Ein OpenID-Provider mit Proxy-Funktionalität für den nPA

Sebastian Feld · Norbert Pohlmann

Institut für Internet-Sicherheit – if(is)
Fachhochschule Gelsenkirchen
[feld | pohlmann]@ internet-sicherheit.de

Zusammenfassung

OpenID ist ein offener, dezentraler und URL-basierter Standard für Single Sign-On (SSO) im Internet. Darüber hinaus wird in Deutschland ab November 2010 der neue, elektronische Personalausweis (nPA) eingeführt. Ziel dieser Arbeit ist eine Sicherheitsbewertung des OpenID-Protokolls sowie die Darstellung, wie ein OpenID-Provider die Restricted Identification (RI) des nPAs mit einer OpenID-Identität verknüpfen kann. Es wird erörtert, welche Mehrwerte die Kombination der beiden Technologien nPA und OpenID erschaffen können.

1 Motivation

In diesem Abschnitt wird die der Arbeit zugrunde liegende Problematik bei Identity Management (IdM) im Internet aufgezeigt und mögliche Lösungsansätze des Problems aufgeführt.

1.1 Identitäten im Internet

Das Internet scheint den klassischen Desktop immer weiter zu verdrängen. Eine Vielzahl an Software wird heute nur noch als Dienstleistung über das Internet angeboten. Die treibenden Konzepte in diesem Bereich sind Software as a Service (Saas) sowie Cloud Computing. Dabei reicht die Bandbreite der Anwendungen von E-Mail-Clients über Office-Paketen bis hin zu sozialen Netzwerken.

Die meisten Internetdienste haben die Forderung einer Identifizierung und Authentisierung gemeinsam. Bei nahezu jedem Dienst findet erst ein Login (die Behauptung und der anschließende Beweis einer Identität) statt, bevor der Dienst genutzt werden kann. Es bestehen hohe Sicherheitsanforderungen von Seiten der Dienstanbieter, aber auch seitens der Nutzer.

Im Einklang mit SaaS und Cloud Computing prägt sich immer mehr der Begriff der digitalen Identität, für den viele unterschiedliche Verständnisse existieren (vgl. bspw. [BSSW10, S. 14ff.]). Im Kontext dieser Arbeit stellt eine digitale Identität einen Identifikator (z.B. ”johnDoe”) zusammen mit einer bestimmten Anzahl an Attributen (Alter, Nationalität, E-Mail-Adresse usw.) dar. Darüber hinaus werden im Internet zahlreiche digitale ”Fußspuren” hinterlassen. Neben Accounts bei Webshops oder E-Mail-Providern, erstellten Profilen auf Plattformen wie Xing oder Facebook sowie veröffentlichten Inhalten auf Webseiten, in Blogs oder bei Twitter können auch Bewegungsinformationen wie Logdaten der Server oder Verbindungsda-

P. Horster (Hrsg.) · D•A•CH Security 2008 · syssec (2008) 1-10.

ten der Internet Service Provider (ISPs) zu einer digitalen Identität gezählt werden. Schließlich können Profile kombiniert werden, was durch gewollte und ungewollte Verlinkungen zwischen den Diensten realisiert wird.

Zusammengefasst lässt sich sagen, dass Internetnutzer eine Vielzahl freiwillig aber auch unfreiwillig preisgegebener Daten besitzen. Ein aktuelles Ziel sollte sein, die digitale Identität der Internetnutzer möglichst stark zu schützen.

1.2 Identifikatoren und das Passwort-Dilemma

In den meisten Fällen steht eine einmalige Registrierung dem Login und der Nutzung eines Dienstes bevor. Ein Nutzer muss mehrere Entscheidungen treffen: Es ist ein Identifikator zu wählen, der u.U. vorgegeben oder bereits belegt sein kann. Ferner muss der Nutzer Credentials (deutsch ”Berechtigungsnachweis”) für einen Dienst festlegen, womit nachfolgend die Identität bewiesen wird. Üblich ist derzeit der Einsatz einer Benutzername-Passwort-Kombination. Bei Passwörtern besteht die Möglichkeit, dass ein Dienstanbieter die Stellenanzahl oder die erlaubten Zeichen des Passworts beschränkt oder gänzlich vorgibt. Nicht zuletzt entscheidet der Nutzer, ob dem Dienstanbieter personenbezogene Daten – insbesondere Credentials – anvertraut werden kann. Bei der kontinuierlich steigenden Anzahl an Internetdiensten endet dies in vielen verstreuten, oft redundaten Identitäten mit verschiedenen Identifikatoren und Passwörtern.

Die beschriebene Problematik wird oft Passwort-Dilemma genannt. Ein Nutzer wählt für die Vielzahl an Internetdienste zu kurze oder zu einfache Passwörter, verwendet der Bequemlichkeit halber gleiche Passwörter für verschiedene Dienste oder schreibt Passwörter auf. Die Verwendung von Passwörtern ist zudem vergleichsweise unsicher, da diese mittels Keylogger, Man-in-the-Middle-Angriffen oder Phishing-Attacken abgegriffen werden können.

1.3 Mögliche Lösungen

Es existieren unterschiedliche Herangehensweisen als Abhilfe für das Passwort-Dilemma. Ein Passwortsafe ist eine Software, die die gesicherte Speicherung der unterschiedlichen und stark gewählten Passwörter übernimmt. Ein Passwortsafe schränkt jedoch die Benutzerfreundlichkeit und Mobilität ein, da er stets mitgeführt werden muss.

Eine Alternative stellt die Technologie Web Single Sign-On (Web-SSO) dar. Bei der ”Einmalanmeldung” existiert nur ein Identifikator und ein stark gewähltes Passwort. Nachteilig ist der Single Point of Failure (der Dienst des Identitätenverwalters) und die akute Gefahr des Phishings. OpenID ist ein Beispiel eines Web-SSO-Protokolls.

Eine dritte Möglichkeit stellt die sogenannte starke Authentisierung dar, bei der mehrere Faktoren – Wissen, Besitz und Eigenschaft – zur Identitätsbestimmung genutzt werden. Ein klassisches Beispiel ist der Einsatz von digitalen Zertifikaten. Eine konkrete Umsetzung dieser Strategie stellt die eID-Funktion des neuen, elektronischen Personalausweises dar.

2 OpenID als SSO-System für das Internet

Dieser Abschnitt stellt das Protokoll OpenID als möglichen Lösungsansatz des Passwort-Dilemmas vor und dient als inhaltliche Grundlage für das beschriebene Lösungskonzept.

2.1 Überblick über das Protokoll

OpenID ist ein offener, dezentraler und URL-basierter Standard für SSO im Internet [RR06]. Bei der seit 2007 in der Version 2.0 vorliegenden Spezifikation [RR07] kann ein Nutzer sowohl die Identität als auch den Identitätenverwalter frei wählen. Die Identifizierung eines Nutzers erfolgt über den Beweis des Besitzes einer URL, der sogenannten OpenID-Identität.

