Code Voting mit prüfbaren Code Sheets

Jörg Helbach

Ruhr-Universität Bochum
Lehrstuhl für Netz- und Datensicherheit
44780 Bochum
joerg@helbach.info

Abstract: Eine wesentliche Voraussetzung für ein Code Voting Verfahren ist die Annahme, dass die den Wählern übermittelten Code Sheets korrekt sind. Dies wird in der Regel erreicht, indem vorausgesetzt wird, dass der Wahlvorstand, der die Code Sheets erstellt und an die Wahlberechtigten verteilt, vertrauenswürdig ist. Der Wähler hat allerdings zu keiner Zeit die Möglichkeit zu überprüfen, ob die Nummerncodes, die auf seinem Code Sheet aufgedruckt sind, nach der Zuordnung im Wahlserver auch zu einer Stimmabgabe für die korrespondierende Wahloption führen. Um diesen Mangel abzuschwächen, wird eine Möglichkeit aufgezeigt, wie ein Wähler sein Code Sheet auf Korrektheit überprüfen kann.

1 Einleitung

Eines der größten Probleme bei der Nutzung von Internetwahlsystemen ist, dass der Wahlclient, d.h. der PC[1], den der Wähler zur Übertragung seines Wählerwillens verwendet, nicht als sicher angenommen werden kann. Die beiden generellen Ursachen dafür sind zum einen, dass die Stimmen über ein unsicheres Netzwerk übertragen werden, zum anderen, dass zur Stimmabgabe der Computer oder allgemein ein elektronisches Gerät des Wählers verwendet wird, welches nicht a priori vertrauenswürdig ist. Da heutzutage Schadsoftware weit verbreitet ist, sind kryptographische Maßnahmen, z.B. die Verschlüsselung der Stimmen vor Versendung über das Internet, wirkungslos, da entsprechend eingesetzte Schadsoftware die Stimme bereits vor der Verschlüsselung auslesen oder verändern kann. Zusätzlich können sämtliche Ausgaben der Wahlsoftware durch diese Schadsoftware manipuliert und der Wähler somit in seiner Wahlentscheidung beeinflusst oder beeinträchtigt werden, indem beispielsweise die Reihenfolge der Kandidaten auf der Webseite in einer anderen Reihenfolge dargestellt wird, als sie an den Wahlserver übermittelt wird.

Schätzungen[2] zufolge sind 15% bis 25% der Rechner im Internet mit Viren, Würmern, Trojanischen Pferden oder anderer, schadhafter Software verseucht.

---

[1] In diesem Beitrag werden nur PCs als Wahlclient betrachtet. Code Voting ist generell aber auch mit anderen Arten von Wahlclients, wie z.B. Mobiltelefonen oder PDAs, anwendbar.

[2] Vinton Cerf schätzt, dass von insgesamt etwa 600 Millionen Rechnern im Internet etwa 100 bis 150 Millionen Rechner Teil eines BotNets sind (http://news.bbc.co.uk/1/hi/business/6298641.stm).

Ein Wähler kann daher niemals sicher sein, dass sein Wahlclient korrekt arbeitet. Für diese Problematik wurde bereits im Jahr 2002 von Ronald Rivest der Begriff Secure Platform Problem geprägt [Riv02]. Eine gute Möglichkeit das Secure Platform Problem einzugrenzen ist die Nutzung von Code Voting, für das David Chaum bereits in 2001 mit seinem Wahlsystem „Sure Vote“ [Cha01] die Grundlagen entwickelte.

2 Code Voting

Das Sure Vote Verfahren basiert auf der Trennung der Kommunikationskanäle zwischen Wähler und Wahlbehörde sowie Wahlclient und Wahlserver, indem die Wähler vor der Wahl ein „Code Sheet“ erhalten, auf dem für jede Wahloption ein Nummerncode (Voting-TAN) aufgedruckt ist. Statt einer Wahloption im Klartext übermittelt der Wähler nur diesen Code an den Wahlserver. Dadurch kann Schadsoftware, die ggf. auf dem Wahlclient installiert ist, zwar diesen Code unter Umständen auslesen, die Anonymität der Wahl aber nicht brechen.

