Entwicklung eines formalen IT-Sicherheitsmodells für Online-Wahlsysteme

Rüdiger Grimm / Melanie Volkamer

Universität Koblenz-Landau
Universitätsstr. 1, D-56070 Koblenz
grimm@uni-koblenz.de
Universität Passau
Innstr. 43, D-94032 Passau
volkamer@uni-passau.de

Schlagworte: Online-Wahlsystem, Common Criteria, Schutzprofil, Formales IT-Sicherheitsmodell

Abstract: Online-Wahlen werden mehr und mehr eingesetzt – nicht zwingend für Parlamentswahlen, aber dennoch für Wahlen auf niedrigen Ebenen wie in Vereinen und an Universitäten. Um eine Basis für die Prüfung und Zertifizierung zu haben, wurde in Deutschland ein Common-Criteria-Schutzprofil, in dem Basisanforderungen für Online-Wahlprodukte definiert werden, entwickelt. Dieses Schutzprofil verlangt eine eher geringe Evaluierungstiefe (EAL2+). Für Wahlen auf höheren Ebenen ist eine entsprechende Anpassung der Evaluierungstiefe empfehlenswert. Dieser Artikel zeigt zunächst auf, dass eine Erhöhung derzeit nicht beliebig möglich ist, da ab der Stufe 6 formale Methoden und insbesondere ein formales IT-Sicherheitsmodell verlangt werden, ein solches Modell aber erst noch entwickelt werden muss. Im zweiten Schritt diskutiert dieser Artikel einen ersten Ansatz eines IT-Sicherheitsmodells für Online-Wahlsysteme, der aber nur eine Untermenge der im Schutzprofil definierten Sicherheitsziele (Security Objectives) berücksichtigt.

1. Einleitung

Die Gesellschaft für Informatik hat in den vergangenen zwei Jahren zusammen mit dem Bundesamt für Sicherheit in der Informationstechnik (BSI) und dem Deutschen Forschungszentrum für Künstliche Intelligenz das Schutzprofil für einen „Basissatz von Sicherheitsanforderungen an Online-

Wahlprodukte“ (Volkamer, Vogt 2008)[1] entwickelt. Das Schutzprofil basiert auf den Common Criteria (Common Criteria 2006). Die Stimmabgabe erfolgt bei den im Schutzprofil adressierten Systemen remote, über ein offenes Netzwerk, von einem beliebigen Endgerät, und die Stimmen werden auf einem Wahlserver gespeichert. Wie der Titel schon vermuten lässt, wird ein Minimum an Sicherheitszielen definiert, die ein Online-Wahlsystem erfüllen muss. Neben diesen Sicherheitszielen für das System wird auch eine Reihe von Annahmen an die Umgebung, in der das System eingesetzt wird, definiert. Eine geheime, freie, gleiche und allgemeine Wahl ist nur dann mit einem gegen dieses Schutzprofil zertifizierten Online-Wahlsystem möglich, wenn das System in einer Umgebung eingesetzt wird, in der die Annahmen erfüllt sind.

Die Common Criteria (CC) zusammen mit der Common Evaluation Methodology (Common Criteria 2006) definieren, wie die Einhaltung der Sicherheitsziele durch das System vom Evaluator zu überprüfen ist. Dabei unterscheiden die CC verschiedene Prüftiefen. Generell gilt: Je tiefer diese Prüfung geht, desto mehr steigt die Vertrauenswürdigkeit in das evaluierte System. Das Schutzprofil, das einen „Basissatz von Sicherheitsanforderungen an Online-Wahlprodukte“ definiert, fordert die Evaluierungsstufe (EAL) 2+ bei einer Skala von 1 bis 7+, wobei 7+ die intensivste Prüfung bedeutet. Generell steigen die Anforderungen an den zu prüfenden Umfang, an die Prüftiefe und an die Prüfmethoden mit steigender EAL-Nummer.

EAL 2+ zeichnet sich vor allem durch die folgende Aspekte aus:

- Durchführung von unabhängigen und strukturierten Tests beim Evaluator
- Analyse der Dokumentation bis hin zum Entwurf auf hoher Ebene und der Schnittstellenspezifikation
- Analyse der Stärke der Funktionen
- Suche nach offensichtlichen Schwachstellen durch den Evaluator
- Vorhandensein eines Konfigurationsverzeichnisses
- Nachweis einer sicheren Auslieferung.

