Stephan Neumann, Melanie Volkamer, Moritz Strube, Achim Brelle, Wolfgang Jung

Cast-as-intended-Verifizierbarkeit für das Polyas-Internetwahlsystem

Dieser Artikel erscheint in der Zeitschrift Datenschutz und Datensicherheit.

Internet-Wahlsysteme müssen einer Vielzahl von Anforderungen genügen.[1] Besonders drei Eigenschaften – die Übereinstimmung von Stimmabgabe und Wählerintension, die korrekte (und vertrauliche) Speicherung der Stimmabgabe und die fehlerfreie Auszählung – sollte für ein Wahlsystem nachweisbar sein. Der vorliegende Beitrag stellt Ansätze zur Realisierung einer nachweisbaren Übereinstimmung von Stimmabgabe und Wählerintension (*Cast-as-intended-Verifizierbarkeit*) vor und zeigt, wie das verbreitete Polyas-Internetwahlsystem um diese Eigenschaft erweitert werden kann.

[1] Siehe z. B. das Schwerpunktheft DuD 2/2009.

1 Einleitung

Das Thema Internetwahlen gewinnt als zeitgemäße Alternative zu Präsenz- und Briefwahlen weiter an gesellschaftlicher Bedeutung. Neben der rein wissenschaftlichen Betrachtung von Internetwahlsystemen findet auch deren Anwendung zunehmend Zuspruch. Die Anwendung von Internetwahlsystemen reicht dabei von Wahlen und Abstimmungen in Vereinen bis hin zu politischen Wahlen, zum Beispiel in Estland und der Schweiz. Ein in Deutschland eingesetztes Internetwahlsystem ist das von der Polyas GmbH angebotene Polyas-System. Damit konnten bis zum heutigen Tag mehr als 2.200.000 Stimmen (gemäß Angaben der Polyas GmbH) abgegeben werden. Zu den bedeutendsten Wahlen, die mit Polyas durchgeführt werden, zählen die jährlichen Wahlen der Gesellschaft für Informatik e.V. (seit 2004) und Wahlen der Deutschen Forschungsgemeinschaft (2007 und 2011, 2015 in Vorbereitung). Insbesondere ist hier auch die erste erfolgreich durchgeführte rechtsverbindliche Internetwahl in Deutschland zu erwähnen, die Wahl der Initiative D21.

Durch das Wahlcomputer-Urteil des BVerfG aus dem Jahr 2009 kommt der Öffentlichkeit und Nachvollziehbarkeit elektronisch gestützter Wahlen eine besondere Bedeutung zu. Diese Nachvollziehbarkeit kann aus technischer Sicht durch Verifizierbarkeit umgesetzt werden. Dabei werden drei Untereigenschaften der Verifizierbarkeit gefordert: Die erste und zweite Eigenschaft ermöglichen es den Wählern die korrekte Bearbeitung ihrer Stimme an ihren Endgeräten (*Cast-as-intended*) und die korrekte Übermittlung an sowie die korrekte Speicherung ihrer Stimme in zentralen Systemkomponenten (*Stored-as-cast*) überprüfen zu können. Der dritte Schritt der Verifizierbarkeit sichert die korrekte Auszählung der gespeicherten Stimmen (*Tallied-as-stored*). Systeme, die alle Eigenschaften besitzen werden unisono als Ende-zu-Ende-verifizierbare Systeme bezeichnet; Systeme, die eine oder zwei der drei Eigenschaften besitzen, werden als partiell verifizierbare Systeme bezeichnet.

Während die Systeme, die zum Zeitpunkt des Urteils im Einsatz waren, keine Verifizierbarkeit boten, findet man in der wissenschaftlichen Literatur eine Reihe Ende-zu-Ende-verifizierbarer Systeme, siehe zum Beispiel [1, 2], sowie eine Reihe partiell-verifizierbarer Systeme, siehe zum Beispiel [3, 4] [2]. Dabei hängt die Umsetzung der Verifizierbarkeit nicht nur von systemischen Aspekten ab, sondern ebenso von der konkreten Einbettung des Systems in dessen Einsatzumgebung. Allerdings belegen wissenschaftliche Arbeiten auch, dass die Effektivität der eingesetzten Verifizierbarkeitsmechanismen unzulänglich ist, beispielsweise bedingt durch fehlende Benutzbarkeit [5] oder die Nicht-Vertrauenswürdigkeit der Endgeräte [6]. Eine Reihe wissenschaftlicher Arbeiten adressiert die Herausforderung der Effektivität von Ende-zu-Ende-verifizierbaren Ansätzen, siehe zum Beispiel [6, 7]. Die Umsetzung dieser Ansätze bringt jedoch neue Annahmen an die Einsatzumgebung mit sich, an deren Umsetzung der Wähler fast immer beteiligt ist.