Der große Nutzen von Web-SSO im Allgemeinen und OpenID im Speziellen ist die einmalige Anmeldung beim Identitätenverwalter (OpenID-Provider, OP) und die anschließende Nutzung aller OpenID-unterstützender Dienste (Relying Party, RP). Die Credentials eines Nutzers (Client, C) müssen nicht mehr an vielen Punkten im Internet (bei den RPs) hinterlegt werden, sondern nur noch an einer zentralen und vertrauenswürdigen Stelle (dem OP). Infolgedessen ist die digitale Identität eines Nutzers nicht mehr redundant verteilt, es existiert nur noch ein Identifikator – die OpenID-Identität (Identifier, I).

Die größte Gefahr im Zusammenhang mit OpenID ist die hohe Anfälligkeit für Phishing bei der Verwendung von Passwörtern. Kommt ein Angreifer in den Besitz des Passworts einer OpenID-Identität, so stehen ihm alle angeschlossenen Dienste zur Verfügung. Dies kann durch das erwähnte Phishing geschehen oder durch die Tatsache, dass ein Nutzer nicht gezwungen ist, ein starkes Passwort zu wählen. Ein weiteres Problem ist die Möglichkeit zur Profilbildung seitens des OPs. Der OP kennt sowohl die vom Nutzer eingesetzten Dienste als auch die Frequenz der Nutzung.

2.2 Protokollablauf

Der Protokollablauf von OpenID besteht aus sieben Schritten (vgl. Abbildung 1), die im Folgenden näher beschrieben werden:

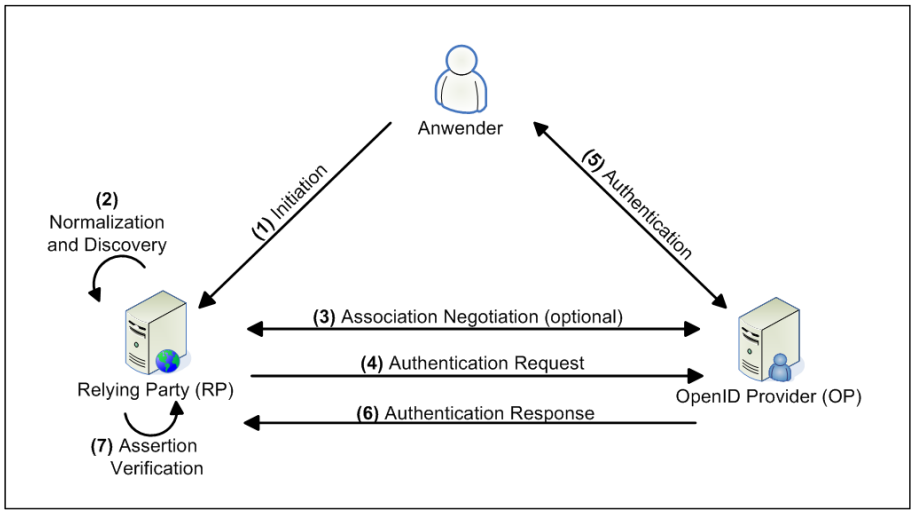

Abb. 1: Protokollablauf OpenID

Initiation
Initiation bezeichnet das Übermitteln des vom Nutzer gewählten Identifikators an die Relying Party, wodurch der Login-Prozess gestartet wird. Ein Nutzer C ruft die Webseite des Internetdienstes RP auf und gibt im Login-Formular statt einer Benutzername-Passwort-Kombination nur die OpenID-Identität (den Identifier) I an. Dies kann bspw.

https://openid.internet-sicherheit.de/johnDoe sein. Das Abschicken des HTML-Formulars an die RP beendet den ersten Schritt.

Normalization/Discovery
Normalization beschreibt den Prozess, bei der die vom Nutzer eingegebene OpenID-Identität in eine standardisierte Form gebracht wird. Bei Discovery ermittelt die Relying Party weitere Informationen über den zuständigen OpenID-Provider. Die RP beginnt mit der Normalisierung der von C eingegebenen Identität I (vgl. [BLFM05, Kap. 6]). Ein Beispiel ist das Ergänzen eines fehlenden Schemas wie z.B. *https://* an *openid.internet-sicherheit.de/johnDoe*. Anschließend führt die RP die Discovery aus, bei der die für die Generierung einer Authentisierungsanfrage benötigten Informationen ermittelt werden. Es bestehen drei Möglichkeiten: XRI-Auflösung [RM05], Yadis-Protokoll [Mil06] oder HTML-basiertes Discovery. Das via Discovery ermittelte XRDS- oder HTML-Dokument enthält Informationen über die Lokation des OpenID-Dienstes seitens des OPs (OP Endpoint URL), die Version des unterstützten OpenID-Protokolls (Protocol Version), den Namen der behaupteten Identität (Claimed Identifier) sowie eine alternative Darstellung des Identifiers (OP-Local Identifier).

Association Negotiation (optional)
Associaton Negotiation baut eine integritätsgesicherte Kommunikationsverbindung zwischen Relying Party und OpenID-Provider auf. RP und OP handeln ein Shared-Secret aus, um nachfolgende OpenID-Nachrichten signieren und verifizieren zu können. Dazu generiert der OP einen MAC-Schlüssel und übermittelt diesen optional verschlüsselt an die RP. Beide Parteien einigen sich auf einen Signieralgorithmus (HMAC-SHA1 oder HMAC-SHA256) sowie auf die Art der Verschlüsselung des MAC-Schlüssels (no-encryption, DH-SHA1 oder DH-SHA256). Die Assoziierungsanfrage von RP an OP hat die Form $AssocRequest(AT, ST, [P, G, DH_{RP}])$, wobei AT und ST die von RP bevorzugten Signier- und Verschlüsselungsalgorithmen sind. Sofern der MAC-Schlüssel verschlüsselt übertragen werden soll ist P die Modulo-Primzahl, G der Diffie-Hellman-Generator und DH_{RP} der öffentliche Schlüssel von RP. Der OP antwortet mit einer Assoziierungsantwort der Form $AssocResponse(AH, T, AT, ST, [K|DH_{OP}, H])$, wobei AH ein Alias der Assoziierung (Association Handle), T die Lebensdauer der ausgehandelten Association in Sekunden und AT bzw. ST die tatsächlich eingesetzten Signier- bzw. Verschlüsselungsalgorithmen sind. Falls keine verschlüsselte Übertragung des MAC-Schlüssel stattfindet ist K das Shared-Secret. Andernfalls ist DH_{OP} der öffentliche Schlüssel des OPs und H das verschlüsselte Shared-Secret. Mittels des Association Handles AH wird im späteren Verlauf auf den MAC-Schlüssel zur Signierung und Verifizierung der OpenID-Nachrichten zugegriffen.

Ist eine RP nicht in der Lage Associations zu erstellen oder zu speichern (Optionalität dieses Schrittes), so kann der sogenannte "Stateless Mode" genutzt werden. Dazu erstellt der OP ein privates Geheimnis zur Signierung der OpenID-Nachrichten. Die RP verifiziert erhaltene Nachrichten über direkte Kommunikation mit dem OP [RR07, Kap. 11.4.2].

