Bern University
of Applied Sciences

Verifizierbare Internetwahlen:
Technische Lösungen und Grenzen

Rolf Haenni

6. November 2015

Im Zentrum der Sicherheitsanforderungen steht die Verifizierbarkeit.

Bericht des Bundesrates zu Vote électronique
Schweizerischer Bundesrat, 2013

Inhalt

- Einführung
- Die Mathematik der Verifizierbarkeit
- Individuelle Verifizierung
- Universelle Verifizierung
- Zusammenfassung

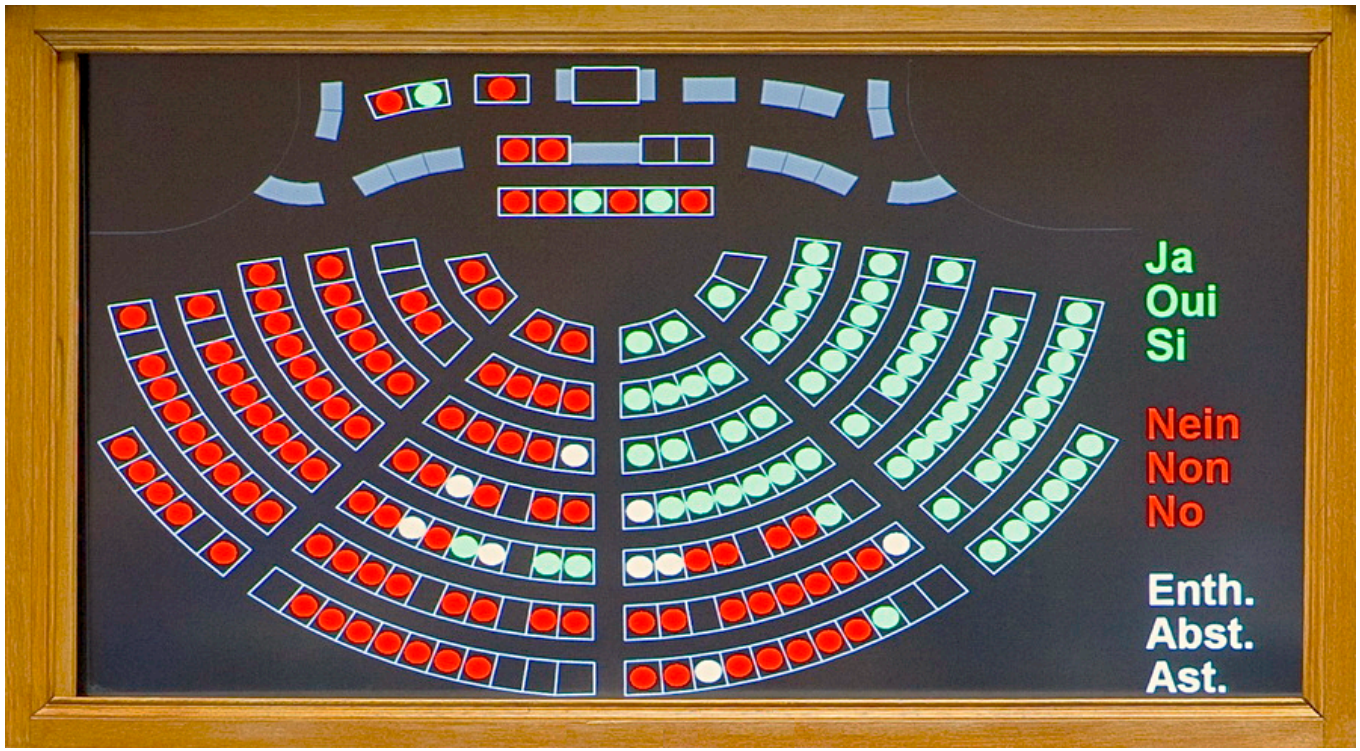
Einführung

Die Leute, die die Stimmen abgeben, entscheiden nichts. Die Leute, die die Stimmen zählen, entscheiden alles.

Josef Stalin

One should verify the election, not the election system.

Ben Adida


Ja
Oui
Si
Nein
Non
No
Enth.
Abst.
Ast.