Es wird zunächst angenommen, dass ein vertrauenswürdiger Wahlvorstand existiert, der jedem Wahlberechtigten ein gültiges Code Sheet zur Verfügung stellt und die verwendeten Wahlserver und Datenbanken zuverlässig, verfügbar und sicher sind. Damit Code Voting sinnvoll angewendet werden kann, müssen zwei weitere, leicht zu realisierende, Annahmen getroffen werden. Zum einen müssen alle Voting-TANs eindeutig und zufällig über alle Kandidaten und alle Code Sheets verteilt sein, zum anderen dürfen alle Code Sheets nicht auf elektronischem Weg zum Wähler übertragen werden.

Mit diesen Voraussetzungen kann ein Angreifer bei einer passiven Attacke die übermittelte Voting-TAN mitlesen. Da diese TAN aber zufällig gewählt ist und es keine, außer für den Wähler selbst erkennbare, Korrelation zwischen der übermittelten Zahl und dem favorisierten Kandidaten gibt, kann der Angreifer die Wahlentscheidung nur raten. Im Falle einer aktiven Attacke kann der Angreifer die übermittelte TAN nicht nur mitlesen, sondern auch ändern bzw. die Übertragung der Voting-TAN an den Wahlserver ganz verhindern. Um einen solchen Angriff durchzuführen muss ein Angreifer nur die übermittelte Voting-TAN zurückhalten oder ändern, so dass sie ungültig wird und somit vom Wahlserver abgewiesen wird. Code Voting ist daher an dieser Stelle anfällig gegen einen Denial of Service Angriff, der weder vom Wähler noch von einer sonstigen Instanz entdeckt werden kann, d.h. insbesondere der Wähler hat keinerlei Möglichkeiten nachzuvollziehen, ob seine Wahlentscheidung den Wahlserver erreicht hat oder nicht.

Eine sinnvolle Erweiterung von Code Voting ist daher die Einführung einer Bestätigungs-TAN [HS07]. Die Bestätigungs-TAN muss dabei ebenfalls eindeutig und zufällig über alle Kandidaten und jedes Code Sheet gewählt sein.

Sobald der Wahlserver eine Voting-TAN erhält, antwortet er mit der korrespondierenden Bestätigungs-TAN, d.h. wenn der Wähler nach Übermittlung der Voting-TAN die korrekte Bestätigungs-TAN erhält, dann kann er sicher sein, dass seine Stimme korrekt beim Wahlserver angekommen ist[3].

Ein grundsätzliches Problem von Code Voting Verfahren bleibt jedoch, dass weder der Sender noch der Empfänger einer TAN sicher sein können, dass eine Nachricht unverändert bzw. überhaupt angekommen ist. Diese Problematik kann auf das Zwei-Armeen-Problem [AEH75] [Gra78] zurückgeführt werden, welches die Probleme und Herausforderungen zweier Parteien beschreibt, die über einen unsicheren Kanal kommunizieren wollen. Es kann gezeigt werden, dass das Zwei-Armeen-Problem unlösbar ist. Aus diesem Grund wird als Lösungsannäherung oftmals ein 3-Wege-Handshake verwendet. Analog dazu wird dem Code Sheet eine weitere TAN, die Finalisierungs-TAN, hinzugefügt, mit der der Wähler, nach dem Erhalt der korrekten Bestätigungs-TAN, seine Stimmabgabe finalisiert. Eine Stimme wird nur dann im Wahlergebnis berücksichtigt, wenn die passende gültige Finalisierungs-TAN übermittelt wurde (vgl. Abbildung 1). Ein weiteres Problem von Code Voting ist der Stimmenhandel. Ein Angreifer ist in der Lage mehrere Code Sheets zu kaufen, so dass er mehrere Stimmen abgeben kann. Somit ist die Gleichheit einer Wahl nicht gewährleistet. Ein Lösungsansatz dazu ist die Kopplung der Stimmabgabe an eine Gruppensignatur.