EAL2+ ist für Wahlen im Umfeld von Vereinen, Schulen und Universitäten in vielen Fällen sicher ausreichend, allerdings nicht für Wahlen auf höheren Ebenen oder gar für parlamentarische Wahlen. So wurde beispielsweise bei der Entwicklung des Schutzprofil für den digitalen Wahlstift (Volkamer, Vogt 2006)[2], der in Hamburg zur Bürgerschaftswahl im Februar 2008 einge-

1 Für die Erstellung des Schutzprofils wurde CC Version 3.1 verwendet. Das Schutzprofil wurde im Mai 2008 durch das BSI zertifiziert.

2 Für die Erstellung des Schutzprofils wurde CC 2.3 verwendet.

setzt werden sollte, entschieden, dass hier EAL3+[3] gefordert wird. Einige Kritiker fordern sogar EAL4 und noch höher.

Generell wurden in der Vergangenheit vorwiegend Evaluierungen bis EAL4+ durchgeführt, da ab der Stufe EAL5 semi-formale bzw. formale Methoden verlangt werden und dies einen erheblichen Mehraufwand für Hersteller und Evaluator bedeutet. Die Entscheidung zu einer solchen Stufe sollte vor der Entwicklung getroffen werden, da (semi-)formale Methoden nicht im Nachgang implementiert werden können (bzw. der Aufwand hierzu genauso groß wie eine komplette Neuentwicklung ist). Dafür stellt EAL5 aber auch einen erheblichen Zuwachs an Vertrauenswürdigkeit im Vergleich zu EAL4 dar, da eine semi-formale Beschreibung (siehe ADV_FSP.5[4]) des Designs sowie eine stärker modularisierte und daher besser analysierbare Architektur gefordert wird. Ein entsprechender Zuwachs ist ab EAL6 von den semi-formalen zu den formalen Beschreibungssprachen zu verzeichnen. Insbesondere kann „durch eine formale Modellierung der Sicherheitspolitik – als formales Sicherheitsmodell – [kann] ein Zugewinn an Vertrauen in die Sicherheit des nach dieser Sicherheitspolitik arbeitenden Produktes erreicht werden [. . .]“ (Mantel, Stephan, Ullmann, Vogt 2002).

Ab EAL6 wird mit der CC-Komponente ADV_SPM.1 (Formal TOE security policy model) die Verwendung eines solchen formalen IT-Sicherheitsmodells (security policy model) gefordert. Darüber hinaus fordert diese Komponente einen Konsistenznachweis (in Form eines mathematischen Beweises) für das Modell und einen Compliance-Nachweis zwischen Systemspezifikation und definiertem Modell. Dabei ist es möglich, bereits veröffentlichte formale Sicherheitsmodelle[5] als Ganzes oder auch in Teilen zu verwenden. Wenn kein geeignetes IT-Sicherheitsmodell existiert, muss ein solches zunächst erstellt werden.

Letzteres trifft für Online-Wahlsysteme zu, daher müsste ein solches IT-Sicherheitsmodell erst einmal entwickelt werden, bevor eine Evaluierung nach EAL6 bzw. 7 angestrebt werden kann. Im Rahmen dieses Beitrags zeigen wir am Beispiel einiger konkreter Sicherheitsziele aus dem Schutzprofil auf, wie ein solches IT-Sicherheitsmodell konstruiert werden kann. Generell würde man ein solches Modell für eine konkrete Umsetzung des Schutzprofils entwickeln, da dies den Grad der möglichen Präzisierung

3 Das Schutzprofil verwendet eine Erweiterung von EAL 3 mit den Komponenten ADV_SPM.1 (Informal TOE security policy model), und AVA_MSU.3 (Analysis and testing for insecure states) ersetzt AVA_MSU.1.

4 FSP steht für Funktionale Spezifikation.

5 Beispiele für existierende IT-Sicherheitsmodelle sind das Bell/LaPadula-Modell, das Modell von Clark Wilson und das Biba-Modell.

erhöhen würde. Im weiteren Beitrag wird zunächst der Begriff des IT-Sicherheitsmodells als solches eingeführt und definiert (Kapitel 2), dann wird diskutiert, ob und wenn ja welche existierenden IT-Sicherheitsmodelle zum Einsatz kommen könnten (Kapitel 3). Anschließend werden Anforderungen aus dem Schutzprofil identifiziert, die als Kandidaten einer formalen Modellierung näher betrachtet werden (siehe Kapitel 4), und anschließend wird ein Modell entwickelt und gezeigt, dass es alle Eigenschaften eines IT-Sicherheitsmodells erfüllt (Kapitel 5). Der Beitrag schließt mit einem Ausblick auf zukünftige Forschungsaufgaben (Kapitel 6).