Bern University
of Applied Sciences

Die Mathematik der Verifizierbarkeit

Potenzieren

$$
\begin{aligned}
z &= b^x \\
&= \underbrace{b * b * \cdots * b}_{x \text{ mal}}
\end{aligned}
$$

Beispiel: $z = 2^4 = 16$

Modulares Potenzieren

$$
\begin{aligned}
z &= b^x \bmod n \\
&= \underbrace{b * b * \cdots * b}_{x \text{ mal}} \bmod n
\end{aligned}
$$

Beispiel: $z = 2^4 \bmod 7 = 16 \bmod 7 = 2$

Mod. Potenzieren mit grossen Zahlen

Falls n und x sehr gross sind ($>$300 Ziffern), zum Beispiel

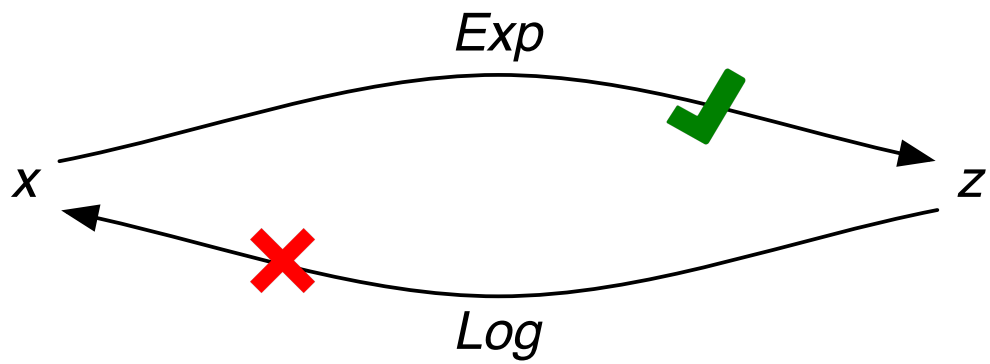
$$
\begin{aligned}
n = {} & 161931481198080639220214033595931441094586304918744740281 \\
& 350651054723722378777547542599144392497741933066317043332 \\
& 245697880199001800501144684304139086873298712511087431280 \\
& 878786588515668012772798298511621634145464600626613553954 \\
& 882323818539003486835493305012811566266345538418426995, \\
x = {} & 129854309843509097594837982374987436123824739827423874387 \\
& 783090109873283928709701274983270971298460129739827019273 \\
& 329857439098023983748273498723098098273074982374098230479 \\
& 439580349580349850039028938309282390981212378328473298478 \\
& 849593048509834095832092386483769203849870971231238231,
\end{aligned}
$$

dann ...

Eigenschaft 1: Einweg-Funktion

. . . ist die Berechnung von

- $z = Exp(x)$ leicht
- $x = Log(z)$ schwierig (unmöglich?)



Eigenschaft 2: Homomorphismus

$$Exp(x) * Exp(y) = Exp(x + y)$$

Eigenschaft 2: Homomorphismus

$$Exp(x) * Exp(y) = Exp(x + y)$$

$$Exp(x)^y = Exp(x \cdot y)$$

Eigenschaft 2: Homomorphismus

$$Encrypt(x) * Encrypt(y) = Encrypt(x + y)$$

$$Encrypt(x)^y = Encrypt(x \cdot y)$$

Kryptographische Beweise

- Wie kann ich beweisen, dass ich einen Wert x kenne, so dass $z = Exp(x)$ gilt?

Kryptographische Beweise

- Wie kann ich beweisen, dass ich einen Wert x kenne, so dass $z = Exp(x)$ gilt?

- Variante 1: den Wert x offenlegen

Kryptographische Beweise

- Wie kann ich beweisen, dass ich einen Wert x kenne, so dass $z = Exp(x)$ gilt?