Motiviert durch die Diskussionen rund um das Wahlcomputer-Urteil und die Forderung der Wissenschaftler nach Ende-zu-Ende Verifizierbarkeit stieg man in vielen Kontexten in der jüngeren Vergangenheit auf partielle Verifizierbarkeit um. So fordert die Schweizer Gesetzgebung seit dem 08. März 2015 den Einsatz individuell verifizierbarer Systeme (Eigenschaft 1 und 2 der Verifizierbarkeit).[3] *Cast-as-intended*-verifizierbare Systeme kamen bereits bei den Kommunalwahlen in Estland im Oktober 2013 zum Einsatz [8] sowie bei den Kommunal- und Landkreiswahlen in Norwegen im September 2011 [9].

Im Fokus dieser Arbeit steht das in Deutschland vielfach eingesetzte Polyas-Internetwahlsystem. Eine Besonderheit des Polyas-Systems besteht darin, dass es stets dahingehend erweitert wird, um dem zunehmenden Anspruch wissenschaftlicher Betrachtungen Rechnung zu tragen. So wurde 2011 in einer Zusammenarbeit von Forschern und den Entwicklern von Polyas eine Erweiterung des Polyas-Internetwahlsystems vorgeschlagen, die dem System die dritte Eigenschaft der Verifizierbarkeit verleiht [4]. Ein Erfahrungsbericht zum Einsatz des partiell verifizierbaren Polyas-Systems für die GI Wahlen wurde 2012 publiziert [10]. Aktuelle wissenschaftliche Literatur sowie große Anwendungsfälle internetbasierter Wahlen widmen sich zunehmend dem ersten Aspekt (Cast-as-intended) der technischen Verifizierbarkeit, da diesem im Hinblick auf die weite Verbreitung von Schadsoftware ein besonderer Stellenwert zukommt. Ziel dieses Beitrags ist die Erläuterung einer Erweiterung des Polyas-Systems zur Umsetzung der *Cast-as-intended*-Verifizierbarkeit. Aktuell befindet sich Polyas im abschließenden Begutachtungsprozess für die Zertifizierung nach dem *Common Criteria Schutzprofil für Basissatz von Sicherheitsanforderungen an Online-Wahlprodukte* [11]. Für die Erweiterung des Polyas-Systems ist auf Grund dieser anstehenden *Common-Criteria*-Zertifizierung des Polyas-Systems und der notwendigen Rezertifizierung eine Nebenbedingung, dass das Gesamtsystem möglichst geringfügig geändert wird.

Die Umsetzung der *Cast-as-intended*-Verifizierbarkeit kann auf verschiedene Weisen erfolgen, zum Beispiel in Form einer iterativen *Cut-and-choose*-Prüfung [12] (wie zum Beispiel in Helios [1] implementiert), über logisch getrennte Geräte (wie in Estland) oder über so genannte Bestätigungscodes (wie in Norwegen).

Abschnitt 2 führt in das Polyas-System ein, beschreibt Erweiterungen des Systems und liefert ein fundamentales Verständnis von der Sicherheit des Systems, sowie den Schwachstellen aufgrund des Fehlens der *Cast-as-intended*-Verifizierbarkeit. Abschnitt 3 fasst bestehende Ansätze zur Umsetzung der *Cast-as-intended*-Verifizierbarkeit zusammen und widmet sich im Folgenden der Entwicklung einer Erweiterung des Polyas-Systems zur Umsetzung dieser Verifizierbarkeit. Abschnitt 4 belegt, wie die Schwachstellen des Ausgangssystems durch die vorgestellte Erweiterung geschlossen werden.

2 Polyas-Internetwahlsystem

2.1 Beschreibung

Das Polyas-Wahlsystem ist Gegenstand mehrerer wissenschaftlicher Arbeiten, z.B. [4, 10, 13, 14]. Die folgende Be-

[2] Fokus dieser Arbeit sind Internetwahlsysteme, so dass (partiell oder ende-zu-ende-verifizierbare) Systeme, die sich auf den Einsatz in Wahllokalen beschränken nicht betrachtet werden. Solche Systeme sind beispielsweise Scantegrity II, Prêt-à-Voter und Bingo Voting.

[3] http://www.admin.ch/aktuell/00089/index.html?lang=de&msg-id=55727 (Stand 25.03.2015)

schreibung ist an die Beschreibung vorheriger Arbeiten [4, 13] angelehnt. Das Polyas-Wahlsystem verteilt die Pflichten (auch Separation of Duty genannt) des gesamten Wahlablaufs auf mehrere (teilweise) unabhängige Instanzen: Erzeugungsserver, elektronisches Wählerverzeichnis, Validierungsserver, Druckdienstleister, Urnenserver und Auszählkomponente.