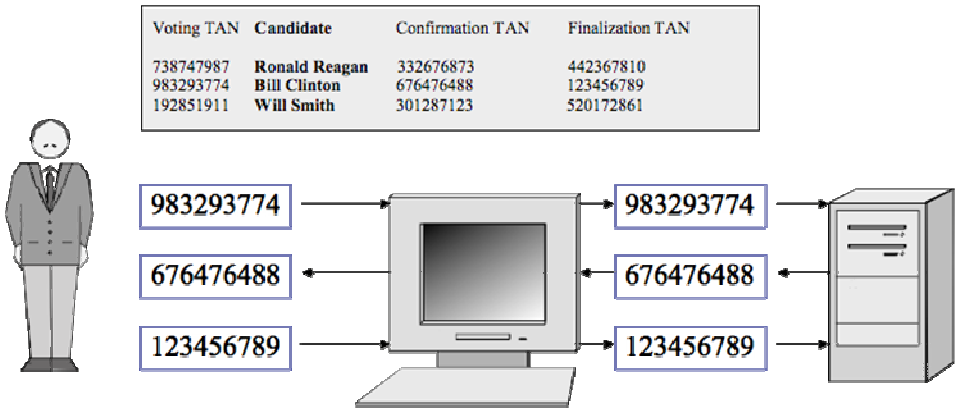

Abbildung 1: Kommunikation bei einem Code Voting Verfahren mit Finalisierung

[3] Zu bedenken sind an dieser Stelle noch Maßnahmen und Methoden, die ein Wähler ergreifen kann, falls er eine falsche oder gar keine Bestätigungs-TAN erhält. Diese werden hier jedoch nicht weiter betrachtet.

3 Code Voting mit Gruppensignaturen

1991 stellten Chaum und van Heyst das Prinzip der Gruppensignaturen [CH91] vor. Eine Gruppensignatur wird verwendet, um allen Mitgliedern einer Gruppe zu ermöglichen, eine Nachricht im Namen dieser Gruppe zu signieren. Die Signaturen sind dabei nicht zurückverfolgbar, d.h. es ist für ein beliebiges Gruppenmitglied oder auch eine Person außerhalb der Gruppe nicht möglich herauszufinden, welches Mitglied der Gruppe eine spezielle Nachricht signiert hat. Auch kann bei zwei gegebenen signierten Nachrichten nicht festgestellt werden, ob beide Nachrichten von demselben Gruppenmitglied signiert wurden. Es gibt allerdings einen Gruppenmanager, der die Mitgliederliste und die Signaturen der Gruppe verwaltet und somit feststellen kann, welches Gruppenmitglied eine Nachricht signiert hat, indem er die Gruppensignatur öffnet. Ein zu lösendes Problem besteht darin, dass der Gruppenmanager zur Erkennung einer mehrfachen Stimmabgabe jede Gruppensignatur öffnen muss, so dass die Anonymität der Wähler stark gefährdet ist.

Für ein elektronisches Wahlsystem ist es daher sinnvoll, diese grundsätzliche Funktionalität abzuschwächen, d.h. Gruppensignaturen rückverfolgbar zu machen, so dass ein Wähler nicht zwei Stimmen abgeben kann. Dazu veröffentlichten Canard, Schoenmakers, Stam und Traorè mit Listensignaturen eine spezielle Variante einer Gruppensignatur [CSS06], mit der die Anzahl der Signaturen, die ein Gruppenmitglied ausstellen kann, in einem bestimmten Zeitfenster beschränkt wird. Diese Beschränkung kann dabei nicht nur von einem Gruppenmanager, sondern auch von allen anderen Gruppenmitgliedern überprüft werden. D.h. sofern ein Mitglied mehr als die erlaubte Anzahl an Signaturen ausstellt, kann die Identität des Mitglieds von jedem anderen Mitglied ermittelt werden.

4 Prüfbare Code Sheets

Code Voting ist, wie in den vorigen Abschnitten dargestellt, eine gute Möglichkeit elektronische Wahlen über ein unsicheres Netzwerk anonym zu gestalten und gleichzeitig den Stimmenhandel durch einfache Weitergabe eines Code Sheets zu behindern, da ein potentieller Stimmenkäufer aufgrund der Gruppensignatur nicht in der Lage ist ein gekauftes Code Sheet auch zur Stimmabgabe einzusetzen. Das Verfahren setzt allerdings voraus, dass der Wahlvorstand, der die Code Sheets erstellt, vertrauenswürdig ist und die Code Sheets stets fehlerfrei sind. In diesem Abschnitt wird eine Möglichkeit aufgezeigt, die es einem Wähler ermöglicht sich von der Korrektheit des Code Sheets zu überzeugen.