Authentication Request
Authentication Request stellt die Anfrage der Relying Party an den OpenID-Provider dar, den Nutzer zu authentisieren. Die RP leitet den User Agent von C mit einer Authentisierungsaufforderung der Form $AuthRequest(I, [AH], RU, R)$ zum OP um. I ist der zu bestätigende Identifier (z.B. *http://openid.internet-sicherheit.de/johnDoe*) und AH der optionale Association Handle. RU repräsentiert die sogenannte return_to-URL, zu die der Nutzer C nach erfolgter Authentisierung geleitet werden soll. Der Realm R ist eine Art URL, für die der aktuelle Request

gültig ist und wird dem Nutzer beim folgenden Schritt präsentiert.

Authentication
Authentication ist die eigentliche Authentisierung des Nutzers. Der OP prüft, ob der Nutzer C im Besitz der OpenID-Identität I ist und ob dieser die aktuelle Authentisierung wünscht. Die Ausprägung der Nutzer-Authentisierung ist im Standard nicht spezifiziert [RR07, Kap. 3]. Die Verantwortung liegt gänzlich beim OP, auf dessen Aussage eine RP vertraut. Die Durchführung der Authentisierung wird faktisch ausgelagert. Der heutzutage übliche Mechanismus zur Authentisierung ist die Benutzername-Passwort-Kombination.

Authentication Response
Authentication Response ist die Antwort des OpenID-Providers an die Relying Party zusammen mit der Aussage, ob die Authentisierung des Nutzers erfolgreich verlief oder nicht. Dazu leitet der OP den User Agent von C mit einer positiven oder negativen Antwort der Form $AuthResponse(I, [AH], N, TS, S, \{Sig\})$ zur RP um. I ist der zu bestätigende Identifier, AH der optionale Association Handle, N eine Nonce, T ein Zeitstempel und S eine Liste der signierten Felder dieser Nachricht. $\{Sig\}$ ist die eigentliche Signatur bestehend aus dem Hashwert der Konkatenation der zu signierenden Werte zusammen mit dem geheimen Schlüssel K. Die Nonce und der Zeitstempel dienen der Verhinderung von Replay-Angriffen.

Assertion Verification
Assertion Verification ist die Integritätsprüfung der empfangenen Authentisierungsantwort durch die Relying Party. Die RP überprüft die Antwort des OPs mittels zuvor ausgehandelter Association (Zugriff über den Association Handle AH) oder direktem Request (Stateless Mode). Im Fall einer korrekten und positiven Antwort des OPs ist der Nutzer C erfolgreich beim Dienst der RP angemeldet.

3 Sicherheitsbewertung

In diesem Abschnitt wird eine Abschätzung der Sicherheit des OpenID-Protokolls über die Grenzen der Spezifikation hinweg gegeben. Erst werden Angriffe, Bedenken und Ideen bezugnehmend auf die sieben Schritte des Protokolls dargestellt und anschließend ergänzende oder grundlegenden Punkte besprochen.

3.1 Identifier Creation

Bevor ein Nutzer ein Web-SSO mittels OpenID nutzen kann, muss dieser eine OpenID-Identität anlegen. Es gestalten sich verschiedene Überlegungen bezüglich der Sicherheit.

Bei der Wahl des OpenID-Providers gilt es zu überlegen, ob dem OP vertraut werden kann. Dieses Problem ist genereller Natur und gilt nicht nur in Bezug auf OpenID, sondern für jegliche Art von Diensten im Internet. Aspekte des Vertrauens betreffen bspw. den Umgang mit den zur Verfügung gestellten personenbezogenen Daten. Einem OpenID-Provider, der ausdrücklich darauf hinweist, dass die hinterlegten Daten keinesfalls anderweitig verwendet werden, sollte mehr vertraut werden als einem OP, der diesen Punkt nicht anspricht. Ein Nutzer sollte zusätzlich ermitteln, welches Geschäftsmodell der OP verfolgt. Ein weiterer Aspekt des Vertrauens ist das Maß der Angreifbarkeit des Dienstes. Ein Nutzer sollte abwägen, ob die angebotenen Sicherheitsfeatures des OPs mit dem eigenen Sicherheitsbewusstsein im Einklang stehen. Die Wahl des OPs ist ein wichtiger Schritt, da dieser fortan für die Sicherheit der eigenen digitalen Identität zuständig ist.

Eine weitere Überlegung betrifft den Identifier. Je nach dessen Wahl (z.B. *https://openid.internet-sicherheit.de/johnDoe*) können Rückschlüsse auf die wahre Identität der Person (hier: John Doe) getroffen werden. Ein Nutzer muss entscheiden, ob dies gewollt ist oder nicht.

Die Eingabe persönlicher Informationen ist ebenfalls ein sicherheitsrelevanter Schritt. Ein OpenID-Provider kann verschiedene Informationen wie Vor- und Zuname, Geburtstag und dergleichen vorhalten und auf Wunsch des Nutzers an Relying Partys übermitteln. Die Protokollerweiterung "OpenID Attribute Exchange" [HBH07] beschleunigt hierdurch bspw. Registrierungsprozesse. Ein Nutzer muss entscheiden, wieviele Daten er beim OP hinterlegt. Eine Möglichkeit ist die Angabe vieler Informationen, woraus ein vollständiges Profil für eine zentrale, digitale Identität entsteht. Alternativ kann ein Nutzer nur ein Pseudonym angeben, falls lediglich die Funktionalität der Authentisierung gewünscht ist.

Schließlich muss die Authentisierungsmethode festgelegt werden. Entscheidet sich ein Nutzer für einen Login mittels Benutzername-Passwort-Kombination, so muss ein sehr starkes Passwort gewählt werden. Das Erraten oder Brechen des Passworts bedeutet eine Übernahme der OpenID-Identität und somit auch der angebundenen Dienste. Beispielhafte Alternativen sind digitale Zertifikate oder die eID-Funktion des neuen Personalausweises. Je nach angebotenen Sicherheitsmechanismen des OPs sind weitere Authentisierungsmethoden denkbar.

3.2 Normalization/Discovery

Eine Relying Party bringt den Identifier in eine standardisierte Form und versucht mittels Discovery notwendige Informationen über den zuständigen OpenID-Provider zu ermitteln. Das hinter der OpenID-Identität hinterlegte XRDS- bzw. HTML-Dokument wird geladen und die entsprechenden Informationen extrahiert.

