBSI/grundschutz/kataloge/baust/b03/b03101.html.

[45]. **Bundesamt für Sicherheit in der Informationstechnik.** Baustein 4.1. *Heterogene Netze.* [Online] 2009. [Zitat vom: 14. November 2011.] https://www.bsi.bund.de/ContentBSI/grundschutz/kataloge/baust/b04/b04001.html.

[46]. **Eckert, Claudia.** *IT-Sicherheit.* München : Oldenbourg Wissenschaftsverlag GmbH, 2009. S. 187.

[47]. **Bundesamt für Sicherheit in der Informationstechnik.** *Webkurs IT-Grundschutz 7.3 Risikoanalyse - traditionelles Vorgehen und Ansatz des BSI.* [Online] 2008. [Zitat vom: 31. August 2011.] https://www.bsi.bund.de/DE/Themen/weitereThemen/WebkursITGrundschutz/Risikoanalyse/Vorgehensweisen/vorgehensweisen_node.html.

[48]. **BITKOM.** Bundesverband Informationswirtschaft, Telekommunikation und neue Medien e.V. *Internet-Kriminalität nimmt weiter zu.* [Online] 2011. [Zitat vom: 27. November 2011.] http://www.bitkom.org/de/presse/30739_68473.aspx.

[49]. **Bundesministerium des Innern.** Bundesgesetzblatt Online. *Verordnung über Personalausweise und den elektronischen Identitätsnachweis.* [Online] 22. April 2010. [Zitat vom: 26. November 2011.] http://www.bgbl.de/Xaver/text.xav?bk=Bundesanzeiger_BGBl&start=%2F%2F*%5B%40attr_id%3D'bgbl110054.pdf'%5D&wc=1&skin=WC.

[50]. **Bundesamt für Sicherheit in der Informationstechnik.** BSI für Bürger. *Der neue Personalausweis - Sicherheitstipps.* [Online] 2011. [Zitat vom: 26. November 2011.] https://www.bsi-fuer-buerger.de/BSIFB/DE/SicherheitImNetz/Personalausweis/Sicherheitstipps/Sicherheitstipps_node.html.

[51]. **Bundesamt für Sicherheit in der Informationstechnik.** BSI für Bürger. *Basisschutz für den Computer.* [Online] 2011. [Zitat vom: 02. Dezember 2011.] https://www.bsi-fuer-buerger.de/BSIFB/DE/MeinPC/BasisschutzComputer/basisschutzComputer_node.html.

[52]. **Nagel, Kurt.** *Nutzen der Informationsverarbeitung.* München : Oldenbourg Verlag, 1990. S. 39, 98. 3486216074.

[53]. **Nagel, Kurt.** *Erfolg: effizientes Arbeiten, Entscheiden Vermitteln und Lernen.* München : Oldenbourg Verlag, 2000. S. 59 ff. ISBN 978-3-486-25616-.

[54]. **Leymann, Frank Prof. Dr.** Gabler. *Wirtschaftlexikon.* [Online] 2010. [Zitat vom: 02. Dezember 2011.] http://wirtschaftslexikon.gabler.de/Definition/hardware-virtualisierung.html.

[55]. **Zhang, Xiaolan, et al., et al.** ACM Digital Library. *XenSocket: A High-Throughout Interdomain Transport for Virtual Machines.* [Online] 2008. [Zitat vom: 01. Dezember 2011.] http://dl.acm.org/.

[56]. **Tanenbaum, Andrew S.** *Moderne Betriebssysteme.* München : Pearson Education Deutschland GmbH, 2009. S. 664f. 978-3-8273-7342-7.

[57]. **Intel Corporation.** Xen Summit April 2007. *Intel Virtualization Technology Roadmap and VT-d Support in Xen.* [Online] 2007. [Zitat vom: 19. November 2011.] http://xen.org/files/xensummit_4/VT_roadmap_d_Nakajima.pdf.

[58]. **Intel Corporation.** Intel Virtualization Technology for Directed I/O (VT-d):. *Enhancing Intel platforms for efficient virtualization of I/O.* [Online] 20. Februar 2009. [Zitat vom: 20. November 2011.] http://software.intel.com/en-us/articles/intel-virtualization-technology-for-directed-io-vt-d-enhancing-intel-platforms-for-efficient-virtualization-of-io-devices/.