In der Wahlvorbereitungsphase erzeugt der *Erzeugungsserver* eine Liste von Wahl-TANs (*transaction number*) für alle Wahlberechtigten. Jede TAN wird anschließend gehasht und einem Wähler zugewiesen. Die Liste der Wähler-IDs (ID, über die der Wähler bei dem wahlausrichtenden Zusammenschluss identifiziert werden kann) sowie Hashwerte der zugehörigen TANs werden zur weiteren Verwaltung an das *elektronische Wählerverzeichnis* übermittelt. Die Hashwerte der erzeugten TANs werden zusätzlich an den *Validierungsserver* übermittelt. Die Wählernamen sowie -adressen werden zusammen mit den entsprechenden TANs verschlüsselt an den *Druckdienstleister* übermittelt, der die Authentifizierungsmaterialien für die Wähler vorbereitet und an die Wähler (auf dem Postweg) verschickt.

In der Vorbereitungsphase erzeugt die Auszählkomponente ein asymmetrisches Schlüsselpaar. Der öffentliche Schlüssel wird während der Wahlphase zum Verschlüsseln der Wählerstimmen verwendet. Der private Schlüssel wird in verschlüsselter Form gespeichert, so dass die Entschlüsselung dieses Schlüssels nur unter Eingabe zweier Passwörter durchgeführt werden kann. Diese Passwörter werden von unabhängigen Wahloffiziellen verwaltet. Die elektronischen Stimmzettel werden auf den *Urnenserver* gespielt. Schließlich tauschen alle miteinander kommunizierenden Wahlinstanzen ihre öffentlichen Schlüssel untereinander aus, um eine gesicherte Kommunikation während der Wahl zu ermöglichen.

Zu Beginn seiner Wahlhandlung besucht der Wähler die Webseite des elektronischen Wählerverzeichnisses. Der Wähler authentifiziert sich mit seiner Wähler ID sowie der erhaltenen Wahl-TAN. Die Wahlberechtigung des potentiellen Wählers wird vom elektronischen Wählerverzeichnis durch Berechnung des Hash-Wertes der abgegebenen TAN sowie durch Vergleich mit dem Eintrag in der internen Datenbank überprüft. Stellt das elektronische Wählerverzeichnis die Wahlberechtigung des potentiellen Wählers fest, so wird die TAN an den Validierungsserver weitergeleitet. Ist die TAN in der Datenbank des Validierungsservers enthalten, so erzeugt der Validierungsserver ein Wahltoken und übermittelt dieses Wahltoken sowohl an das elektronische Wählerverzeichnis als auch an den Urnenserver. Das elektronische Wählerverzeichnis leitet das erzeugte Wahltoken an den Wähler weiter, der anschließend automatisch an den Urnenserver weitergeleitet wird. Der Urnenserver prüft die Wahlberechtigung des Wählers, indem das Wahltoken des Wählers mit der Liste der intern gespeicherten Token abgeglichen wird. Ist der Wähler wahlberechtigt, so wird dem Wähler der Stimmzettel übermittelt. Der Wähler füllt den Stimmzettel entsprechend seiner Präferenzen aus und übermittelt den Stimmzettel an den Urnenserver.

Zur Umsetzung eines Übereilungsschutzes wird dem Wähler der ausgefüllter Stimmzettel erneut angezeigt und verschlüsselt zwischengespeichert. Bestätigt der Wähler die Stimmabgabe, so löscht der Urnenserver das Wahltoken. Abgegebene Stimmen werden in zufälliger Reihenfolge in Blöcken von 30 Stimmen gespeichert. Nach Vervollständigung eines Blocks wird der Hashwert des entsprechenden Blocks vom Urnenserver signiert und zur Integritätssicherung an das elektronische Wählerverzeichnis übermittelt.

Am Ende der Wahl werden die verschlüsselten Stimmen des Urnenservers auf die *Auszählkomponente* übertragen. Nach Eingabe der Zugriffsdaten durch zwei Wahloffizielle werden die verschlüsselten Stimmen entschlüsselt.

Zur Umsetzung der universellen Verifizierbarkeit wurde ein Programm entwickelt [4], das nach Eingabe der im Urnenserver gespeicherten Stimmen, der signierten Hashwerte des elektronischen Wählerverzeichnisses und des privaten Schlüssels der Auszählkomponente die Entschlüsselung der Auszählkomponente simuliert, die abgegebenen Stimmen gegen die zwischengespeicherten Hashwerte des elektronischen Wählerverzeichnisses prüft und somit die Bestätigung der korrekten Auszählung unabhängig ermöglicht (hierzu sind das offizielle Ergebnis mit dem des Verifizierungsprogramms zu vergleichen).