Es ist offensichtlich, dass für eine Relying Party das Laden von Daten fremder Hosts prinzipiell gefährlich ist. Ein Angreifer kann durch geschickte Wahl des Identifiers verschiedene Aktionen erreichen (vgl. [TT07]). Das wiederholte Angeben eines Hostnamens samt Portnummer kann einen Portscan auf einen beliebigen Host im Internet ausführen. Ein Angreifer erfährt zwar keinen Gewinn, jedoch wird eine ungewollte Aktion von der RP ausgeführt. Desweiteren ist die Möglichkeit eines Denial-of-Service-Angrifs (DoS-Angriff) gegeben, wenn statt ein Identifier eine große Datei oder ein bösartiges Skript eingegeben wird. Ein Angreifer hofft, dass die Relying Party durch das Laden der gesamten Datei oder das Ausführen des Skripts überlastet und den eigentlich vorgesehenen Dienst einschränkt oder komplett verweigert.

Eine Maßnahme gegen diesen Missbrauch ist die Definition von Zeit- und Datenlimits für den Discovery-Prozess. Außerdem sollten nur bestimmte Protokolle erlaubt (HTTP und HTTPS, kein FTP und dergleichen) und die möglichen Ports eingeschränkt werden.

3.3 Association Negotiation

Zum Aufbau einer integritätsgesicherten Kommunikationsverbindung handeln RP und OP ein Shared-Secret (die Association) aus. Der MAC-Schlüssel kann entweder im Klartext oder mit einem Geheimnis verschlüsselt übertragen werden, das zuvor mittels Diffie-Hellman-Algorithmus (DH-Algorithmus) ausgetauscht wurde.

Das unverschlüsselte Übersenden des Shared-Secrets ist anfällig für Man-In-The-Middle-Attacken (MITM-Attacken). Kommt ein Angreifer in den Besitz des MAC-Schlüssels können fortan mitgelesene OpenID-Nachrichten unbemerkt verändert werden. Die OpenID-Spezifikation

definiert, dass die unverschlüsselte Übermittlung des Shared-Secrets nur beim Einsatz einer Verschlüsselung auf Transportebene (bspw. SSL/TLS) geschehen darf (vgl. [RR07, Kap. 8.1.1]). Die verschlüsselte Übertragung mittels DH-Algorithmus ist jedoch ebenfalls anfällig für MITM-Attacken. Ein Angreifer, der in der Lage ist Datenpakete zu verändern, fängt die Nachrichten des DH-Schlüsseltauschs ab und führt den Algorithmus auf beiden Seiten (RP und OP) getrennt durch. Nachfolgende Nachrichten entschlüsselt der Angreifer, liest sie aus und verschlüsselt sie anschließend mit dem passenden DH-Schlüssel neu.

Desweiteren kann ein DoS-Angriff über offene Association Handles ausgeführt werden. OPs halten generierte Associations für eine gewisse Zeit vor, um OpenID-Nachrichten signieren zu können. Ein Angreifer kann mittels einer Vielzahl an Association Requests versuchen, den Normalbetrieb eines OPs zu stören. Darüber hinaus hat das "mutwillige" ungültig machen bestehender Associations keinen direkten Vorteil für einen Angreifer. Es würde jedoch einen Mehraufwand an Rechenzeit für OP und RP bedeuten.

Eine Maßnahme gegen die akute Gefahr des MITM-Angriffs ist die forcierte Nutzung einer Transportverschlüsselung (z.B. SSL/TLS). Es bleibt zu bemerken, dass in diesem Falle die Nutzung des DH-Algorithmus (also das Verschlüsseln des MAC-Schlüssels) als unnötig erscheint und zusätzliche Komplexität bringt (vgl. [TT07]). RP und OP müssen geeignete Beschränkungen und Limits definieren, um den Gefahren DoS-Angriff und mutwilliges ungültig machen der Associations vorbeugend zu begegnen.

3.4 Authentication Request

Authentication Request ist ein simpler HTTP-Redirect des Nutzers von der Relying Party zum OpenID-Provider. Wichtig ist, dass die RP entscheidet, wohin der Nutzer geleitet wird.

Ein Phishing-Angriff durch die RP ist hier möglich. Eine bösartige RP leitet den Nutzer nicht zum "richtigen" OP, sondern zu einer Fälschung, die ebenfalls unter der Kontrolle des Angreifers ist. Das Aussehen des originalen OPs kann mittels Proxying kopiert werden. Der Nutzer gibt beim gefälschten OP die Credentials ein (das Phishing findet statt), wonach eine Übernahme der OpenID-Identität durch den Angreifer möglich ist.

Desweiteren ist ein Phishing-Angriff durch einen böshaften oder kompromittierten URL-Host möglich. Ein Angreifer muss die Angabe des zuständigen OPs in den HTML-Tags der OpenID-Identität austauschen. In disesem Szenario sendet eine RP den Nutzer unwissentlich zu einem falschen OP, der potentiell einen Phishing-Angriff ausführen kann.

Die Wahl einer RP eines übermäßig allgemeinen Realms stellt ein weiteres Sicherheitsrisiko dar. Eine RP teilt dem OP bei jedem Authentication Request eine Art URL (den Realm) mit, für die der Request gültig sein soll. Hierbei kann ein Nutzer bestimmen, dass er dieser RP vertraut und zukünftige Authentication Requests automatisch bewilligt werden. Eine bösartige RP kann bspw. durch den übermäßig allgemeinen Realm "*http://*.de*" erwirken, dass fortan alle Authentication Requests von Relying Partys mit einer deutschen Top-Level-Domain automatisch bewilligt werden.

Die Funktionalität "Immediate Request" der OpenID-Spezifikation beinhaltet ebenfalls ein Sicherheitsrisiko. Die RP kann einem OP mitteilen, dass keine Interaktion mit dem Nutzer stattfinden soll. Ist der Nutzer zum Zeitpunkt des Authentication Requests nicht beim OP angemeldet, so bedeutet dies eine unmittelbare negative Antwort an die RP. Gefährlich ist dieser Mechanismus in Verbindung mit einem bereits bestätigten übermäßig allgemeinen Realm. Die

Möglichkeit einer Cross-Site Request Forgery (CSRF) ist gegeben, was weiter unten näher beschrieben wird.

Die größte Gefahr beim Einsatz von OpenID ist der Phishing-Angriff. Diese Bedrohung kann eliminiert werden, wenn der Nutzer statt einer Benutzername-Passwort-Kombination eine starker Authentisierung verwendet. Die Nutzung der eID-Funktion des neuen Personalausweises ist eine Möglichkeit. Die gezielte Einschränkung der Wildcard * kann einen Angriff mittels übermäßig allgemein gewählten Realms begegnen. Es ist anzuraten, dass das automatisierte Vertrauen von Nutzern nicht verwendet und von OPs nicht angeboten werden sollte. Ein OpenID-Provider sollte außerdem prüfen, ob die im Authentication Request angegebene return_to-URL ein tatsächlicher OpenID-Endpoint der entsprechenden Relying Party ist. Somit wird verhindert, dass eine positive Authentication Reponse an Dritte gelangt.

3.5 Authentication

Authentication ist die eigentliche Authentisierung des Nutzers. Je nach gewählter Methode bestehen unterschiedliche Angriffsmöglichkeiten, die allesamt auf das Abhorchen bzw. Mitlesen des Authentisierungsvorgangs zielen. So ist die derzeit meist gewählte Authentisierungsmethode – die Benutzername-Passwort-Kombination – hochgradig anfällig gegen Malware auf dem Client-PC, Key-Logger, MITM-Attacken und weiteren.

