Eine prototypische Diskussion von M-Voting am Fallbeispiel der Wahl zum österreichischen Bundespräsidenten

Peter Leitner, Barbara Ondrisek, Thomas Grechenig

MOSO - Mobilizing Society, Forschungsgruppe INSO
Institut für Rechnergestützte Automation, Technische Universität Wien
Operngasse 9 / Hochparterre, 1040 Wien
{peter.leitner, barbara.ondrisek, thomas.grechenig}@inso.tuwien.ac.at

Schlagworte: **M-Voting, M-Government, E-Voting, E-Democracy, Mobile Application Development, Mobile Wahlkommission, Bürgerkarte, Digitale Signatur**

Abstract: Dieser Beitrag beschreibt die Entwicklung eines Prototypen für mobile Wahlen am Beispiel der Wahl zum österreichischen Bundespräsidenten. Mobile Wahlen auf digitalen mobilen Endgeräten bilden speziell bei mobilen Wahlkommissionen eine papierlose Alternative zu klassischen Wahlverfahren. Obwohl in Österreich bereits die rechtlichen Grundlagen für elektronische Wahlen geschaffen wurden, ist der Einsatz von E-Voting nicht unumstritten. Umfassende Sicherheitsvorkehrungen sind notwendig, um Manipulationen auszuschließen. Konzepte wie das der asymmetrischen Verschlüsselung, der digitalen blinden Signatur und ein erweiterter Wahl-Algorithmus helfen dabei, die nötige Sicherheit zu garantieren. Der beschriebene Prototyp, bestehend aus einem Wahl- und einem Urnenserver, sowie einem PDA-Client mit Kartenlesegerät, stellt ein dementsprechend sicherheitsoptimiertes System dar.

1. Einleitung

Der Einsatz moderner Informationstechnologie bietet nicht nur im Bereich der Wirtschaft nachhaltiges Optimierungspotenzial, sondern auch Bund, Länder und Kommunen können von einem solchen Schritt profitieren. Nicht ohne Grund prägte in der Vergangenheit das Schlagwort „E-Government“ verstärkt den öffentlichen Bereich.[1] Die Anbindung der behördlichen Instanzen an das Internet stellt jedoch

---

[1] Vgl *Blaschke, P./Karrlein, W./Zypries, B. (Hrsg),* E-Public: Strategien und Potenziale des E- und Mobile Business im öffentlichen Bereich (2002), Springer, Berlin ua, 59-124.

nur einen ersten Schritt in Richtung einer völligen Vernetzung dar.[2] Vor allem durch die schnelle Verbreitung von mobilen Endgeräten, wie Smartphones, PDAs, und Notebooks, aber auch durch die rasch ansteigende Datenübertragungsgeschwindigkeit, eröffnen sich viele neue Wege der elektronischen Interaktion. Die mobile Revolution schreitet unaufhaltsam voran und hält auch im öffentlichen Sektor verstärkt Einzug. E-Government wird zu M-Government. Der Bürger profitiert im Zuge dieser Entwicklung durch verkürzte Amtswege und eine nachhaltige Verbesserung der Interaktionsmöglichkeiten.[3]

Besonders die kompakten Geräteklassen, wie das Mobiltelefon oder der PDA, werden im Alltag immer stärker zur persönlichen Kommunikationszentrale und dienen neben der Kommunikation verstärkt zur Information (Datendienste, Nachrichtendienste usw) und Transaktion (M-Commerce, M-Payment, usw).[4] Auch im Bereich des M-Government sind es insbesondere diese handlichen Geräte, welche für zukunftsweisende Applikationen am besten geeignet sind.

Eine besonders innovative Möglichkeit der Anwendung von drahtloser Technologie stellt das M-Voting[5] dar, eine Sonderform der elektronischen Wahl, welche unter Verwendung mobiler Endgeräte durchgeführt wird. Bereits heute werden mobile Wahlen im klassischen Sinne durchgeführt, denn in Österreich haben kranke oder physisch beeinträchtigte Bürger durch den Besuch einer mobilen Wahlkommission die Möglichkeit, an Wahlen teilzunehmen.

