Ausblick

Christoph Bieringer, Simon Schneider

27.10.2022

Verschiedene Schwachstellen, Probleme und Erweiterungsmöglichkeiten, die in dieser Arbeit nicht behandelt werden konnten, werden aufgelistet und beschrieben. Diese könnten Ausgangspunkte für zukünftige Arbeiten am System sein.

Inhalt

Fazit dieser Arbeit 3
Unterstützung anderer Wahlformen 3
Öffentliche Verifizierbarkeit der Auszählung 3
Bessere Sicherung des Administrator-Schlüssels 3
Verbesserung des Mix-Netzwerks 4
Verlagerung der Kryptographie hin zum Client 4
Untersuchung anderer theoretischer Ansätze 4

Fazit dieser Arbeit

Das aktuelle System weißt, wie in der Analyse gezeigt werden konnte, zahlreiche konzeptionelle, technische und methodische Mängel auf. Im Rahmen dieser Arbeit konnten allerdings einige davon bereits beseitigt werden. Zukünftige Arbeiten könnten das System noch weiter verbessern und damit eines Tages einen funktionierenden Prototypen hervorbringen. Eine (unvollständige) Reihe an solchen möglichen Verbesserungen bietet der Rest dieses Dokuments.

Unterstützung anderer Wahlformen

Eine mögliche Erweiterung für das System besteht darin, Unterstützung für verschiedene Wahlformen (und angepasste Frontend-Oberflächen) hinzuzufügen. Neben der aktuellen Auswahl aus einer Liste von Alternativen wären dies auch Ja/Nein-Fragen mit Checkboxen oder sog. Write-In-Ballots mit Texteingabefeldern.

Öffentliche Verifizierbarkeit der Auszählung

In der nach dem Ende dieser Arbeit übergebenen Version des Systems ist es den Benutzern zwar mittlerweile möglich, die Integrität der Blockchain zu verifizieren, die Auszählung der Stimmen können sie allerdings nicht selbst nachvollziehen. Dies wäre allerdings technisch möglich.

Hierzu müsste der Wahlserver nur noch die zum entschlüsseln benötigten Informationen (den privaten Administratorschlüssel sowie den Inhalt von directory_server3, d.h. die privaten Wählerschlüssel) öffentlich (z. Bsp. über weitere API-Endpoints) zur Verfügung stellen (erst nach Ende der Wahl, um den schon in der Analyse beschriebenen Angriff mittels der Auszählfunktion auszuschließen). Dann könnte das Frontend oder ein selbstgeschriebener Client mithilfe dieser Daten die in der Blockchain gespeicherten Stimmen selbst entschlüsseln und auszählen. Da in der Blockchain nur die persönlichen Hashwerte (die ja mit einem benutzerdefinierten PIN „gesalzen“ wurden) gespeichert sind, lassen sich von den so verfügbaren Informationen auch keine Rückschlüsse auf die Wahl einzelner Personen ziehen.

Auf diese Weise wäre ein weiterer wichtiger Schritt in Richtung einer vollständig verifizierbaren und damit vertrauenswürdigen Abstimmung gemacht. Vor der endgültigen Implementierung dieser Funktionalität sollte aber evtl. untersucht werden, ob dadurch andere, bisher noch nicht entdeckte Sicherheitslücken geöffnet würden.

Bessere Sicherung des Administrator-Schlüssels

Eine mögliche Erweiterung besteht darin, den Administrator-Schlüssel (bzw. dessen privaten Teil) besser zu schützen. Wie in der Analyse bereits angemerkt wurde, stellt die Verfügbarkeit des Administratorschlüssels ein Problem dar, da er von einem kompromittierten Wahlserver aus in falsche Hände gelangen könnte. Böswillige Akteure könnten dann mit ihm u.a. versuchen, Wähler zu deanonymisieren).