- Variante 1: den Wert x offenlegen

- Variante 2: Erstellen eines kryptographischen Beweises
  - Wähle r zufällig
  - Berechne $t = Exp(r)$
  - Berechne $c = hash(t, z)$
  - Berechne $s = r + x \cdot c$
  - Veröffentlichung von (t, c, s)

  Der Beweis ist gültig, falls $Exp(s) = t \cdot z^c$ und $c = hash(t, z)$

Anwendungen beim E-Voting

- Verschlüsselung der Stimme
- Beweisen dass die verschlüsselte Stimme 0 oder 1 ist
- Zusammenzählen von verschlüsselten Stimmen (ohne sie zu entschlüsseln)
- Berechnen von Prüfcodes ohne die Stimmen zu entschlüsseln
- Wiederverschlüsselung von verschlüsselten Stimmen
- Kryptographisches Mischen von verschlüsselten Stimmen
- Beweisen dass korrekt gemischt wurde
- Teilen des privaten Schlüssels für die Entschlüsselung
- Entschlüsselung mittels privaten Teilschlüsseln
- etc.

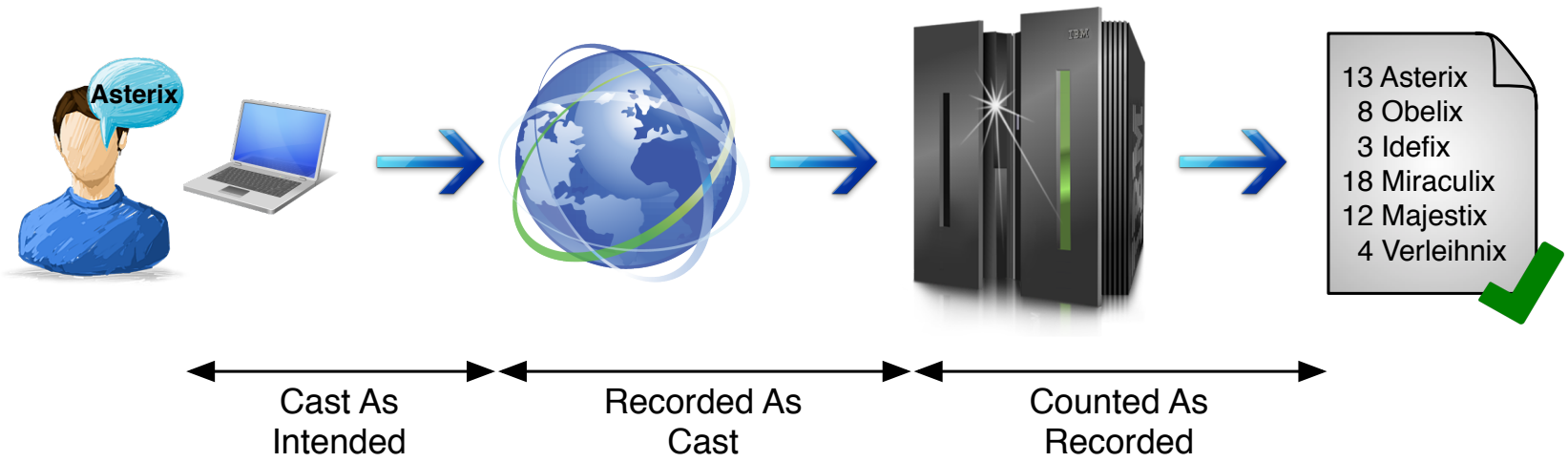
Bern University
of Applied Sciences

Individuelle Verifizierung

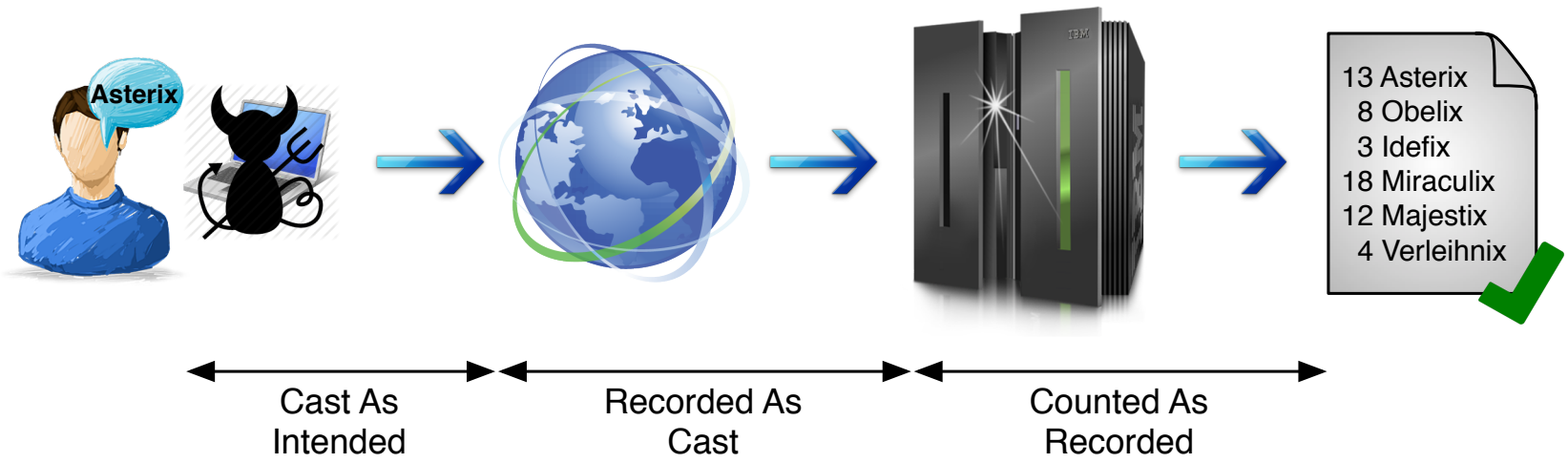
Die Stimmenden müssen die Möglichkeit haben, zu erkennen, ob ihre Stimme auf der Benutzerplattform oder auf dem Übertragungsweg manipuliert worden ist.