Der Einsatz starker Authentisierung schafft Abhilfe für das Hauptproblem von OpenID ”Phishing”. Ein Angreifer kann das Geheimnis zwischen Nutzer und OpenID-Provider nicht mehr abfangen, wenn der Nutzer bspw. den neuen Personalausweis einsetzt.

Die Erstellung von Bewegungsprofilen ist ein weiteres Problem. Der OpenID-Provider ist die zentrale Stelle für Logins des Nutzers. Ein OP kann potentiell die Aktivitäten des Nutzers im Internet ausspionieren, da er in Kenntnis der verwendeten Dienste und der Frequenz der Nutzung ist.

Die Kenntnis der verwendeten Dienste ist jedoch unvermeidlich, da die RP den OP um Durchführung einer Authentisierung bittet. Diese temporären Daten sind stets vorhanden. Der menschliche Faktor ”Vertrauen” spielt hier eine Rolle. Ein vertrauenswürdiger OpenID-Provider sollte die Nutzer überzeugen, dass die eigene zentrale Stellung nicht zum Missbrauch von Informationen genutzt wird. Mittels verpflichtender Policies, Geschäftsbedingungen und dergleichen sollte ein OP versichern, dass keine Profilbildung unternommen wird. Eine Zertifizierung nach Common Criteria oder die Offenlegung des Quellcodes kann dies unterstreichen.

3.6 Authentication Response

Authentication Response ist wie der Request ein simpler HTTP-Redirect, diesmal ausgehend vom OpenID-Provider in Richtung Relying Party. Der OP sendet den Nutzer mit einer positiven oder negativen Authentisierungsantwort zurück zur ihm übermittelten return_to-URL.

Eine positive Authentisierungsantwort kann Ziel eines Replay-Angriffs werden. Ein Angreifer fängt die Nachricht über eine erfolgreiche Authentisierung (den Redirect) mittels Sniffing ab. Ein erneutes Einspielen dieser Nachricht bewirkt die Authentisierung des Angreifers als Opfer.

Die OpenID-Spezifikation empfiehlt als Maßnahme gegen einen Replay-Angriff den Einsatz von Nonces (number used once) und Timestamps [RR07, Kap. 11.3]. Ein OP integriert Nonces in die Authentication Response, um die Antwort einmalig (unique) zu gestalten. Eine RP akzeptiert Authentication Responses nur, wenn die enthaltene Nonce bislang unbekannt ist. Empfängt

eine RP eine Response mit einer bereits genutzten Nonce, so wird die Nachricht mit Verdacht auf einen Replay-Angriff verworfen. Timestamps können zusätzlich genutzt werden, um den Zeitraum zwischen Authentication Request und Response einzuschränken. Hierdurch werden zu alte Antworten verworfen. Außerdem wird der Zeitraum für das Vorhalten bereits genutzter Nonces verkürzt, was Ressourcen der RP eingespart. Problematisch bei dem Einsatz von Nonces ist die Tatsache, dass ein Angreifer "schneller" als das Opfer sein kann. Ein Angreifer kann beim MITM-Angriff den Redirect des Opfers zur RP abfangen, verwerfen und statt des Opfers ausführen (vgl. [TT07]).

3.7 Assertion Verification

Assertion Verification ist die Integritätsprüfung der Authentisierungsantwort durch die Relying Party. Eine RP akzeptiert eine positive Authentication Response (Assertion) erst, nachdem die Integrität festgestellt wurde.

Eine RP sollte neben der Signaturprüfung die in der Authentisierungsantwort enthaltene Nonce kontrollieren. Wie oben beschrieben darf eine RP keine Antwort akzeptieren, die eine bereits empfangene Nonce beinhaltet. Ein OP muss dementsprechend sicherstellen, dass niemals zwei Assertions mit derselben Nonce erstellt und versendet werden.

Eine RP sollte außerdem testen, ob der antwortende OP autorisiert ist, Assertions für die bestätigte Identität zu geben. Hierzu führt die RP eine erneute Discovery auf den Identifier der Reponse aus. Ist der OP der Authentication Response gleich dem OP der via Discovery ermittelten Informationen, so ist die Authentisierungsantwort legitim.

3.8 Weitere Angriffe, Bedenken und Ideen

Beim Einsatz von OpenID ist die Durchführung einer Cross-Site Request Forgery (CSRF, deutsch "Seiten-übergreifende Aufruf-Manipulation") denkbar. Zwei Voraussetzungen sind für den erfolgreichen Verlauf einer CSRF notwendig: Einerseits muss das Opfer zum Zeitpunkt des Angriffs beim OpenID-Provider angemeldet sein. Andererseits muss das Opfer dem OP bereits bestätigt haben, dass einer bestimmten Relying Party, dem späteren Ziel des Angriffs, vertraut wird und Authentication Requests automatisch bewilligt werden. Der Angreifer leitet das Opfer auf eine präparierte Webseite, in der zwei versteckte iFrames eingebunden sind. Während das erste iFrame das Opfer mittels "Immediate Authentication Request" bei der RP einloggt, führt das zweite iFrame eine Aktion bei der entsprechenden RP ohne Wissen und Zustimmung des Opfers aus. Nutzer sollten die Funktionalität der automatischen Genehmigung von Authentisierungsanfragen nicht in Anspruch nehmen, um sich vor Angriffen dieser Art zu schützen. Ein OpenID-Provider sollte das automatische Bestätigen von Authentication Requests nicht anbieten, wobei jedoch zwischen Komfort und Sicherheit abgewägt wird.

Weitere Angriffe zielen auf das Domain Name System (DNS) ab. Die Namensauflösung wird mehrfach bei der Discovery und den Redirections (Authentication Request und Response) genutzt. Ist ein Angreifer in der Lage bspw. den DNS-Cache des Opfers zu manipulieren, so kann ein augenscheinlich korrekter Redirect den Nutzer zu einer Kopie des OpenID-Providers führen. Ein Phishing-Angriff findet statt. Der Einsatz starker Authentisierung wie die eID-Funktions des nPAs und dergleichen unterbindet das Abgreifen von Credentials.

(Persistentes) Cross-Site Scripting (XSS) ist grundsätzlich denkbar. Einer kompromittierten Relying Party kann ein fremdes HTML-Formular übergelegt werden, wodurch der Nutzer zu einem böshaften OpenID-Provider geleitet wird. Ein kompromittierter OP hingegen kann mittels

falscher Eingabefelder zum Abgreifen von Credentials missbraucht werden. Ein Nutzer sollte sorgfältig überlegen, bei welchen Diensten er eine Authentisierung mittels OpenID nutzt. Darüber hinaus sollten RPs und OPs auf eine fehlerfreie Implementierung der Dienste achten, um Gefahren wie XSS und dergleichen möglichst auszuschließen.