2.2 Kryptographische Grundlagen und Problemstellung

Zahlreiche Internetwahlsysteme, unter ihnen das Polyas-Internetwahlsystem, basieren auf dem Einsatz asymmetrischer Kryptosysteme. Derartige Systeme zeichnen sich dadurch aus, dass zwei voneinander verschiedene, jedoch abhängige Schlüssel existieren, ein öffentlicher und ein privater Schlüssel. Der öffentliche Schlüssel ist öffentlich bekannt und kann von jeder Entität zur Verschlüsselung geheimer Nachrichten verwendet werden. Der private Schlüssel ist in der Regel nur einer Person bekannt[4] und kann zur Entschlüsselung von Nachrichten verwendet werden, die mit dem zugehörigen öffentlichen Schlüssel verschlüsselt wurden. Asymmetrische Kryptosysteme werden weiter unterteilt in deterministische und probabilistische Systeme. Die mehrfache Verschlüsselung ein und derselben geheimen Nachricht mithilfe deterministischer Verfahren führt stets zu demselben Schlüsseltext. Im Gegensatz dazu führt die mehrfache Verschlüsselung ein und derselben Nachricht mithilfe probabilistischer Verfahren zu unterschiedlichen Schlüsseltexten. Dies wird dadurch erreicht, dass jeder Verschlüsselungsoperation ein Randomisierungsfaktor hinzugegeben wird, der bei der Entschlüsselung automatisch entfernt wird. Aufgrund der Tatsache, dass Internetwahlsysteme in der Regel eine fest vorgegebene Liste von Wahloptionen beinhalten, würde die Verschlüsselung einer bestimmten Option mit einem deterministischen Verfahren stets zu demselben Schlüsseltext führen. Damit würde der abgegebene Schlüsseltext eines Wählers bereits Informationen über seine Wahl preisgeben. Daher basieren Internetwahlsysteme fast immer auf probabilistischen Verfahren, so auch das Polyas-System.

Insbesondere im Umfeld internetbasierter Wahlsysteme geht der Einsatz probabilistischer Kryptosysteme mit einer besonderen Herausforderung einher. So ist es nach Eingabe der Stimme über eine Benutzerschnittstelle dem Wähler nicht direkt ersichtlich, ob tatsächlich seine Präferenz verschlüsselt wurde oder sein womöglich korrumpiertes Endgerät eine andere Option verschlüsselt hat. *Cast-as-intended*-Verifizierbarkeit setzt genau an dieser Stelle an: Sie bietet dem Wähler die

[4] Es gibt auch kryptographische Techniken, die den privaten Schlüssel auf mehrere Entitäten aufteilen.

Möglichkeit, die korrekte Verschlüsselung seiner Präferenz zu verifizieren.

3 Cast-as-intended-Verifizierbarkeit in Polyas

3.1 Ansätze zur Umsetzung der Cast-as-intended-Verifizierbarkeit

Sowohl die wissenschaftliche Literatur als auch die Praxis haben diverse Ansätze zur Umsetzung der *Cast-as-intended*-Verifizierbarkeit entwickelt. Wir werden die bedeutendsten Ansätze in diesem Abschnitt kurz vorstellen. Im Anschluss daran werden die Ansätze auf ihre Tauglichkeit zur Umsetzung der *Cast-as-intended*-Verifizierbarkeit in Polyas beleuchtet.

3.1.1 Unabhängige Geräte

Der erste Ansatz beruht auf der Idee, ein oder mehrere unabhängige Geräte zur Prüfung der korrekten Verschlüsselung durch das Endgerät des Wählers einzusetzen. Ein derartiger Ansatz kommt seit 2013 im estnischen Internetwahlsystem [8] zum Einsatz, das wir hier beispielhaft anführen. Nachdem der Wähler seiner Präferenz über die Benutzerschnittstelle Ausdruck verliehen hat, verschlüsselt die lokale Wahlapplikation die Stimme des Wählers probabilistisch und speichert den verwendeten Randomisierungsfaktor. Der Wähler signiert den Schlüsseltext, woraufhin dieser an die zentralen Komponenten des Wahlsystems übermittelt wird. Das System weist dem Schlüsseltext einen eindeutigen Wert zu, unter dem der Schlüsseltext im Urnenserver gespeichert wird. Dieser Wert wird dem Wähler zurückgeliefert. Der Wert sowie der Randomisierungsfaktor werden dem Wähler über die Benutzerschnittstelle in Form eines QR-Codes dargestellt. Der Wähler hat die Möglichkeit diesen QR-Code mithilfe einer speziellen Smartphone-Verifizierungsapplikation einzulesen. Die Applikation interpretiert den QR-Code und stellt eine Anfrage an den Urnenserver, den gespeicherten Schlüsseltext an der Stelle des eindeutigen Wertes zurückzuliefern. Nach Erhalt des Schlüsseltextes vom Urnenserver verschlüsselt die Applikation alle möglichen Wahloptionen mit dem erhaltenen Randomisierungsfaktor und vergleicht das Ergebnis mit dem vom Urnenserver erhaltenen Schlüsseltext. Kommt es zu einer Übereinstimmung, wird dem Wähler die entsprechende Wahloption dargestellt. Der Wähler prüft diese Option gegen seine Präferenz und kann somit sicherstellen, dass sein Endgerät die Stimme korrekt bearbeitet hat.