Speziell für diese Benutzergruppe wurde ein Prototyp für M-Voting entwickelt, der direkt beim Bürger durch die mobile Wahlkommission eingesetzt wird. Wesentliche Motivation bestand darin, eine attraktive papierlose Alternative zum herkömmlichen Wahlverfahren zu schaffen, wobei auch klare Vorteile durch eine raschere Ergebniserfassung und bedeutend geringerem Administrationsaufwand im Rahmen der Wahlvorbereitung entstehen.

Die nachfolgenden Kapitel beschreiben die Prototypentwicklung am Beispiel der Wahl zum österreichischen Bundespräsidenten. Aufbauend auf den Konzeptentwurf wird der sicherheitsoptimierte Prozess der Stimmabgabe in seinen Teilschritten erläutert und die techni-

---

[2] Vgl *Mattern, F. (Hrsg),* Total vernetzt: Szenarien einer informatisierten Welt (2003), Springer, Berlin ua, 1-38.

[3] Vgl *Karlson, B. ua,* Wireless Foresight: Scenarios of the Mobile World in 2015 (2003), Wiley, Chichester ua, 67-84.

[4] Vgl *Zobel, J.,* Mobile Business und M-Commerce: Die Märkte der Zukunft erobern (2001), Hanser, Müchen, Wien, 183-198.

[5] Vgl *Ahonnen, T./Barret, J.,* Services for UMTS : Creating Killer Applications in 3G (2001), Wiley, Chichester, 203.

sche Umsetzung beschrieben. Den Abschluss bildet eine prototypische Diskussion der Entwicklungsergebnisse.

2. Konzeption des Prototypen

Die Systemarchitektur des M-Voting-Prototypen kann grundsätzlich in eine Serverseite und eine Clientseite unterteilt werden. Serverseitig werden ein Wahl- und ein Urnenserver betrieben und clientseitig kommt ein PDA mit Kartenlesegerät zum Einsatz. Der Wahlserver dient zur Überprüfung des Wählers und ist für die Registrierung und die Stimmzettelausgabe bestimmt. Der Urnenserver ist für die anonyme Stimmabgabe notwendig, wodurch die Geheimhaltung der Wähleridentität gewährleistet wird. Der Client wird im Zuge der Wahl von den Mitarbeitern der mobilen Wahlkommission eingesetzt, die sich vor Benutzung der Applikation authentifizieren müssen. Dem Wahlberechtigten wird der Client ausschließlich zur Stimmabgabe übergeben.

Bei der Umsetzung eines elektronischen Wahlsystems liegt die Herausforderung einerseits in der Einhaltung sämtlicher bereits bei der normalen Wahl vorgeschriebenen rechtlichen Bestimmungen, andererseits in der bestmöglichen Implementierung von zusätzlichen Sicherheitsrichtlinien, wie der asymmetrischen Verschlüsselung[6] und blinden Signaturen[7] (basierend auf asymmetrischer Verschlüsselung und Hash-Verfahren[8]), die den Ablauf der Stimmabgabe vor Angriffen und Missbrauch schützt.

Bei M-Voting-Systemen sind zusätzlich die Charakteristiken der mobilen Endgeräte zu berücksichtigen. Beispielsweise müssen beim „Mobile Application Development“ vor allem das fehlende technische Know-How der User, die eingeschränkten grafischen Möglichkeiten und die Modellvielfalt der Ausgabegeräte berücksichtigt werden. Der Aufbereitung der Benutzeroberfläche eines M-Voting-Systems muss deshalb besondere Aufmerksamkeit gewidmet werden.

Zusätzlich wurden beim vorliegenden Prototyp zur Vereinfachung beschränkende Annahmen getroffen. So wurde die Anzahl der Parteien auf fünf beschränkt und es ist keine Vergabe von Vorzugsstimmen möglich. Aus diesem Grund wurde auch die Wahl zum österreichi-

---

[6] Vgl *Lipp, P. ua*, Sicherheit und Kryptographie in Java - Einführung, Anwendung und Lösungen (2000), Addison-Wesley, München ua.

[7] Vgl *Chaum, D.,* Blind Signatures for Untraceable Payments. Advances in Cryptology Proceedings of Crypto (1982), Springer, Berlin, 199-203.

[8] Vgl *Pieprzyk, J., Sadeghiyan B.*, Design of Hashing Algorithms (1993), Springer, Berlin ua.

schen Bundespräsidenten als Beispiel herangezogen, da es sich dabei um eine Direktwahl handelt.