2. Methode der IT-Sicherheitsmodellierung

Nach (Grimm 2008) definieren IT-Sicherheitsmodelle Systemzustände und ihre Übergänge, unterscheiden sichere von unsicheren Zuständen und erklären, unter welchen Umständen sichere Zustände erreicht werden. Ein IT-Sicherheitsmodell kann mehr oder weniger formal sein. Alle IT-Sicherheitsmodelle enthalten die folgenden fünf Beschreibungselemente[6]:

1. die Definition eines übergeordneten Sicherheitsziels,
2. die Spezifikation sicherer Systemzustände[7], die als Ganzes das übergeordnete Sicherheitsziel darstellen,
3. ein Vertrauensmodell, das eine Menge von Anforderungen an die Umgebung definiert, in der das System eingesetzt werden muss, damit die Menge der spezifizierten Systemzustände äquivalent zu dem übergeordneten Sicherheitsziel ist,
4. ein Regelwerk für erlaubte Zustandsübergänge,
5. ein Sicherheitstheorem, in dem bewiesen wird, dass bei Einhaltung des Regelwerks das System von sicheren Zuständen notwendig in sichere Zustände überführt wird.

Ein IT-Sicherheitsmodell hat also zwei Lücken zu schließen, nämlich die Lücken

- zwischen den sicheren Systemzuständen und seinen Anwendungen, die sich über das übergeordnete Sicherheitsziel definieren (dies geschieht in 3)

6 Diese Methode ist i.W. von (Grimm 2008) übernommen und hier der Aufgabe von OnlinE-Voting angepasst.

7 Die Spezifikation sicherer Systemzustände entspricht im Rahmen der Common Criteria bei niedrigen Stufen den Sicherheitszielen (Security Objectives) für das System.

- und zwischen den erlaubten Zustandsübergängen und den sicheren Systemzuständen (dies geschieht in 5).

Die erste Lücke hat das Schutzprofil bereits durch die folgenden Abschnitte geschlossen:
- Definition der „Security Problem Definition“ in Form von Annahmen, Bedrohungen und organisatorischen Sicherheitspolitiken,
- abgeleitete Liste der Sicherheitsziele (Security Objectives) für das System und
- Diskussion im Abschnitt „Erklärung der Sicherheitsziele“.

Dieser Punkt wird daher hier nicht weiter behandelt. Die zweite Lücke schließt das Sicherheitstheorem mit seinem Nachweis, dass erlaubte Zustandsübergänge einen sicheren Systemzustand wieder in einen sicheren Systemzustand überführen.

Die Systemzustände (2. Beschreibungselement eines IT-Sicherheitsmodells) und die erlaubten Zustandsübergänge (4. Beschreibungselement) müssen möglichst genau beschrieben sein. Die sicheren Systemzustände können zunächst umgangssprachlich als spezielle Sicherheitsziele (im Sinne der Common Criteria, 2006) formuliert sein. Der Beweis des Theorems erfolgt durch eine sprachlich überzeugende und schlüssige Argumentation. Bei entsprechend hohen Anforderungen an die Vertrauenswürdigkeit, dass das Theorem wirklich gilt, müssen die sicheren Systemzustände und die erlaubten Zustandsübergänge sogar mathematisch formuliert und das zugehörige Sicherheitstheorem formal bewiesen werden. Als Ganzes erhält man dabei ein formales IT-Sicherheitsmodell[8].

Im Fall des formalen IT-Sicherheitsmodells ist eine dritte Lücke noch zu schließen, und zwar zwischen den sprachlich formulierten speziellen Sicherheitszielen und den formal definierten sicheren Systemzuständen. Dies ist nicht formalisierbar, sondern Gegenstand eines argumentativen Diskurses von Sicherheits- und Anwendungsexperten.

8 Die Common Criteria verstehen unter einem formalen IT-Sicherheitsmodell: „A formal security model is a precise formal presentation of the important aspects of security and their relationship to the behaviour of the TOE; it identifies the set of rules and practises that regulates how the TSF manages, protects, and otherwise controls the system resources. [...] the formal security policy model is merely a formal representation of the set of SFRs being claimed” (Common Criteria 2006).