3.1.2 Iterative Cut-and-choose-Prüfung

Ein weiterer Ansatz zur Umsetzung der *Cast-as-intended*-Verifizierbarkeit beruht auf der Arbeit von Benaloh [12] und kommt in dem wissenschaftlich anerkannten Helios-System [1] zum Einsatz. Grundidee dieses Ansatzes ist, die Wahloption des Wählers zu verschlüsseln und ihm einen kryptographischen Hashwert[5] seines Schlüsseltextes zu präsentieren. Der Wähler hat dann die Möglichkeit diesen Schlüsseltext als valide Stimme zu verwenden und diese abzugeben oder die Korrektheit der Verschlüsselung zu prüfen (*cut-and-choose*). Entscheidet sich der Wähler für eine Prüfung, so liefert das System den Randomisierungsfaktor der Verschlüsselungsoperation zurück und der Wähler kann die Verschlüsselungsoperation nachrechnen und anschließend den kryptographischen Hashwert der Verschlüsselungsoperationen gegen den vom Wahlsystem präsentierten Hashwert prüfen. Stimmen die Werte überein so ist der Wähler davon überzeugt, dass die Operation korrekt durchgeführt wurde. Der Wähler kann den geprüften Schlüsseltext nicht mehr zur validen Stimmabgabe verwenden, da er sonst einen Beweis über die abgegebene Stimme hätte und somit Ziel von Erpressungsangriffen werden könnte oder seine Stimme verkaufen könnte. Daher wiederholt der Wähler diesen *Cut-and-choose*-Prozess so oft bis er davon überzeugt ist, dass das System seine Stimme stets korrekt verschlüsselt. Dann erst entscheidet er sich eine nicht geprüfte Stimme abzugeben.

3.1.3 Bestätigungscodes

Der dritte Ansatz ist eine spezielle Ausprägung so genannter Code-basierter Internetwahlsysteme, z. B. [9, 15, 16]. Bestätigungscodes fanden beispielsweise im Norwegischen Internetwahlsystem Anwendung. Grundidee dieses Ansatzes ist es, dem Wähler vorab und unabhängig von dessen Endgerät eine Liste von Codes zukommen zu lassen, die einen eindeutigen Bezug zwischen Wahloptionen und Codes herstellen. Nachdem der Wähler seine Präferenz über das eigentliche Internetwahlsystem in verschlüsselter Form abgegeben hat, wird seine Wahl von den zentralen Komponenten des Systems interpretiert.[6] Die abgegebene Wahloption wird anschließend auf den eindeutigen Bestätigungscode des Wählers abgebildet und dem Wähler zurückgeliefert. Aufgrund der Tatsache, dass die Bestätigungscodes dem Endgerät des Wählers nicht bekannt sind, kennt dieses nur den Code der tatsächlich interpretierten Wahloption und könnte keinen anderen Code ableiten, für den Fall, dass das Endgerät eine andere Stimme abgegeben hätte.

3.1.4 Tauglichkeit der Ansätze

Jeder der hier dargelegten Ansätze geht mit Vor- und Nachteilen einher. So ist der Einsatz unabhängiger Geräte zwar durchaus komfortabel, erfordert jedoch den Besitz eines unabhängigen Gerätes, z. B. Smartphones, und erfordert entsprechend eine Umsetzung entsprechender Software-Lösungen für diese Geräte. Darüber hinaus erlaubt die nachträgliche Überprüfung der abgegebenen Stimme dem Wähler einen Beweis über seine eigene Stimme zu erstellen und somit seine Stimme zu verkaufen. Zur Reduzierung der Gefahr eines Stimmenverkaufs kann zwar die Möglichkeit zur Verifikation zeitlich beschränkt werden. Dies führt jedoch zu einer Verringerung des Integritätsschutzes.