3. Ablauf der Stimmabgabe

Der Prozess der Stimmabgabe durch den Wähler ist in Abb 1 schematisch dargestellt und soll die komplexe Interaktion zwischen dem Client und den beiden Servereinheiten verdeutlichen.

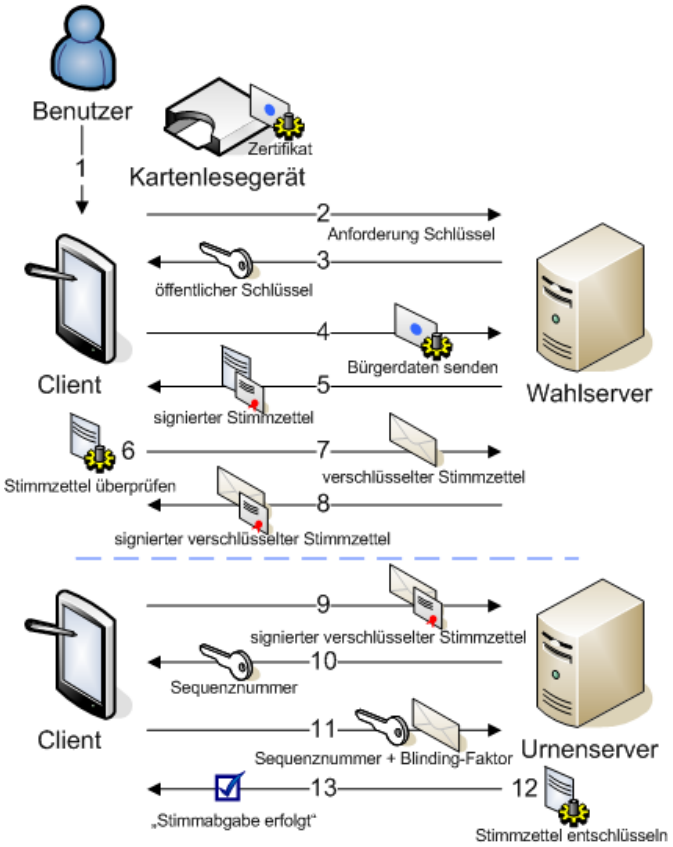

Abb 1: Technische Detailschritte der Stimmabgabe

Die Vielzahl der technischen Detailschritte bei der Stimmabgabe sind wesentliche Voraussetzung zur Sicherheitsoptimierung während

des Wahlvorganges bzw zur Einhaltung aller rechtlichen Vorgaben nach den Grundsätzen des Wahlrechts, wobei dem geheimen und dem persönlichen Wahlrecht besondere Bedeutung zukommen. Zur Erhöhung der Transparenz des Wahlvorganges, werden die einzelnen Schritte der Stimmabgabe im Anschluss ausführlich beschrieben:

Schritt 1-3: Zu Beginn erfolgt die Identifikation des Bürgers durch das Einlesen der Daten von dessen Bürgerkarte. Zu diesem Zweck wird das am PDA angebrachte Kartenlesegerät verwendet. Abb 2 zeigt die grafische Benutzeroberfläche des Clients beim Einlesevorgang. Im nächsten Schritt wird der öffentliche Schlüssel vom Wahlserver durch den Client angefordert. Dieser ist für die Echtheitsprüfung des elektronischen Stimmzettels erforderlich. Der angeforderte Schlüssel wird anschließend vom Wahlserver an den Client übermittelt. Alternativ könnte dieser vor dem Wahlvorgang auch direkt auf dem Client eingegeben werden.

Abb 2: Einlesevorgang

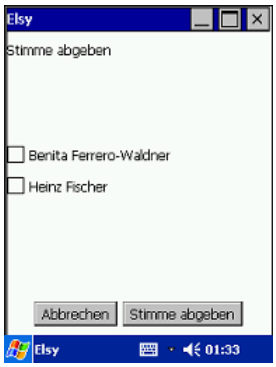

Abb 3: Stimmabgabe

Schritt 4-5: Die Daten des Bürgers von der Bürgerkarte werden an den Wahlserver geschickt, welcher den Status (etwa ob der Bürger registriert ist oder bereits gewählt hat) des Wählers überprüft. Nach erfolgter Verifikation retourniert der Wahlserver einen verschlüsselten und signierten Stimmzettel, der am Client überprüft, entschlüsselt und angezeigt wird, wie beispielhaft in Abb 3 dargestellt ist.