Bei der Implementierung eines OpenID-Providers (aber auch bei der Integration einer OpenID-Schnittstelle in eine Relying Party) müssen verschiedene Aspekte bedacht werden. Die Vorgehensweise beim Recycling von Identitäten eines OPs muss diskutiert werden, um eventuelle Überschneidungen korrekt zu behandeln. Die Dienste eines Nutzers, der eine OpenID-Identität bei einem OP löscht, dürfen von einem weiteren Nutzer, der anschließend genau diesen Identifier registriert, nicht zur Verfügung stehen.

4 Konzeption eines OpenID-Providers mit nPA-Unterstützung

In diesem Abschnitt wird das Konzept der Verknüpfung einer OpenID-Identität mit der Restricted Identification des neuen Personalausweises beschrieben. Außerdem wird auf die verschiedenen Mehrwerte beim Einsatz eines solchen OpenID-Providers eingegangen.

4.1 Der neue Personalausweis (nPA)

Am 1. November 2010 wird in Deutschland der neue Personalausweis (nPA) eingeführt. Das grundsätzliche Ziel ist die Ausweitung der herkömmlichen Nutzung des Personalausweises auf die elektronische Welt und damit die Ermöglichung einer sicheren und rechtsverbindlichen Kommunikation im Internet [Mar09]. Der nPA ist mit einem kontaktlosen Chip (RF-Chip) ausgestattet, der über Funk mit einem RF-Lesegerät kommuniziert. Die drei elektronischen Funktionalitäten des Chips werden nachfolgend kurz erläutert.

Die Funktionalität ePass besteht aus der altbekannten Identitätsfeststellung wie beim aktuellen Personalausweis. Eine Person beweist einer anderen Person, dass sie tatsächlich diejenige ist, die sie vorgibt zu sein. Als ausschließlich hoheitliche Anwendung ist zudem die Biometriefunktion vorgesehen, bei der ein digitales Lichtbild sowie optional zwei Fingerabdrücke auf dem Ausweis hinterlegt werden können. Diese biometriegestützte Identitätsfunktion zusammen mit kryptografischen Mechanismen und optischen Sicherheitsmerkmalen auf dem Kartenkörper sorgen für einen erhöhten Fälschungsschutz [Rei09].

Die Funktion Online-Authentisierung stellt den elektronischen Identitätsnachweis (auch eID-Funktion) dar. Es wird eine gegenseitige Authentisierung zweier Kommunikationspartner über das Internet realisiert, wonach jede Partei weiß, mit wem sie kommuniziert. Das Recht auf informationelle Selbstbestimmung wurde berücksichtigt, da allein der Nutzer entscheidet, ob ein Dienstanbieter auf bestimmte Daten des nPAs zugreifen darf oder nicht [Rei09]. Gedacht ist die Anwendung für eBusiness und eGovernment.

Die Funktionalität qualifizierte elektronische Signatur (QES) gemäß deutschem Signaturgesetz (SigG) ist ebenfalls für eBusiness und eGovernment gedacht. Eine QES stellt das Äquivalent zur eigenhändigen Unterschrift im elektronischen Rechts- und Geschäftsprozess dar [Mar09]. Diese Funktionalität muss auf dem nPA kostenpflichtig aktiviert werden.

4.2 Restricted Identification (RI)

Die Spezifikationen des nPAs sehen das Wiedererkennens eines bereits registrierten Nutzers vor. Die eindeutige Identifizierung mittels Seriennummer des Personalausweises ist rechtlich nicht zulässig, weshalb die sogenannte sektorspezifische Identifikation (Restricted Identification, RI) eingeführt wurde. Die RI besitzt zwei besondere Eigenschaften [BSI10a, Kap. 2.1.5]: Einerseits ist die RI eines Chips innerhalb eines Sektors eindeutig. Das bedeutet, dass ein Nutzer wiedererkannt werden kann, ohne die tatsächliche Identität zu kennen. Andererseits ist es rechnerisch unmöglich, die RI eines Chips zwischen zwei Sektoren zu verbinden. Das bedeutet, dass gesammelte RIs nicht mit denen anderer Sektoren verglichen und somit Verbindungen von Nutzern über Anwendungsgrenzen hinweg hergestellt werden können.

Die Protokolle Chip Authentication (CA) und Terminal Authentication (TA) müssen erfolgreich durchgeführt worden sein, um die RI auslesen zu können. Die Chip Authentication baut einen sicheren Kommunikationskanal zwischen dem Chip des nPAs und dem sogenannten Terminal, der auslesenden Einheit, auf [BSI10a, Kap. 2.1.2]. Das Terminal erhält über dieses Protokoll den Beweis, dass der Chip nicht gefälscht ist. Die Terminal Authentication ermöglicht dem Chip des nPAs die Überprüfung, ob das Terminal die Berechtigung hat, Informationen aus dem nPA auszulesen [BSI10a, Kap. 2.1.3]. Es findet eine gegenseitige Authentisierung statt.

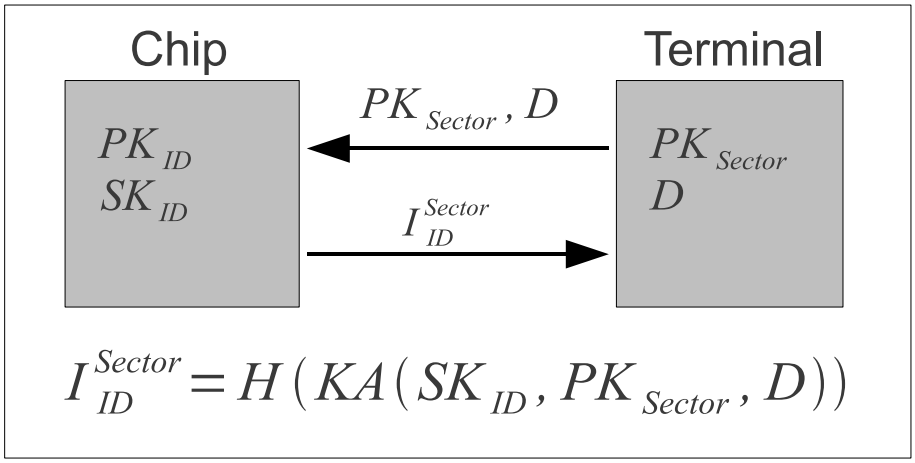

Abb. 2: Protokollablauf Restricted Identification, in Anlehnung an [BSI10a, Abb. 4.4]

Das eigentliche Protokoll zur Berechnung des sektorspezifischen Identifikators I_{ID}^{Sector} ist ein Schlüsselaustausch basierend auf dem Diffie-Hellman-Algorithmus (vgl. Abbildung 2) [BSI10a, Kap. 4.5.1]. Das Terminal sendet den öffentlichen Schlüssel des Sektors PK_{Sector} sowie Domain-Parameter D zum Chip des nPA. Daraufhin überprüft der Chip PK_{Sector} und berechnet mittels der Operation $KA(SK_{ID}, PK_{SECTOR}, D)$ ein gemeinsames Geheimnis K, wobei SK_{ID} der geheime Schlüssel des Chips ist, PK_{Sector} der öffentliche Schlüssel des Sektors und D die Domain-Parameter. Anschließend wird mittels der Operation $H(K)$ ein Hashwert über das Geheimnis K erstellt, was den eigentlichen sektorspezifischen Identifikator I_{ID}^{Sector} darstellt. Dieser wird schließlich zum Terminal gesendet.