Die Anwendung iterativer *Cut-and-choose*-Prüfungen ist aus kryptographischer Sicht ein interessanter und etablierter Ansatz, da neben der Prüfung der korrekten Verschlüsselungsoperation auch die Geheimhaltung der Wahl in einem hohen Maß umgesetzt werden kann: Ein Wähler erhält niemals einen Beleg, der als Beweis für seine abgegeben Stimme genutzt werden kann. Der wesentliche Nachteil dieses Ansatzes ist die Komplexität der Prüfung und die Tatsache, dass eine derartige Prüfung auch über ein anderes Gerät durchgeführt werden soll.

[5] Vereinfacht ausgedrückt bildet eine kryptographische Hashfunktion einen beliebig großen Wert auf einen Wert fester Länge ab, so dass es schwierig ist aus dem Ergebnis einen anderen Eingabewert zu finden, der auf denselben Wert abbildet.

[6] Die Interpretation der verschlüsselten Stimme kann unter Berücksichtigung verschiedener Sicherheitseigenschaften durchgeführt werden. Der Leser sei an dieser Stelle auf das ehemals eingesetzte norwegische Internetwahlsystem verwiesen.

Es soll hier erwähnt werden, dass durchaus Ansätze existieren, die diese iterative *Cut-and-choose*-Prüfung über unabhängige Geräte implementieren [17]. Die Komplexität des iterativen *Cut-and-choose*-Ansatzes bleibt damit dennoch erhalten, so dass die praktische Bedeutung dieser Ansätze gering sein wird, wie Studien belegen [5]. Letztlich spricht gegen die ersten beiden Ansätze die Tatsache, dass die Wahlprozesse eine umfangreiche Änderung erfahren würden, was der Vorgabe einer geringfügigen Änderung des Gesamtsystems widersprechen würde.

Zur weitgehenden Erhaltung des ursprünglichen Prozesses und somit einer vereinfachten Rezertifizierung des Polyas-Wahlsystems wurde folglich entschieden, Bestätigungscodes als Grundlage der *Cast-as-intended*-Verifizierbarkeit in Polyas heranzuziehen.

3.2 Umsetzung der Cast-as-intended-Verifizierbarkeit in Polyas

In der Wahlvorbereitungsphase erzeugt der Erzeugungsserver für jeden Wähler eine geordnete Liste zufälliger Codes. Jeder Wahlmöglichkeit wird ein eindeutiger Bestätigungscode zugeordnet. Des Weiteren erzeugt der Erzeugungsserver für jeden Wähler einen Offline-Authentifizierungscode. Der Erzeugungsserver übermittelt diese persönlichen Codes an den Druckdienstleister. Der Offline-Authentifizierungscode wird unter einem Rubbelfeld gedruckt. Die Liste der Codes wird zusammen mit den Authentifizierungsmaterialien an den Wähler übermittelt.

Nach Beginn seiner Wahl erhält der Wähler zusätzlich zu dem Wahltoken eine verschlüsselte Liste von Bestätigungscodes. Diese Codes sind mit dem öffentlichen Schlüssel des Urnenservers verschlüsselt und werden vom Wähler unbearbeitet zusammen mit seiner Stimme an den Urnenserver übertragen. Der Urnenserver entschlüsselt die erhaltene Liste, interpretiert die Wahl des Wählers und sendet den Bestätigungscode an der Stelle der entsprechenden Wahloption an den Wähler zurück. Der Polyas-Prozess ändert sich dahingehend, dass einerseits das Wählerverzeichnis die Relation zwischen Wähleridentität und Wahltoken vorbehält, andererseits der Urnenserver die Relation zwischen Wahltoken und abgegebener Stimme vorbehält. Nach Erhalt des Bestätigungscodes prüft der Wähler diesen Code gegen den Bestätigungscode innerhalb der Liste, die er zusammen mit seinen Authentifizierungsmaterialien erhalten hat. Stimmt der Code überein so ist der Wähler davon überzeugt, dass seine Präferenz unverändert an den Urnenserver übermittelt wurde und führt keine weiteren Aktionen durch. Wählerverzeichnis und Urnenserver verwerfen die zwischengespeicherte Relation nach einer vorgegebenen Zeit (zum Beispiel 1 Stunde) und der Urnenserver speichert die abgegebene Stimme zur Auszählung. Stimmt der Code nicht mit der Erwartung des Wählers überein, so kann der Wähler einen Service des Wählerverzeichnisses in Anspruch nehmen. Der Wähler öffnet zunächst den Offline-Authentifizierungscode. Der Wähler ruft den Service des Wählerverzeichnisses an, meldet die Diskrepanz zwischen erhaltenem und erwartetem Bestätigungscode und authentifiziert sich gegenüber dem Service mit Hilfe seines Offline-Authentifizierungscode. Das Wählerverzeichnis identifiziert die Wahltoken des entsprechenden Wählers und meldet dem Urnenserver, dass die zu der Wahltoken gehörende Stimme vom Urnenserver zu entfernen ist. Das Wählerverzeichnis schaltet die erneute Stimmabgabe des Wählers frei.