3. Anwendung existierender IT-Sicherheitsmodelle für Wahlen

Es gibt unseres Wissens kein IT-Sicherheitsmodell, das das übergeordnete Ziel der sicheren elektronischen Wahl vollständig abdeckt. Angesichts der zahlreichen verschiedenen Aufgaben eines Wahlsystems halten wir das auch für unrealistisch. Das Integritätsmodell von Clark/Wilson (Clark, Wilson 1987) und das Vertraulichkeitsmodell von Bell/LaPadula (Bell, LaPadula 1973) können möglicherweise Teilziele beschreiben.

Das Clark-Wilson-Modell hat das Prinzip der Aufgabenteilung („Separation of Duty") in die Sicherheitsmodellierung eingeführt. Für verschiedene Handlungen im Rahmen einer elektronischen Wahl kann man die Aufgabenteilung im Sinne von Clark/Wilson sinnvoll einsetzen, um einige spezielle Sicherheitsziele zu formalisierten, zum Beispiel einige Aufgaben des Wahlvorstands. Das Schutzprofil „Basissatz von Sicherheitsanforderungen an Online-Wahlprodukte" (Volkamer, Vogt 2008) fordert etwa explizit als „Security Objective"[9]:

O.AuthWahlvorstand: Der EVG verfügt über eine Authentisierungsfunktion, die eine Separation of Duty auf mindestens zwei Mitglieder des Wahlvorstandes unterstützt. [...] Dadurch wird sichergestellt, dass sich immer mindestens zwei Mitglieder des Wahlvorstandes gegenseitig kontrollieren können.

Diesem Teilziel entspricht in etwa die Zertifizierungsregel C3 und die Durchsetzungsregeln E2 und E3, die im Clark-Wilson-Modell die „interne Konsistenz" eines Systems beschreiben:

- E2: Das System führt eine Zuordnungsliste von Nutzern auf Transaktionsprozeduren (User X, TPi, (CDIa,CDIb,CDIc, ...)) und sorgt dafür, dass Nutzer nur gemäß dieser Liste Transaktionsprozeduren ausführen dürfen.
- C3: Die Zuordnungsliste aus Regel E2 folgt dem Aufgabenteilungsprinzip.
- E3: Das System authentifiziert vor Ausführung einer Transaktionsprozedur die Identität des Nutzers.

Es ist eine reizvolle, aber bisher nicht gelöste Forschungsaufgabe, das Clark-Wilson-Modell auf das Wahlgeheimnis eines elektronischen Wahlsystems anzuwenden.

Das Bell-LaPadula-Modell verhindert vertrauliche Informationsflüsse in unzuständige Bereiche hinein. Das wird durch eine globale Zugriffskon-

9 Klauseln, die „Security Objectives" beschreiben, haben ein Überschriftskürzel, hier „Auth-Wahlvorstand", mit dem Buchstaben „O" als Präfix.

trolle („Mandatory Access Control“) durchgesetzt. Es ist denkbar, die Wähler, den Wahlvorstand, die Stimmzettel und die Urne in einem hierarchischen Informationsflussmodell à la Bell-LaPadula zu strukturieren und damit das Wahlgeheimnis bei der Stimmabgabe zu modellieren. Auch das ist eine bisher noch nicht gelöste Forschungsaufgabe. Es wäre reizvoll zu untersuchen, ob sich an dieser Stelle das Clark-Wilson-Modell und das Bell-LaPadula-Modell ergänzen könnten.

Wir wenden uns im Folgenden aber anderen speziellen Sicherheitszielen zu, die weder mit Bell-LaPadula und Clark-Wilson noch mit einem der anderen bekannten formalen IT-Sicherheitsmodelle sinnvoll zu modellieren sind. Daher entwerfen wir dafür ein eigenes formales IT-Sicherheitsmodell. Die sich daraus ergebenden Transaktionsprozeduren zur Durchsetzung dieses Sicherheitsziels könnten in ein übergeordnetes Aufgabenteilungsmodell nach Clark/Wilson eingebettet werden. Dies ist eine Herausforderung an zukünftige Forschungsarbeiten.