Schritt 6-8: Der Wähler kann nun seine Stimme auf dem Client (am PDA) abgeben, wobei natürlich auch die Möglichkeit bestehen muss, ungültig zu wählen. Vor dem tatsächlichen Absendevorgang der Stimme muss der Wähler nochmals seine getroffene Wahl bestätigen. Erst nach erfolgter Bestätigung wird der Stimmzettel mittels Blin-

ding-Faktor (einer Zufallszahl für das Chiffrieren) mathematisch verschlüsselt und zurück an den Wahlserver zur blinden Signierung geschickt. Der Wahlserver signiert nun den verschlüsselten und daher für ihn nicht lesbaren Stimmzettel und setzt den Status des Wählers auf „gewählt“. Abschließend schickt der Wahlserver den blind signierten Stimmzettel an den Client zurück.

Schritt 9-10: Der nun signierte, aber anonyme Stimmzettel wird ohne Angaben zum Bürger an den Urnenserver geschickt, der die Signatur des Wahlservers überprüft und den verschlüsselten und daher für ihn (noch) nicht lesbaren Stimmzettel temporär speichert. An den Client retourniert der Urnenserver eine Sequenznummer, welche den verschlüsselten Stimmzettel referenziert.

Schritt 11-13: Erhält der Client diese Referenznummer, so wird der Blinding-Faktor, mit dem der Stimmzettel verschlüsselt wurde, zusammen mit der Sequenznummer an den Urnenserver geschickt. Erst dadurch kann der Urnenserver den zwischengespeicherten Stimmzettel mit dem erhaltenen Blinding-Faktor entschlüsseln. Nach erfolgter Entschlüsselung wird eine Antwort an den Client gesendet, welcher dem Wähler, wie in Abb 4 dargestellt, eine Bestätigung über die erfolgreiche Stimmabgabe anzeigt.

Abb 4: Stimme abgegeben

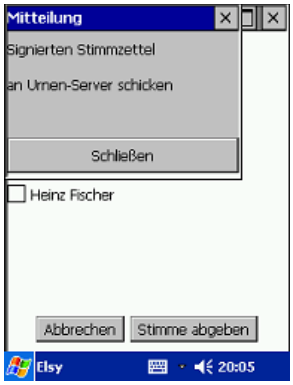

Abb 5: Message Box

Zusätzlich wird für jeden Zwischenschritt der Stimmabgabe, wie in Abb 5 ersichtlich, eine „Message Box“ auf dem Client angezeigt, welche den aktuellen Vorgang beschreibt. Dadurch soll dem Wähler maximale Transparenz während des gesamten Wahlvorganges geboten werden.

4. Technische Umsetzung

Bei der technischen Umsetzung des M-Voting-Systems wurde die gesamte Applikation mit der Programmiersprache Java[9] programmiert, um objektorientierte Programmkonzepte zu verwenden und eine plattformunabhängige Laufsicherheit zu garantieren.

Die Server-Applikation wurde mittels mehrerer Java-Servlets der Klasse *javax.servlet.http.HttpServlet* realisiert. Die Kommunikation zwischen dem Client und den beiden Servern wird jeweils vom Client initiiert, wobei jeweils ein Socket der Klasse *javax.net.ssl.SSLSocket* aufgebaut wird, welcher eine asymmetrisch verschlüsselte Verbindung über HTTPS zu den Servern herstellt. Die Verwendung von Sockets gewährleistet vor allem auch, dass der Urnenserver auf die anonymen Anfragen des Client über den Socket antwortet, ohne selbst eine Verbindung aufbauen zu müssen.

Die Client-Applikation wurde unter Verwendung der Java-Version JDK 1.2 implementiert, um die Kompatibilität mit *Jeode*, der Java Laufzeitumgebung auf dem PDA, sicherzustellen. Für die sichere Übertragung und Verschlüsselung wurden die Klassen des API von *Sun* verwendet, für die blinden Signaturen wurde das Open-Source API *logi.crypto* des isländischen Entwicklers *Ragnarsson*[10] gebraucht, da es Probleme bei der Implementierung der blinden Signatur mit den *Sun*-Klassen gab. Das GUI des Clients wurde vollständig mit *Sun*'s Java Klassen *java.awt* geschrieben, da die für GUI-Programmierung üblichen *javax.swing* Klassen in der am Client verwendeten Java Version nicht unterstützt werden.