Verordnung der BK über die elektronische Stimmabgabe, VEleS, 2013

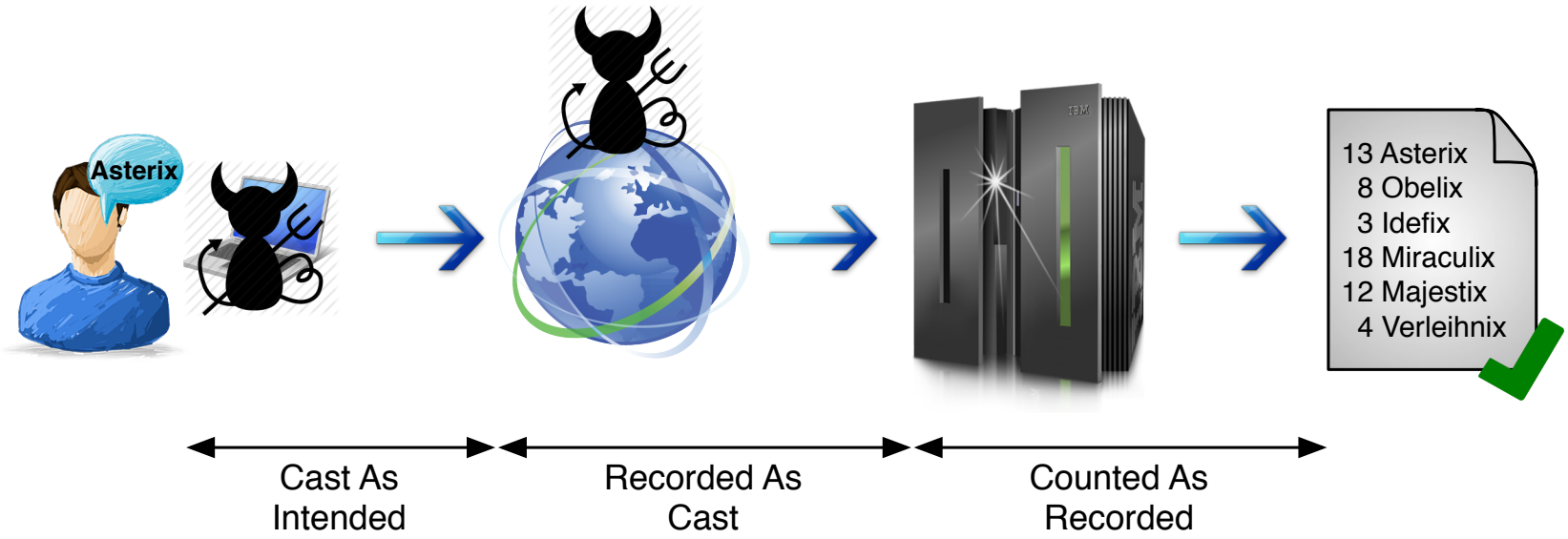
Angriffs- und Vertrauensmodell



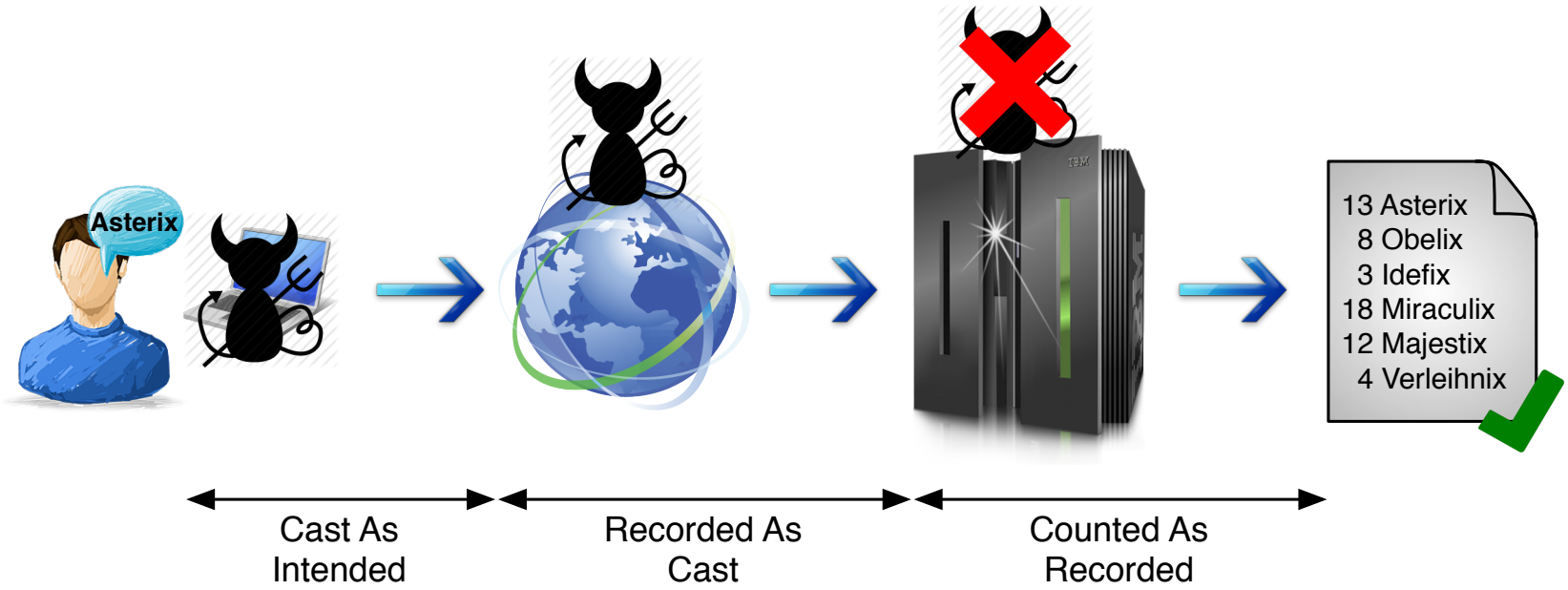
Angriffs- und Vertrauensmodell



Angriffs- und Vertrauensmodell



Angriffs- und Vertrauensmodell



Verifizierung mittels Code-Tabellen

- Vor einer Wahl erhalten die Stimmenden per Post eine Code-Tabelle mit unterschiedlichen Prüfcodes pro Wahloption
- Jede Tabelle enthält andere Prüfcodes

Code-Tabelle Nr.291

Kandidaten	Codes
Asterix	74494
Obelix	84443
Idefix	91123
Miraculix	63382
Majestix	85921
Verleihnix	79174