4. Auswahl an Sicherheitszielen aus dem PP

Die Entwicklung eines IT-Sicherheitsmodells ist eine komplexe Aufgabe und geschieht schrittweise, indem in jedem Schritt immer mehr der im Schutzprofil definierten Sicherheitsziele umgesetzt werden. Das Sicherheitsmodell, das wir in Kapitel 5 vorstellen werden, ist ein erster Ansatz, der für zwei ausgewählte Sicherheitsziele des Schutzprofils für einen „Basissatz von Sicherheitsanforderungen an Online-Wahlprodukte“ (Volkamer, Vogt 2008) vollständig durchgeführt wird und auf diese Weise prinzipiell aufzeigt, wie auch die weiteren Sicherheitsziele formal zu spezifizieren sind. Die beiden ausgewählten Sicherheitsziele sind:

O.UnbefugterWähler: Am EVG können nur Wähler mit Stimmberechtigung, die vom EVG eindeutig identifiziert und authentisiert werden, eine Stimme abgeben und damit einen Stimmdatensatz in der Urne speichern.

O.OneVoterOneVote: Der EVG stellt sicher, (A) dass jeder Wähler mit Stimmberechtigung nur eine Stimme abgeben kann und (B) dass er seine Stimmberechtigung nicht verliert ohne eine Stimme abgegeben zu haben. [...].

5. Formales IT-Sicherheitsmodell für Online-Wahlsysteme

Es existieren verschiedene Möglichkeiten ein bestimmtes System zu modellieren. Dies bezieht sich insbesondere auf den Abstraktionsgrad der Zustandsübergänge. In den folgenden fünf Abschnitten halten wir uns an die Beschreibungsstruktur für IT-Sicherheitsmodelle nach (Grimm 2008), um die beiden in Kapitel 4 ausgewählten Sicherheitsziele („Security Objectives") „O.Unbefugter Wähler" und „O.OneVoterOneVote" zu beschreiben.

5.1 Definition des übergeordneten Sicherheitsziels

Durchführung einer geheimen, gleichen, allgemeinen, freien und unmittelbaren Online-Wahl.

5.2 Spezifikation sicherer Systemzustände

Zur Spezifikation der sicheren Systemzustände sind zunächst die Systemzustände selbst zu definieren, bevor wir unter ihnen die sicheren Systemzustände auszeichnen.

Allgemeiner Systemzustand:

Ein Zustand wird in Form eines Tripels (*W,S,voter*) dargestellt. Die drei Komponenten haben folgende Bedeutung:

1. W – Menge der Wahlberechtigten, die noch eine Stimme abgeben dürfen: diese stehen im Wählerverzeichnis und haben noch keine Stimme abgegeben. Am Anfang enthält die Menge $W=W_{total}$ die Menge sämtlicher Wahlberechtigter.
2. S – Menge der (verschlüsselten) Stimmen in der elektronischen Urne.
3. *voter:* $S \rightarrow M$ – Abbildung von (verschlüsselten) Stimmen zum zugehörigen Wähler.

Dabei ist M eine Obermenge $M \supseteq W_{total}$ von Menschen, die auf das System zugreifen, egal, ob mit oder ohne Stimmberechtigung. Die Funktion *voter* ordnet jeder verschlüsselten Stimme ihren Erzeuger zu.

Bemerkung 1: Bei der Briefwahl wird die Funktion *voter* durch den äußeren Briefumschlag realisiert, die den Absender aufzeigt. Erst bei der Aus-

zählung wird die Absenderadresse auf dem äußeren Umschlag geprüft, d. h. es wird festgestellt, ob $voter(s) \in W_{total}$ oder $voter(s) \in M \backslash W_{total}$, und danach wird der äußere Briefumschlag entfernt.

Bemerkung 2: Das Abbild von *voter* ist nur für die zuletzt abgegebene Stimme, d. h. für $s \in S_i \backslash S_{i+1}$ feststellbar. Nachdem s anonymisiert wird, existiert der Link zwischen Wähler und seiner Stimme nicht mehr. Daher sollte die Zuordnung $voter_i$ praktisch nur während des Zugangsübergangs auf der „sichtbaren" Untermenge $S_i \backslash S_{i+1}$ von S_i genutzt werden, s. u. Abschnitt 4 über „Erlaubte Zustandsübergänge". Für den „unsichtbaren" Teil S_i definieren wir $voter_{i+1} \backslash S_i := voter_i$.