Um dieses Risiko zu vermeiden, könnte beispielsweise der private Teil des Administratorschlüssels mittels Secret-Sharing-Methoden auf mehrere Beteiligte verteilt werden (bspw. unterschiedliche Teile der Wahlbehörde oder unabhängige Organisationen). Dies würde einen Missbrauch stark erschweren, da nun mehrere Parteien zusammenarbeiten müssen, um den privaten Schlüssel zu berechnen. Für herkömmliche Secret-Sharing-Methoden wird allerdings zumindest für das Setup eine vertrauenswürdige Partei benötigt, die das Secret (in diesem Fall den privaten Administratorschlüssel) kennt. Das Problem, dass der Schlüssel in falsche Hände geraten könnte, wird also nur ein Stück verschoben und müsste nach wie vor anderweitig (Vertrauen, Sicherheitsprozeduren, Zugriffsbeschränkungen etc.) behoben werden. Mittels *Distributed Key*

Generation ist es aber auch möglich, ein Schlüsselpaar zu erzeugen, ohne dass eine Stelle den privaten Schlüssel vollständig kennt (dieser wird also überhaupt erst berechnet, wenn genug Teilnehmer dies möchten). Auch andere Methoden zur Sicherung des Administratorschlüssels sind denkbar.

Verbesserung des Mix-Netzwerks

Die aktuelle Version des Wahlservers implementiert nur einen Teil der für ein vollständiges Mix-Netzwerk benötigten Funktionalität. Insbesondere werden die Nachrichten zwar zufällig von einem Wahlserver zum nächsten geroutet, sodass der eigentliche Empfänger (das Register) nicht mehr direkt auf den Absender (den einzelnen Wähler) schließen kann, aber keine weiteren Verschleierungsmaßnahmen getroffen.

Ein Angreifer, der den Verkehr zwischen Wählern, Wahlservern und Registern zumindest teilweise mithören kann, kann also über Timing, Nachrichtenlänge etc. trotz der Verwendung des Mix-Netzwerks u.U. herausfinden, welche Nachricht von welchem Wähler stammt. Mit dieser Information könnte er in der Lage sein, einzelne Wähler zu deanonymisieren.

Aus diesem Grund implementieren reale Mix-Netzwerke neben dem Routing über mehrere Mix-Server hinweg noch weitere Features, etwa dass Verzögern und umordnen der empfangenen Nachrichten sowie das Weiterleiten von Nachrichten in größeren Batches. Solche Features würden auch das Voting-System deutlich verbessern und kommen damit ebenfalls für eine zukünftige Erweiterung in Frage.

Verlagerung der Kryptographie hin zum Client

Weiteres Verbesserungspotenzial bietet die Verlagerung von kryptographischen Berechnungen hin zum Client. In der aktuellen Architektur wird ein Großteil der sensiblen Informationen im Klartext zwischen Wahlserver/Register und Frontend ausgetauscht, obwohl dazu keine theoretische Notwendigkeit besteht. Aufgrund dieser Entscheidung muss der Benutzer Wahlserver und Register absolut vertrauen können, da diese seine Stimme im Klartext sehen.

Hier wäre evtl. zu prüfen, ob eine Möglichkeit besteht, gleiche oder ähnlich sichere kryptografische Funktionen wie in der aktuellen Implementierung auch auf dem Client auszuführen (bspw. mittels der Web Crypto API, die mittlerweile in allen gängigen Browsern unterstützt wird). Wenn dies der Fall ist, könnte die Ver- und Entschlüsselung der Stimme bei der Kommunikation mit Wahlserver und Register vom Client selbst durchgeführt werden. Die restlichen Komponenten des Systems würden dann nur noch die verschlüsselte Stimme sehen, sodass der Benutzer ihnen weniger Vertrauen entgegenbringen muss.

Eine solche Änderung würde große Teile des bisher geschriebenen Codes, sowohl im Frontend wie im Backend, betreffen und wäre vermutlich sehr aufwändig. Der Gewinn durch das zusätzliche Maß an Vertraulichkeit wäre aber beträchtlich und ein wichtiger Schritt hin zu einem ernstzunehmenden Prototypen.

Untersuchung anderer theoretischer Ansätze

Das aktuelle System stellt nur eine von vielen möglichen Architekturen für ein E-Voting-System dar. Ein Vergleich mit anderen Systemen aus Industrie und Forschung könnte dabei helfen, Stärken und Schwächen des gegenwärtigen Entwurfs besser zu verstehen. Auf diese Weise könnten auch mögliche Erweiterungen und Verbesserungen gefunden werden, die in dieser Arbeit noch nicht beschrieben wurden.

Mögliche Beispiele hierfür wären etwa die Verwendung von öffentlichen Blockchains, die Verwendung anderer kryptografischer Werkzeuge (etwa Zero-Knowledge-Verfahren oder homomorphe Verschlüsselung) für Teile des Wahlprozesses oder die Abgabe sog. Decoy-Ballots.