4.3 Ein OpenID-Provider mit nPA-Unterstützung

Neben der Sicherheitsanalyse des OpenID-Protokolls wurde ein OpenID-Provider mit Unterstützung des neuen Personalausweises konzipiert und implementiert. Der Realisierung liegen

zwei grundlegende Ideen zugrunde.

Die erste Idee behandelt den Beweis des URL-Besitzes, also dem Vorgang der Authentisierung. Ein Nutzer meldet sich fortan nicht mehr mit der für Brute-Forcing und vor allem Phishing anfälligen Benutzername-Passwort-Kombination beim OP an, sondern mittels starker Authentisierung. Konkret wird hierfür auf die eID-Funktion des nPA zugegriffen. Bei der Registrierung einer OpenID-Identität wird die Restricted Identification (RI) des Chips mit der OpenID-Identität verknüpft. Die einzige aus dem Personalausweis ausgelesene Information – die RI – dient ausschließlich zur Wiedererkennung des Nutzers und verlässt den OP nie.

Die zweite Idee beschreibt die Proxy-Funktion des OPs für den neuen Personalausweis. Über die OpenID-Schnittstelle wird die Nutzung der eID-Funktion für Dienstanbieter ermöglicht, die kein Berechtigungszertifikat besitzen. Ein Nutzer kann eine starke Authentisierung auch bei Dienstanbietern verwenden, die nicht über die nötigen finanziellen oder organisatorischen Ressourcen für den Einsatz der eID-Schnittstelle verfügen. Es sind mehrere Szenarien denkbar: ”Kleine” Internetdienste ohne die nötigen Ressourcen, geschlossene Systeme in Intranets aber auch private Anwendungen wie Blogs können mit einer entsprechenden OpenID-Schnittstelle nPA-tauglich gestaltet werden.

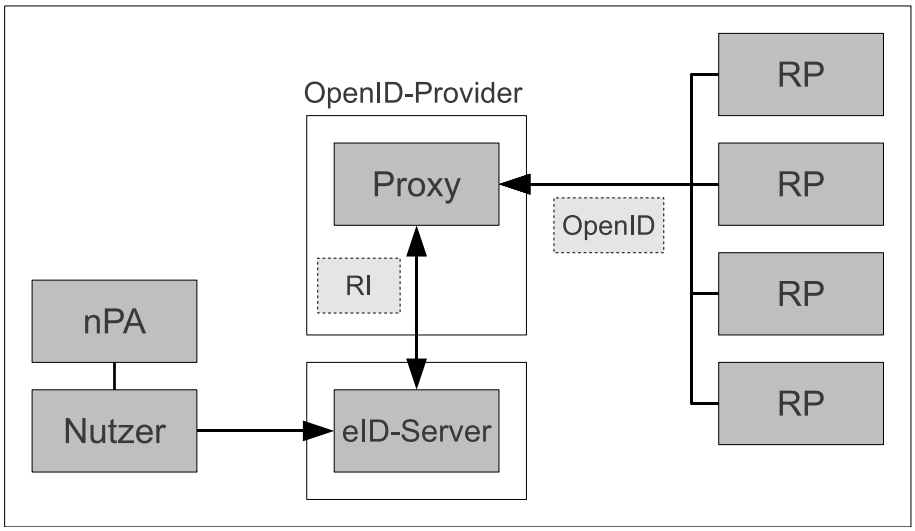

Abb. 3: Schnittstelle Online-Authentication und OpenID

Die schematische Verbindung der RI mit der OpenID-Identität bzw. das Zusammenspiel von Nutzer, OpenID-Provider und Dienstanbietern ist in Abbildung 3 zu sehen.

Abbildung 4 zeigt den Kommunikationsablauf beim Versuch eines Nutzers, sich bei einem Internetdienst mittels OpenID-Identität einzuloggen. Der Nutzer ruft das Login-Formular des Dienstes auf und gibt nur seine OpenID-Identität ein. Nachdem Dienst (RP) und OpenID-Provider (OP) ein Shared-Secret (die Association) ausgehandelt haben, sendet die RP eine Authentisierungsanfrage an den OP. Dieser kontaktiert den eID-Service (Funktion ”useID”) und beantragt das Auslesen der Restricted Identification (RI) aus dem nPA des Nutzers. Der eID-Service antwortet mit Informationen, die der OP direkt an den Nutzer weiterleitet. Die Interaktion zwischen eID-Server und Nutzer findet statt. Dieser bestätigt das Auslesen der RI und gibt eine geheime PIN ein. In regelmäßigen Abständen überprüft der OP, ob das Ergebnis der beantragten Aktion vorliegt (Funktion ”getResult”). Wenn das Ergebnis vorliegt wird die

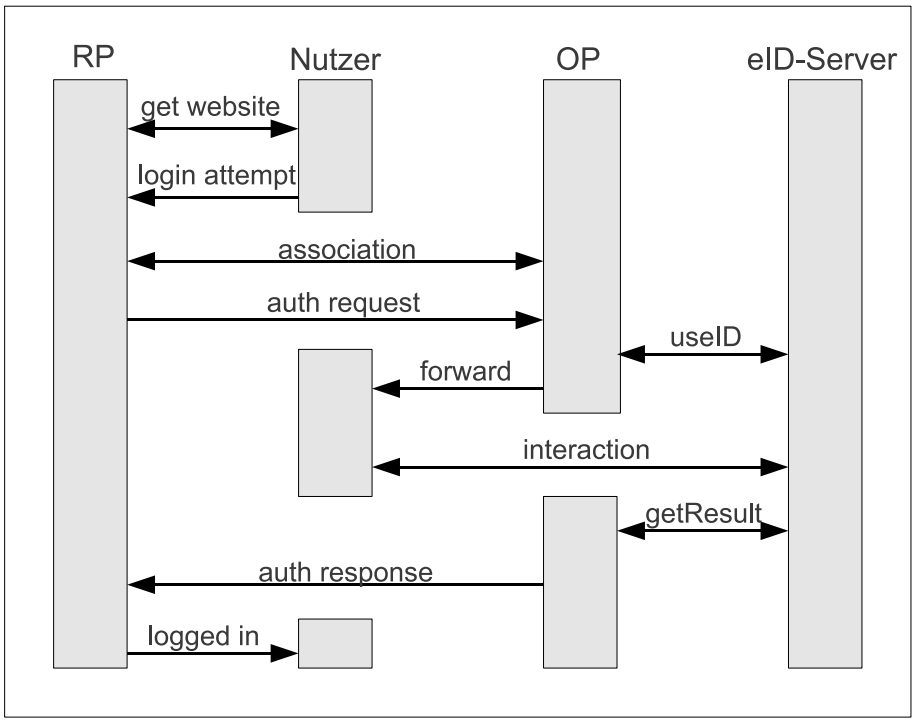

Abb. 4: Ablauf der Kommunikation eines Loginversuchs, in Anlehnung an [BSI10b, Abb. 6]

ermittelte RI zur Authentisierung des Nutzers verwendet und eine positive oder negative Authentisierungsantwort an die RP gesendet. Je nach Ergebnis kann der Dienst den Nutzer fortan als eingeloggt betrachten.