Sicherer Systemzustand:

Es ist zu definieren, welche Eigenschaften die einzelnen Beschreibungselemente eines Zustandes erfüllen müssen, damit der Zustand als sicher gilt. Wie in Kapitel 2 und 3 besprochen, dienen die PP-Sicherheitsziele („Security Objectives") dem übergeordneten Sicherheitsziel einer Online-Wahl. Wir formalisieren hier beispielhaft zwei der in (Volkamer, Vogt 2008) identifizierten Security Objectives. Wie in Kapitel 4 besprochen, werden die PP-Sicherheitsziele „O.UnbefugterWähler" und „O.OneVoterOneVote" als formale Systemzustände formuliert:

O.UnbefugterWähler:
$\forall s \in S: voter(s) \in Wtotal;$ das heißt, in der Urne sind nur Stimmen enthalten ($s \in S$), deren zugehörige Stimmabgeber $voter(s)$ im Wählerverzeichnis stehen. Dabei wird an dieser Stelle vorausgesetzt, dass das Wählerverzeichnis keine unbefugten Wähler enthält.

O.OneVoterOneVote (A): $\forall s, s' \in (S: voter(s) = voter(s') \Rightarrow s = s';$ das heißt, dass, wann immer in der Menge der Stimmen zwei Stimmen vom gleichen Wähler abgegeben wurden, diese beiden Stimmen identisch sind und damit die Stimme nur einmal im Ergebnis berücksichtigt wird. Dies wiederum bedeutet, dass jeder Wähler nur eine Stimme abgeben kann.

O.OneVoterOneVote (B): $\forall x \in W_{total} \backslash W: \exists s \in S: voter(s) = x;$ das heißt, ein Wähler kann nur dann auf „hat gewählt" gesetzt werden, wenn seine Stimme in der Urne ($s \in S$) gespeichert ist; das drückt aus, dass der Wähler sein Wahlrecht nicht verlieren kann, ohne dass seine Stimme in der Urne gespeichert ist.

Anfangszustand:

Als Anfangszustand wird definiert: $< W_0 = W_{total}, S_0 = \{\}, voter_0 = \{\}>$.

Dabei steht W_{total} für die Menge aller Wahlberechtigten im Wählerverzeichnis (solche, die noch eine Stimme abgegeben können, und solche, die bereits eine Stimme abgegeben haben). Die leeren Mengen S_0 und $voter_0$ repräsentieren die leere Urne zu Beginn der Wahl und die leere Abbildung von der leeren Urne auf die Benutzer des Systems. Der Anfangszustand ist offensichtlich sicher, da im Anfangszustand die All-Quantoren „$\forall$“ über die leere Menge zu bilden sind.

5.3 Vertrauensmodel

Die Menge der Anforderungen an die Umgebung und die zugehörige Argumentation befinden sich in (Volkamer, Vogt 2008).

5.4 Erlaubte Zustandsübergänge

Ein Zustandsübergang von Zustand $Z_i=<W_i,S_i,voter_i>$ nach $Z_{i+1}=<W_{i+1},S_{i+1},voter_{i+1}>$ ist zulässig, wenn eine der beiden folgenden Regeln eingehalten wird:

- Zustandsübergang, bei dem keine Stimme abgegeben wird:
  [Regel 1] $W_i = W_{i+1} \wedge S_i = S_{i+1} \wedge voter_i = voter_{i+1}$
- Zustandsübergang, bei dem eine Stimme erfolgreich abgegeben wird, wobei sich die Mengen S und W ändern:
  [Regel 2] $\exists\ s \in S_{i+1} : (voter_{i+1}(s) \in W_i \wedge W_{i+1} = W_i \setminus \{voter_{i+1}(s)\} \wedge S_i = S_{i+1} \setminus \{s\})$

Bemerkung 1: Alle $m \in M$ können einen Zustandsübergang hervorrufen, indem sie versuchen eine Stimme abzugeben. Bei sicherem Zustandsübergang ergibt das für nicht stimmberechtigte $m \in M$ eben $W_{i+1} = W_i$ und $S_{i+1}=S_i$.
Bemerkung 2: Die Zustandsübergangsregel wendet die Abbildung *voter* nur auf den sichtbaren Bereich an, d. h. auf $S_i \setminus S_{i+1}$. Dadurch wird die Zustandsübergangsregel nutzbar für den Einsatz in der Praxis.