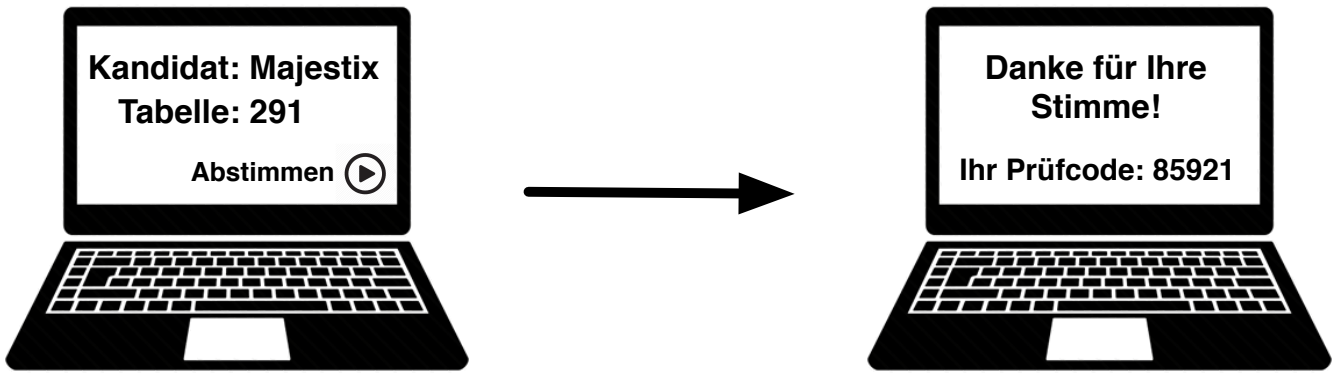
Code-Tabelle Nr.321

Kandidaten	Codes
Asterix	21344
Obelix	29173
Idefix	91123
Miraculix	72282
Majestix	18194
Verleihnix	53382

- Beispiele aus der Praxis: Norwegen (2011–2013), Neuchâtel (seit 2015), Genf (seit 2015)

Verifizierung mittels Code-Tabellen

- Nach dem Abschicken einer Stimme werden entsprechende Prüfcodes angezeigt



- Übereinstimmende Prüfcodes garantieren:
  - Cast-as-Intended
  - Recorded-as-Cast
- Ansonsten erfolgt eine Stimmabgabe auf Papier

Verifizierung mittels Code-Tabellen

- Kritische Prozesse
  - Erzeugen und Drucken der Code-Tabellen
  - Verschicken der Code-Tabellen (Post)
- Abgewehrte Malware-Angriffe
  - Blockieren der Stimmabgabe ✓
  - Falsche Stimmabgabe ✓
  - Blockieren der Prüfcodes ✓
  - Anzeigen falscher Prüfcodes ✓

Verifizierung mittels Code-Tabellen

- Kritische Prozesse
  - Erzeugen und Drucken der Code-Tabellen
  - Verschicken der Code-Tabellen (Post)
- Abgewehrte Malware-Angriffe
  - Blockieren der Stimmabgabe ✓
  - Falsche Stimmabgabe ✓
  - Blockieren der Prüfcodes ✓
  - Anzeigen falscher Prüfcodes ✓
- Nicht abgewehrte Malware-Angriffe
  - Stimmgeheimnis ☒
  - Social Engineering Attack: “Bitte Prüfcode eingeben” ☒