Der für die Stimmabgabe des M-Voting-Prototypen verwendete Algorithmus basiert auf *Cranors* 1996 vorgestelltem *Sensus Protocol*,[11] welches eng an ein Verfahren von *Fujioka, Okamoto* und *Ohta*[12] angelehnt ist. *Sensus* verwendet das Verfahren der blinden Signatur, welches es ermöglicht ein Dokument zu signieren, ohne dessen Inhalt offen zu legen.

---

[9] Vgl *Flanagan, D.,* Java in a Nutshell: A Desktop Quick Reference, 4. ed (2002), O'Reilly, Sebastopol ua.

[10] API logi.crypto ist verfügbar unter http://www.logi.org/logi.crypto/.

[11] Vgl *Cranor, L./Cytron, R.,* Design and Implementation of a Practical Security-Conscious Electronic Polling System (1996), Computer Science Technical Report WUCS-96-02, Washington.

[12] Vgl *Fujioka, A./Okamoto, T./Ohta, K.,* A practical Secret Voting Scheme for Large Scale Elections (1993), Advances in Cryptology - AUSCRYPT '92 - Workshop on the Theory and Application of Cryptographic Techniques, 244-51.

5. Prototypische Diskussion

Das zu Beginn des Projektes gefasste Ziel, eine papierlose Alternative zum herkömmlichen mobilen Wahlverfahren zu entwickeln, wurde durch die Realisierung des beschriebenen Prototypen erreicht. Einige wenige Einschränkungen mussten im Laufe der Entwicklung aufgrund der eingesetzten Hardware vollzogen werden, was aber keinerlei Einfluss auf den Kernprozess der Stimmabgabe hatte. Beispielsweise musste der Einleseprozess der Bürgerkarte simuliert werden, da die mittlerweile verfügbaren Kartenlesegeräte zum Entwicklungszeitpunkt noch nicht einsatzbereit waren.

Eine Analyse der Benutzerfreundlichkeit (Usability) des Prototypen durch Testpersonen lieferte zufriedenstellende Ergebnisse hinsichtlich Darstellung der einzelnen Detailschritte (insbesondere bei der Stimmabgabe) und Bedienbarkeit durch den User. Zur Gewährleistung der Laufzeitsicherheit wurde ein Lasttest am System durchgeführt, welcher ebenfalls äußerst positive Ergebnisse lieferte.

Da beim derzeitigen Prototyp eine permanente Netzverbindung erforderlich ist, wäre als Erweiterung ein Offline-Betrieb denkbar, bei dem der Client ohne Verbindung arbeiten kann und die Ergebnisse nach Beendigung der Stimmabgaben gesammelt an den Urnenserver sendet. Dies würde die Flexibilität des Systems wesentlich erhöhen. Aus technologischer Sicht wäre dadurch aber auch ein größerer Speicher beim Client erforderlich, der bei derzeitigen mobilen Endgeräten zumeist noch begrenzt ist. Ebenfalls sinnvoll erscheint die Einführung einer erweiterbaren Stimmzettelanzeige für den Client.

6. Zusammenfassung und Ausblick

Durch die Entwicklung des Prototypen wurden die typischen Eigenschaften mobiler Endgeräte mit denen eines elektronischen Wahlsystems vereinigt. Die Konzentration lag dabei vor allem in der Einhaltung der rechtlichen und sicherheitstechnischen Vorgaben. Beim Test des M-Voting-Systems zeigte sich eine zufriedenstellende Performance und eine einfache und verständliche Bedienbarkeit durch den Benutzer.

Jedoch sind noch einzelne technische Herausforderungen zu bewältigen, ehe es zum Einsatz eines vollkommen ausgereiften und allgemein akzeptierten M-Voting-Systems in Österreich kommen kann. Neben der technisch notwendigen Optimierung sind vor allem die politischen Entscheidungsträger gefordert, zukunftsweisende Impulse in Richtung E- bzw M-Voting zu setzen.