5.5 Theorem

Alle Systemzustände, die vom Startzustand $Z_0 = <W_{total}, \{\}, \{\}>$ ausgehend nur über erlaubte Zustandsübergänge erreicht werden, sind sicher.

Der Beweis des Theorems wird als Induktionsbeweis geführt. Er erfordert zwar einige Schreibarbeit, besteht aber im Wesentlichen aus naheliegenden Schritten. Aus Platzgründen ist dieser hier nicht aufgeführt, siehe dazu (Grimm, Volkamer 2008).

6. Ausblick

In Deutschland ist gerade ein Schutzprofil für einen „Basissatz von Sicherheitsanforderungen an Online-Wahlprodukte“ (Volkamer, Vogt 2008), welches die Evaluierungsstufe EAL2+ fordert, fertig gestellt und durch das BSI zertifiziert worden. Die aktuellen Diskussionen rund um die Evaluierung von elektronischen Wahlsystemen im Allgemeinen zeigen, dass die Kritiker eine hohe EAL-Stufe fordern. Prinzipiell ist dagegen nichts einzuwenden, denn immerhin reden wir bei politischen Wahlen über das höchste Gut einer Demokratie und wenn man nicht hier formale Methoden verwendet, wo dann? Man steht aber bzgl. einer Evaluierung nach EAL6 oder EAL7 noch ganz am Anfang, nicht nur bzgl. Online-Wahlen, sondern ganz allgemein bzgl. aller Computeranwendungen. Es sind hier zunächst einmal wissenschaftliche Arbeiten erforderlich, die Spezifikationen von IT-Sicherheitsmodellen für Online-Wahlsysteme weiter untersuchen.

Dieser Beitrag zeigt einen ersten Ansatz auf, wie die einzelnen Sicherheitsziele, die im Rahmen des Schutzprofils für einen „Basissatz von Sicherheitsanforderungen an Online-Wahlprodukte“ (Volkamer, Vogt 2008) entwickelt wurden, in ein formales IT-Sicherheitsmodell einfließen können. Bis zur vollständigen Formalisierung aller Sicherheitsziele und ihrer Integration in ein geschlossenes IT-Sicherheitsmodell für Online-Wahlsysteme ist aber noch erhebliche Forschungsarbeit zu leisten.

7. Literatur

D. E. Bell and L. J. LaPadula:	Secure Computer Systems: Mathematical Foundations, and A mathematical model. ESD-TR-73–278, MTR-2547, Vols 1&2. The MITRE Corporation, Bedford, MA, Nov. 1973.
D. Clark and D. Wilson:	A Comparison of Commercial and Military Security Policies. Proceedings of the 1987 IEEE Symposium on Security and Privacy, Oakland, CA. Computer Society Press of the IEEE, Washington DC, 184–194, 1987.
Common Criteria:	Common Criteria for Information Technology Security Evaluation, Version 3.1, 2006.
Common Criteria:	Common Methodology for Information Technology Security Evaluation, Version 3.1, 2006.
R. Grimm:	IT-Sicherheitsmodelle. Arbeitsberichte aus dem FB Informatik der Universität Koblenz-Landau, Feb. 2008, http://www.uni-koblenz.de/FB4/Publications/Reports. Sowie in: wisu 5/08: Das Wirtschaftsstudium, Lange Verlag, Düsseldorf, Mai 2008, S. 720–727.
R. Grimm and M. Volkamer:	Development of a Formal IT-Security Model for Remote Electronic Voting Systems. EVote08, Bregenz, 6.-9. August 2008.
H. Mantel, W. Stephan, M. Ullmann, and R. Vogt:	Leitfaden für die Erstellung und Prüfung formaler Sicherheitsmodelle im Rahmen von ITSEC und Common Criteria. Version 1.0c http://david.von-oheimb.de/cs/teach/BSI-Leitfaden_1.0c.pdf, 2002.
M. Volkamer and R. Vogt:	Digitales Wahlstift-System. Common Criteria Schutzprofil BSI-PP-0031, http://www.bsi.de/zertifiz/zert/reporte/PP0031b.pdf, 2006.
M. Volkamer and R. Vogt:	Basissatz von Sicherheitsanforderungen an Online-Wahlprodukte. Common Criteria Schutzprofil BSI-CC-PP-0037, Version 1.0, 18. April 2008. BSI-Zertifikat erteilt im Mai 2008.