[59]. **Invisible Things Lab.** SSTIC 2011. *Symposium sur la sécurité des technologies de l'information et des communications.* [Online] 8-10. Juni 2011. [Zitat vom: 06.

Dezember 2011.] http://www.google.de/url?sa=t&rct=j&q=sstic%202011&source=web&cd=7&sqi=2&ved=0CEUQFjAG&url=http%3A%2F%2Fwww.invisiblethingslab.com%2Fresources%2F2011%2FSSTIC%25202011.pdf&ei=mGDeTvr3GKrc4QT29ZzxBg&usg=AFQjCNGgCSKqHrl94LnXcnrFLvwZmSuQrw.

[60]. **Invisible Things Lab.** *Architektur Qubes-OS.* [Online] 2010. [Zitat vom: 25. Januar 2011.] http://qubes-os.org/files/doc/arch-spec-0.3.pdf.

[61]. **Eckert, Claudia.** *IT-Sicherheit.* München : Oldenbourg Wissenschaftsverlag GmbH, 2099. S. 681.

[62]. **Invisible Things Lab.** Wiki.qubes-os. *Security-Critical Code in Qubes OS.* [Online] 19. September 2011. [Zitat vom: 08. Dezember 2011.] http://wiki.qubes-os.org/trac/wiki/SecurityCriticalCode.

[63]. **Tanenbaum, Andrew S.** *Moderne Betriebssysteme.* München : Pearson Education Deutschland GmbH, 2009. S. 31. 978-3-8273-7342-7.

[64]. **Citrix Systems Inc.** Xen Summit 2009. *Status update of PVUSB.* [Online] 19. November 2009. [Zitat vom: 03. Oktober 2011.] http://www.xen.org/files/xensummit_intel09/PVUSBStatusUpdate.pdf.

[65]. **Invisible Things Lab.** wiki. *Qrexec.* [Online] 30. August 2011. [Zitat vom: 2011. Dezember 03.]

[66]. **Citrix Systems Inc.** *XEN Networking.* [Online] 30. November 2011. [Zitat vom: 01. Dezember 2011.] http://wiki.xen.org/wiki/Xen_Networking.

[67]. **Drakos, Nikos, et al., et al.** Xen v3.0 for x86. *Xen Interface manual.* [Online] The xen Team University of Cambridge, 2002-2005. [Zitat vom: 08. Dezember 2011.] http://www.linuxtopia.org/online_books/linux_virtualization/xen_3.0_interface_guide/index.html.

[68]. **Eckert, Claudia.** *IT-Sicherheit* München : Oldenbourg Wissenschaftsverlag GmbH, 2009. S. 749.

[69]. **Eckert, Claudia.** *IT-Sicherheit.* München : Oldenbourg Wissenschaftsverlag GmbH, 2009. S. 731, 753f.

[70]. **OpenBSD.** *OpenBSD Reference Manual OpenSSH.* [Online] 11. September 2011. [Zitat vom: 19. November 2011.] http://www.openbsd.org/cgi-bin/man.cgi?query=ssh&sektion=1.

[71]. **Invisible Things Lab,; Qubes-OS.** ArchitekturQubes-OS. [Online] 2010. [Zitat vom: 25. Januar 2011.] http://qubes-os.org/files/doc/arch-spec-0.3.pdf.

[72]. **Citrix Systems Inc.** Xen wiki. *XenUSBPassthrough Xen wiki.* [Online] 2010. [Zitat vom: 12. Februar 2011.] http://wiki.xensource.com/xenwiki/XenUSBPassthrough.

[73]. **Invisible Things Lab.** Projektplan - Qubes -. [Online] 09. September 2011. [Zitat vom: 09. Dezember 2011.] http://wiki.qubes-os.org/trac/roadmap.

[74]. **Invisible Things Lab.** *Installation Guide - Qubes OS* -. [Online] 11. April 2011. [Zitat vom: 20. April 2011.] http://wiki.qubes-os.org/trac/wiki/InstallationGuide.

[75]. **Eckert, Claudia.** *IT-Sicherheit.* München : Oldenbourg Wissenschaftsverlag GmbH, 2009. S. 17.

Eidesstattliche Erklärung

Ulrich Gabele, Matrikelnummer: 870 555

Hiermit erkläre ich, dass ich diese Arbeit selbständig abgefasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.
Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Hofheim, den 28. Dezember 2011

Unterschrift (Vor- und Zuname)