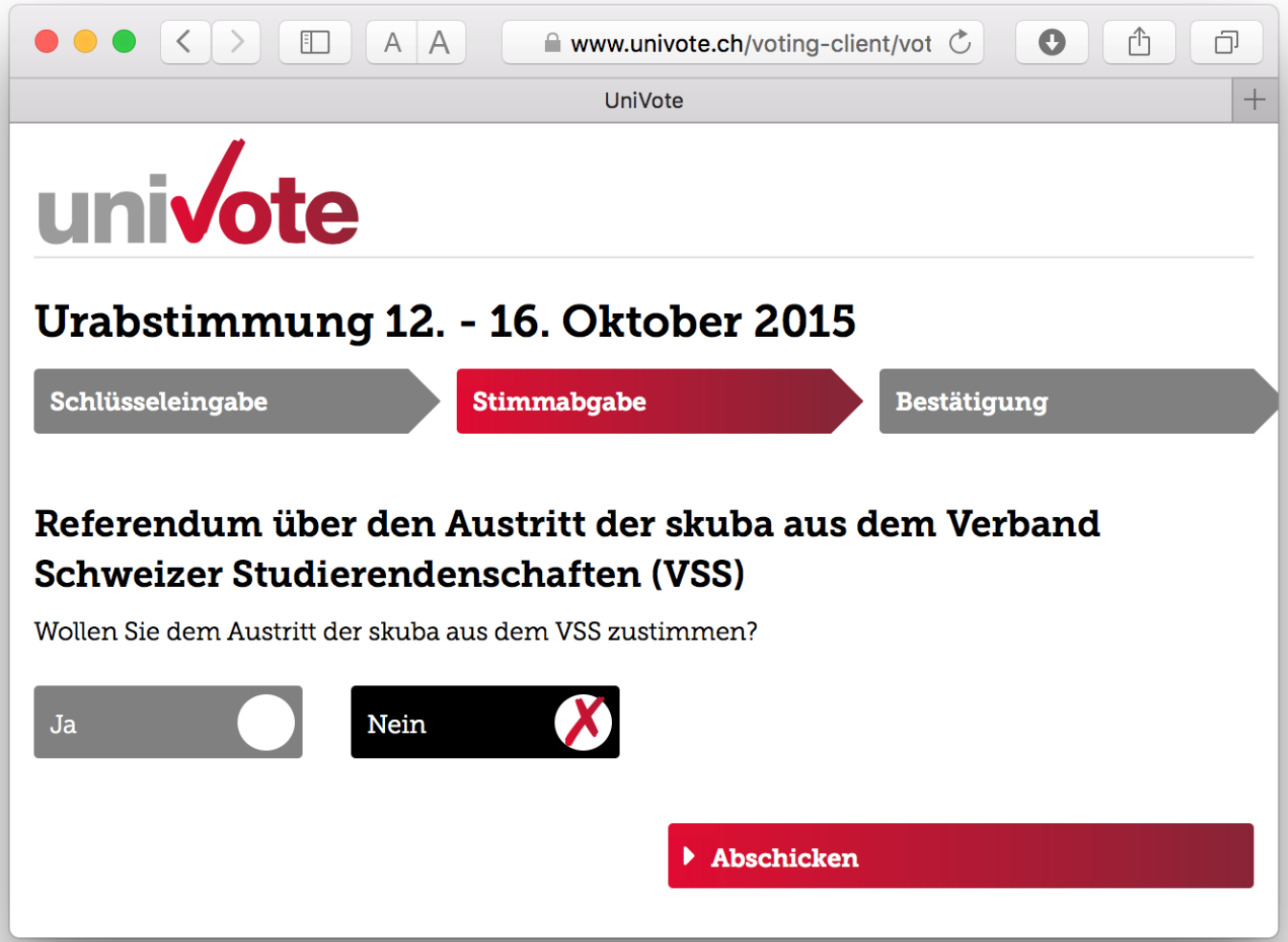
Verifizierung mittels Code-Tabellen

- Kritische Prozesse
  - Erzeugen und Drucken der Code-Tabellen
  - Verschicken der Code-Tabellen (Post)
- Abgewehrte Malware-Angriffe
  - Blockieren der Stimmabgabe ✓
  - Falsche Stimmabgabe ✓
  - Blockieren der Prüfcodes ✓
  - Anzeigen falscher Prüfcodes ✓
- Nicht abgewehrte Malware-Angriffe
  - Stimmgeheimnis ☒
  - Social Engineering Attack: “Bitte Prüfcode eingeben” ☒
- Nachteil: nicht medienbruchfrei