4.1 Voraussetzungen

Es gelten nach wie vor die Voraussetzungen, dass alle Wahlserver und -datenbanken sicher und zuverlässig sind, dass alle TANs auf allen Code Sheets zufällig und gleichmäßig verteilt sind, und dass die Code Sheets nicht auf elektronischem Weg an die Wahlberechtigten verteilt werden dürfen. Die vierte Voraussetzung gilt nur eingeschränkt, d.h. grundsätzlich wird zwar angenommen, dass der Wahlvorstand vertrauenswürdig ist, allerdings muss nicht mehr davon ausgegangen werden, dass die Code Sheets alle korrekt und unverändert an die Wahlberechtigten verteilt wurden.

Weiter existiert ein Listensignaturschema, bei dem jeder Wahlberechtigte Mitglied der Gruppe ist und Nachrichten im Namen der Gruppe signieren kann. Eine mehrfache Stimmabgabe kann somit ohne das Öffnen der Gruppensignatur erkannt werden. Für die beteiligten Wahlclients gelten nach wie vor nur die Voraussetzungen, dass der Wahlclient einen Zugang zum Internet besitzt und ein Internetbrowser zur Verfügung steht. Ansonsten existieren keine weiteren Voraussetzungen, insbesondere können die Wahlclients mit Schadsoftware befallen sein.

4.2 Verfahren

Die Wahlbehörde erstellt Code Sheets, die auf der Vorder- und Rückseite jeweils mit einer Voting-TAN, einer Bestätigungs-TAN und einer Finalisierungs-TAN für jeden Kandidaten bedruckt sind. Jeder Kandidat kommt dabei genau einmal auf der Vorderseite und einmal auf der Rückseite vor. Diese so erstellten Code Sheets werden in anonyme, d.h. unbedruckte, Umschläge verpackt und verschlossen. Anschließend werden die Umschläge gemischt. Im nächsten Schritt wird für jeden Wahlberechtigten ein Adressetikett gedruckt. Jeweils ein Etikett wird auf einen der gemischten Umschläge aufgebracht. Die so präparierten Wahlbriefe werden versendet. Die versendeten Code Sheets sind dabei nicht personengebunden, d.h. zur weiteren, zusätzlichen Anonymisierung darf der Wähler sein Code Sheet mit anderen Wahlberechtigten tauschen.

Der Wahlberechtigte nimmt per Internetbrowser Kontakt zu den Wahlservern auf. Vor der Stimmabgabe hat der Wähler nun die Möglichkeit sein Code Sheet zu überprüfen. Dazu kann er eine beliebige Seite, also die Vorder- oder die Rückseite, seines Code Sheets auswählen und die aufgedruckten TANs überprüfen. Dazu existiert auf dem Wahlserver ein gesonderter Bereich innerhalb der Wahlapplikation. Der Wähler übermittelt die zu prüfende TAN an den Wahlserver, der darauf mit der Wahloption im Klartext antwortet. Sobald eine TAN überprüft wurde, sind alle TANs der entsprechenden Code Sheet Seite nicht mehr für die eigentliche Stimmabgabe zulässig, lassen sich aber ebenfalls überprüfen. Zudem kann keine TAN der anderen bedruckten Seite mehr überprüft werden, da die TANs dieser Seite nun zwingend zur Stimmabgabe vorgesehen sind. Sofern ein Code Sheet nicht überprüft wird, können beide Seiten zur Stimmabgabe verwendet werden.

Die eigentliche Stimmabgabe erfolgt analog zu den üblichen Code Voting Verfahren, indem der Wähler zunächst die entsprechenden TANs an den Wahlserver überträgt. Vor der Übermittlung einer TAN wird diese jeweils mittels der Listensignatur signiert.