4.4 Mehrwerte der nPA-Unterstützung

OpenID besitzt den generellen Mehrwert, dass ein Nutzer seine Identität nur noch an einer zentralen Stelle beweisen muss. Die Stelle der Authentisierung kann bewusster gewählt und die Bemühung zur sicheren Gestaltung konzentriert werden. Es muss nur noch ein Identifikator und ein Satz an Credentials gesichert werden.

Zudem entstehen durch die Integration der eID-Funktion als Authentisierungsmethode eines OPs Mehrwerte in verschiedene Richtungen.

Aus Sicht des OpenID-Protokolls wird die nicht in der Spezifikation behandelte Authentisierung des Nutzers sicherer gestaltet. Der Nutzer ist durch den Einsatz digitaler Zertifikate nicht mehr in der Lage schwache Passwörter (zu kurz, einfach zu erraten usw.) zu wählen. Desweiteren ist das größte Problem von OpenID ”Phishing” beim Einsatz von Benutzername-Passwort-Kombinationen nicht mehr gegeben. Einerseits wird durch die Multi-Faktor-Authentisierung mittels nPA kein Geheimnis mehr über das Internet versendet. Andererseits findet durch den Einsatz der eID-Funktion eine Authentisierung des OpenID-Providers statt. Nur ein OP im Besitz eines gültigen Berechtigungszertifikats kann Informationen – in diesem Falle die Restricted Identification – aus dem nPA auslesen.

Ein Nutzer erfährt den Mehrwert, dass ihm die Infrastruktur für Web-SSO aber auch für eine Multi-Faktor-Authentisierung zur Verfügung gestellt wird. Da der bereits vorhandene nPA genutzt wird sind keine zusätzlichen SmartCards oder Lesegeräte notwendig.

Ein Vorteil aus Sicht des neuen Personalausweises ist eine gewisse ”Internationalisierung” der eID-Funktion. Grundsätzlich ist der Einsatz des nPA für Anwendungen deutscher Dienstanbieter vorgesehen. Dienstanbieter, die über kein Berechtigungszertifikat verfügen, können den nPA bspw. für die Authentisierung nicht einsetzen. Ein Dienstanbieter muss lediglich eine OpenID-Schnittstelle implementieren, um die Proxy-Funktionalität eines OPs mit nPA-Unterstützung nutzen zu können. Fortan können sich Nutzer, die im Besitz eines deutschen nPAs sind, mithilfe des OpenID-Providers beim entsprechenden (internationalen) Dienst anmelden. Der Dienstanbieter kann keine Informationen aus dem nPA auslesen (auch nicht die RI), sondern nutzt die eID-Funktion indirekt über den OpenID-Provider.

5 Zusammenfassung und Ausblick

Identitäten im Internet, das heisst der Zusammenschluss eines Identifikators mit verschiedenen Attributen und Informationen einer Person, werden in der heutigen Zeit immer wichtiger. Damit einhergehend besteht die Notwendigkeit, dass der Beweis der Identität möglichst sicher gestaltet werden muss. Das sogenannte Passwort-Dilemma muss gelöst werden, wofür viele technische, aber auch organisatorische Ideen bestehen. Offene Standards für Single Sign-On im Internet, wie etwa OpenID, bieten eine einfach Möglichkeit.

Die Schwächen des OpenID-Protokolls können durch die Verwendung der eID-Funktion des neuen, elektronischen Personalausweises kompensiert werden. Der nPA bietet ohnehin bereits die Chance einer sicheren Authentisierung im Internet. Durch die Verknüpfung der beiden Technologien werden Mehrwerte in verschiedener Hinsicht generiert. Ein OpenID-Provider mit Unterstützung einer Authentisierung mittels nPA beseitigt einerseits die größte Gefahr bei OpenID ”Phishing”. Andererseits kann durch die Proxy-Funktionalität eines solchen OpenID-Providers die Verwendung des nPAs ausgeweitet, quasi internationalisiert werden.

Literatur

[BLFM05] T. Berners-Lee, R. Fielding, and L. Masinter. RFC 3986, Uniform Resource Identifier (URI): Generic Syntax. http://www.ietf.org/rfc/rfc3986.txt, 2005.

[BSI10a] BSI. Advanced Security Mechanisms for Machine Readable Travel Documents; Extended Access Control (EAC), Password Authenticated Connection Establishment (PACE), and Restricted Identification (RI); Version 2.03, März 2010. Technische Richtlinie TR-03110.

[BSI10b] BSI. Technische Richtlinie eID-Server; Version 1.3, Juni 2010.

[BSSW10] Georg Borges, Jörg Schwenk, Carl-Friedrich Stuckenberg, and Christoph Wegener. Identitätsdiebstahl und Identitätsmissbrauch im Internet. Technical report, Ruhr-Universität Bochum, Universität des Saarlandes, 2010.

[HBH07] D. Hardt, J. Bufu, and J. Hoyt. OpenID Attribute Exchange 1.0 - Final. http://openid.net/specs/openid-attribute-exchange-1_0.html, December 2007.

[Mar09] Marian Margraf. Der elektronische Identitätsnachweis des zukünftigen Personalausweises. SIT-SmartCard Workshop 2009, Darmstadt, Februar 2009.

[Mil06] Joaquin Miller. Yadis 1.0. http://yadis.org/wiki/Yadis_1.0_%28HTML%29, March 2006.

[Rei09] Andreas Reisen. Die Architektur des elektronischen Personalausweises. 11. Deutscher IT-Sicherheitskongress des BSI, Bonn-Bad Godesberg, Mai 2009.

[RM05] D. Reed and D. McAlpin. Extensible Resource Identifier Syntax 2.0, OASIS Committee Specification, OASIS XRI Technical Committee. http://www.oasis-open.org/committees/download.php/15377, November 2005.

[RR06] David Recordon and Drummond Reed. OpenID 2.0: a platform for user-centric identity management. In *DIM '06: Proceedings of the second ACM workshop on Digital identity management*, pages 11–16, New York, NY, USA, 2006. ACM.

[RR07] David Recordon and Drummond Reed. OpenID Authentication 2.0 - Final. http://openid.net/specs/openid-authentication-2_0.html, December 2007.

[TT07] E. Tsyrklevich and V. Tsyrklevich. Single Sign-On for the Internet: A Security Story. BlackHat USA, 2007.