4 Bewertung der Cast-as-intended-Verifizierbarkeit in Polyas

Der dargelegte Vorschlag zur Umsetzung der *Cast-as-intended*-Verifizierbarkeit bietet gegenüber der Ansätze "Unabhängige Geräte" und "iterative Cut-and-choose" den Vorteil, dass weder unabhängige Geräte erforderlich sind, noch, dass eine umfangreiche Veränderung der Wahlprozesses aus Wählersicht notwendig ist. Die Umsetzung von Verifizierbarkeit wird in der Literatur zunehmend mit dem Augenmerk auf Verantwortlichkeit betrachtet, siehe zum Beispiel [18]. So kann die Glaubwürdigkeit eines Internetwahlsystems dadurch negativ beeinflusst werden, dass Wähler fälschlicherweise behaupten, der Verifizierungsprozess liefe nicht erfolgreich ab, oder gar behaupten, der erhaltene Bestätigungscode entspräche nicht dem erwarteten Bestätigungscode und folglich sei die Stimme falsch gewertet worden. Die Möglichkeit einer erneuten Stimmabgabe in Absprache mit dem Service des Wählerverzeichnisses beugt genau dieser Schwachstelle vor. So kann ein Wähler, der einen falschen Bestätigungscode erhalten hat oder vorgibt erhalten zu haben, seine Stimme jederzeit korrigieren.

Allerdings soll nicht unerwähnt bleiben, dass der entwickelte Ansatz den Nachteil mit sich bringt, dass Service-Mitarbeiter den Offline-Authentifizierungscode von Wählern lernen und folglich auch korrigierte Stimmabgaben entfernen könnten. Wir sehen mehrere Möglichkeiten, derartigen Angriffen vorzubeugen: Zunächst könnten Wähler mehrere Offline-Authentifizierungscodes erhalten, sodass bei jeder Korrektur ein neuer Code einzugeben ist. Das Wählerverzeichnis speichert alle Offline-Authentifizierungscodes, die von Service-Mitarbeitern angefragt wurden, sodass ein stichprobenartiger Abgleich zwischen angefragten Offline-Authentifizierungscodes und von Wählern geöffneten und verwendeten Offline-Authentifizierungscodes vorgenommen werden kann. Darüber hinaus könnte das Wählerverzeichnis nach erfolgreichem Entfernen einer Stimme den verwendeten Authentifizierungscode dem Wähler per Mail oder SMS zukommen lassen, sodass ein Wähler durch Offenlegen seiner Wahlmaterialien beweisen kann, dass ein derartiger Code niemals geöffnet wurde.

5 Zusammenfassung und Ausblick

Eines der bedeutendsten Internetwahlsysteme der Praxis ist das Polyas-System, wie die Vielzahl durchgeführter Wahlen belegt. Die (Weiter-)Entwicklung dieses Systems war stets an den Anforderungen wissenschaftlicher und industrieller Standards angelehnt. Die zunehmende Bedeutung der Verifizierbarkeit forderte eine nächste Iteration der technischen Weiterentwicklung. Bereits im Jahr 2011 konnte ein Vorschlag zur Umsetzung des dritten Aspektes der Verifizierbarkeit gemacht werden, der im Jahr 2012 umgesetzt wurde. Ziel war dabei, das Polyas-System um die Eigenschaft der *Cast-as-intended*-Verifizierbarkeit zu erweitern. Dazu wurde ein auf Bestätigungscodes basierender Vorschlag entwickelt.

Die aktuellen Polyas-Erweiterungen zur Verifizierbarkeit sind ein bedeutender Schritt. Dennoch ist zumindest ein weiterer Aspekt abzudecken, bevor die angestrebte Ende-zu-Ende-Verifizierbarkeit erreicht werden kann: So ist zukünftig sicherzustellen, dass Wähler überprüfen können, dass die von ihnen abgegebene Stimme tatsächlich korrekt in den zentralen Komponenten des Systems zur Auszählung zwischengespeichert wird. Darüber hinaus sieht die wissenschaftliche Literatur zunehmend die Notwendigkeit, Verifizierbarkeit um den Aspekt der verifizierbaren Wahlberechtigung zu erweitern. So soll es allen Wählern möglich sein, zu überprüfen, dass nur Wahlberichtigte ihre Stimme einmal abgeben, ohne dabei die Anonymität der Wähler zu verletzen.