Zunächst übermittelt der Wähler die entsprechende Voting-TAN, auf die der Wahlserver mit der korrespondierenden Bestätigungs-TAN antwortet. Wenn der Server die korrekte Bestätigungs-TAN übermittelt hat, finalisiert der Wähler seine Wahlentscheidung, indem er die signierte Finalisierungs-TAN an den Server überträgt. Solange die Wahlentscheidung nicht finalisiert wurde, kann der Wähler während der Wahlphase seine Wahlentscheidung beliebig oft via Vote Updating ändern.

Ein offenes Problem eines dreistufigen Code Voting Verfahrens ist, dass die Übermittlung der Wahlentscheidung per Denial of Service Angriff verhindert werden kann. Analog zu einem einstufigen Code Voting Verfahren genügt es einer Schadsoftware in diesem Fall, die Übermittlung der Finalisierungs-TAN an den Wahlserver zu unterbinden. Daher können beispielsweise in einer zusätzlichen Verifikationsphase nach Abschluss der eigentlichen Wahl die Bestätigungs-TANs aller derjenigen Voting-TANs veröffentlicht, die nicht finalisiert wurden. Die Wähler mit den entsprechenden Code Sheets haben dann noch die Möglichkeit die Stimmen zu finalisieren oder die Stimme per Vote Updating nochmals abzuändern.

Die Wähler, die in der eigentlichen Wahlphase keine Voting-TAN übermittelt haben oder deren Finalisierungs-TAN bereits korrekt an den Server übermittelt wurde, haben innerhalb dieser Phase nicht mehr die Möglichkeit ihre Stimmabgabe auszuführen bzw. zu verändern. Im Wahlergebnis werden nur die Voting-TANs berücksichtigt und gezählt, die mit einer Finalisierungs-TAN bestätigt wurden. Mithilfe der Listensignatur wird sichergestellt, dass jeder Wähler nur eine Stimme abgibt.

5 Fazit

Trotz einiger Angriffsszenarien sowie der grundsätzlichen Möglichkeit des Stimmenhandels ist Code Voting ein Internetwahlsystem, welches auch in der Praxis sinnvoll eingesetzt werden kann. Beispiele dafür sind Vereinswahlen oder Wahlen bei Aktiengesellschaften [Hel08].

Ein Nachteil dieser Verfahren ist, dass die Wähler darauf vertrauen müssen, dass die übermittelten Code Sheets korrekt sind. Mit dem in diesem Beitrag vorgestellten Verfahren kann jeder Wähler sein Code Sheet überprüfen. Für die Zukunft müssen Szenarien und Prozesse betrachtet werden, mit denen ein Wähler ein ggf. fehlerhaftes Code Sheet melden kann. Zu berücksichtigen ist dabei, wie unberechtigt als fehlerhaft gemeldete Code Sheets, bei denen die Prüfung z.B. durch Anwenderfehler fehlgeschlagen ist, erkannt werden können.

Literaturverzeichnis

[AEH75] Akkoyunlu, E. A. ; Ekanadham, K. ; Huber, R. V.: *Some constraints and tradeoffs in the design of network communications*. In: *SOSP '75:* Proceedings of the fifth ACM symposium on Operating systems principles. New York, ACM, S. 67–74, 1975.

[CH91] Chaum, David ; Heyst, Eugène van: *Group Signatures*. In: EUROCRYPT 1991, S. 257–265, 1991.

[Cha01] Chaum, David: *Sure Vote: Technical Overview*. In: Proceedings of the workshop on trustworthy elections (WOTE 01), presentation slides, 2001.

[CSS06] Canard, Schoenmakers, Stam, Traore: *List Signature Schemes*. In: Discrete Appl. Math., Volume 154, S. 189-201, 2006.

[Gra78] Gray, Jim: *Notes on Data Base Operating Systems*. In: Operating Systems, An Advanced Course. London, UK : Springer-Verlag, 1978.

[Hel08] Helbach, Jörg: *Code Voting - Ein Verfahren für Aktiengesellschaften?* In: Informatik 2008, Band 1, S. 417–422, 2008.

[HS07] Helbach, Jörg ; Schwenk, Jörg: *Secure Internet Voting With Code Sheets*. In: *E-Voting and Identity* Bd. Lecture Notes in Computer Science, LNCS 4896, S. 166–177, 2007.

[Riv02] Rivest, Ronald L.: *Electronic voting*. In: Financial Crypthography '01, S. 243-268, 2002.