Acknowledgement

Dieses Projekt (HA-Projekt-Nr.: 435/14-25) wird im Rahmen von Hessen ModellProjekte aus Mitteln der LOEWE – Landes-Offensive zur Entwicklung Wissenschaftlich-ökonomischer Exzellenz, Förderlinie 3: KMU-Verbundvorhaben gefördert.

Literatur

[1] Adida, B. *Helios: Web-based Open-Audit Voting*. In: USENIX Security Symposium 2008, Vol. 17, pp. 335-348.

[2] Clarkson, M.R.; Chong, S.; Myers, A.C. *Civitas: Toward a Secure Voting System*. In: IEEE Symposium on Security and Privacy 2008, IEEE 2008, pp. 354-368.

[3] Budurushi, J.; Neumann, S.; Olembo, M. M.; Volkamer, M. *Pretty Understandable Democracy-A Secure and Understandable Internet Voting Scheme*. In: Eighth International Conference on Availability, Reliability and Security (ARES) 2013. IEEE, pp. 198-207.

[4] Olembo, M. M.; Schmidt, P.; Volkamer, M. *Introducing verifiability in the polyas remote electronic voting system*. In: Sixth International Conference on Availability, Reliability and Security (ARES) 2011, IEEE 2011, pp. 127-134.

[5] Karayumak, F.; Olembo, M. M.; Kauer, M.; Volkamer, M. *Usability analysis of helios – an open source verifiable remote electronic voting system*. In: Proceedings of the 2011 USENIX Electronic Voting Technology Workshop/Workshop on Trustworthy Elections. USENIX 2011.

[6] Neumann, S.; Volkamer, M. *Civitas and the real world: Problems and solutions from a practical point of view*. In: Seventh International Conference on Availability, Reliability and Security, IEEE 2012, pp. 180-185.

[7] Louridas, P.; Tsoukalas, G.; Papadimitriou, K.; Tsanakas, P. *Zeus: Bringing Internet Voting to Greece*. In E-Democracy, Security, Privacy and Trust in a Digital World. Springer International Publishing, pp. 213-223.

[8] Heiberg, S.; Willemson, J. *Verifiable internet voting in Estonia*. In: 6th International Conference on Electronic Voting (EVOTE) 2014. IEEE 2014, pp. 1-8.

[9] Barrat, J.; Chevalier, M.; Goldsmith, B.; Jandura, D.; Turner, J.; Sharma, R. *Internet Voting and Individual Verifiability: The Norwegian Return Codes*. In: Electronic Voting 2012, pp. 35-45.

[10] Olembo, M. M.; Kahlert, A.; Neumann, S.; Volkamer, M. *Partial Verifiability in POLYAS for the GI Elections*. In: Electronic Voting 2012, pp. 95-109.

[11] Volkamer, M.; Vogt, R. *Basissatz von Sicherheitsanforderungen an Online-Wahlprodukte*. Common Criteria Schutzprofil BSI-CC-PP-0037, Version 1, 18.04.2008.

[12] Benaloh, J. *Simple verifiable elections*. In: *Proceedings of the USENIX Workshop on Electronic Voting Technology 2006*. USENIX Association 2006, pp. 5-5.

[13] Grimm, R.; Reinhard, K.; Winter, C.; Witte, J. *Erfahrungen mit Online-Wahlen für Vereinsgremien*. In: Datenschutz und Datensicherheit-DuD, 33(2), pp. 97-101.

[14] Volkamer, M.; Grimm, R. *Determine the resilience of evaluated internet voting systems*. In: First International Workshop on Requirements Engineering for e-Voting Systems (RE-VOTE) 2009, IEEE 2009, pp. 47-54.

[15] Ryan, P. Y.; Teague, V. *Pretty good democracy*. In: Security Protocols XVII. Springer Berlin Heidelberg 2013, pp. 111-130.

[16] Zagórski, F.; Carback, R. T.; Chaum, D.; Clark, J.; Essex, A.; Vora, P. L. *Remotegrity: Design and use of an end-to-end verifiable remote voting system*. In: Applied Cryptography and Network Security. Springer Berlin Heidelberg 2013, pp. 441-457.

[17] Neumann, S.; Olembo, M. M.; Renaud, K.; Volkamer, M. *Helios Verification: To Alleviate, or to Nominate: Is That the Question, or Shall we Have Both?* In: Electronic Government and the Information Systems Perspective. Springer International Publishing 2014, pp. 246-260.

[18] Truderung, T.; Vogt, A. *Accountability: definition and relationship to verifiability*. In: Proceedings of the 17th ACM conference on Computer and communications security. ACM 2010, pp. 526-535.