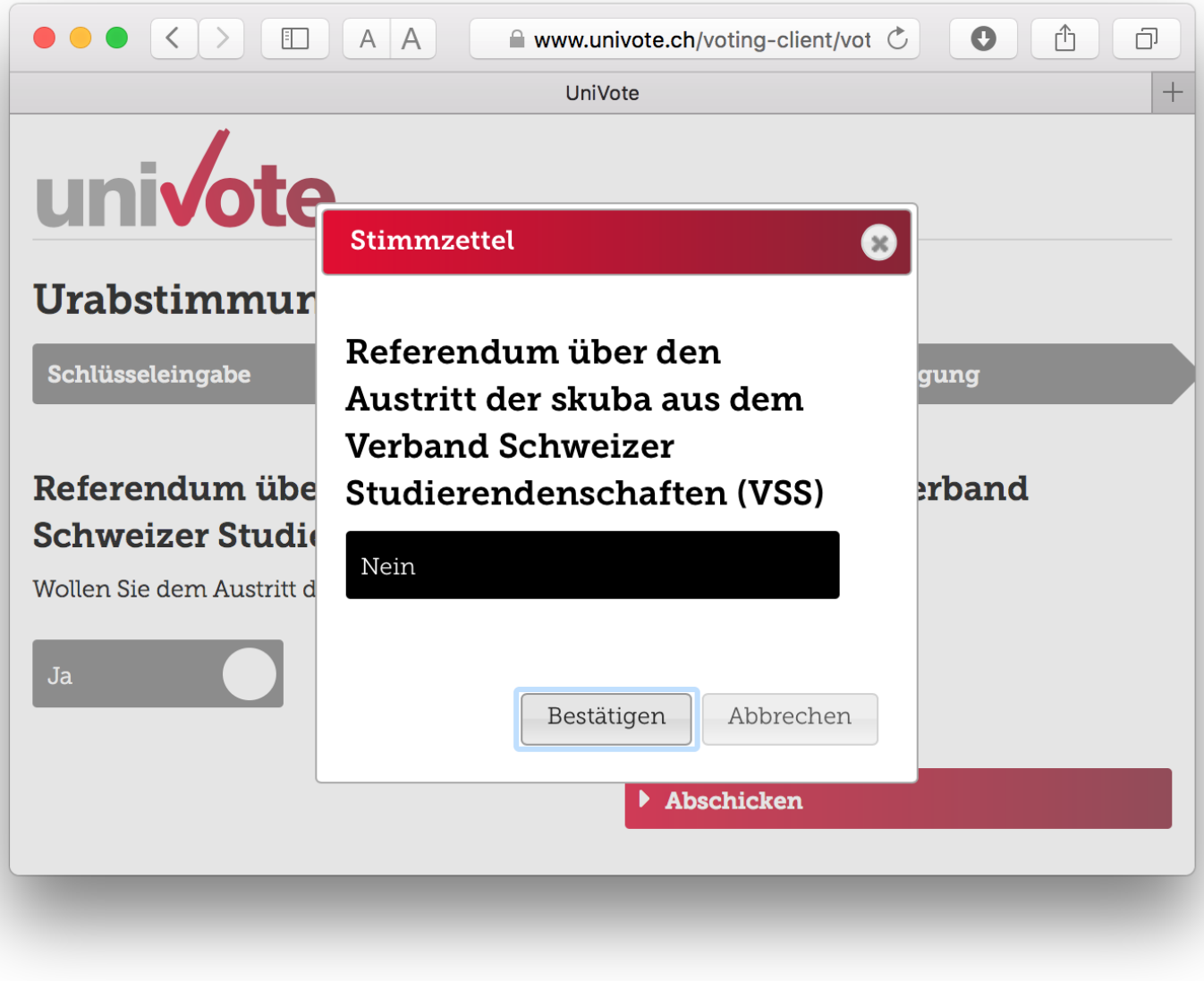
Verifizierung auf separatem Gerät

- Nach einer Wahl wird ein QR-Code mit Informationen über die abgegebene Stimme angezeigt
- Mit einem separaten Gerät (z.B. Smartphone) wird ...
  - der QR-Code eingescannt
  - die abgegebene verschlüsselte Stimme vom System heruntergeladen
  - die verschlüsselte Stimme entschlüsselt
  - die unverschlüsselte Stimme angezeigt
- Beispiele aus der Praxis: Estland (seit 2013), UniVote (BFH, seit 2013)

Verifizierung mit separatem Gerät



Verifizierung mit separatem Gerät



Verifizierung mit separatem Gerät

Verifizierung mit separatem Gerät

- Wichtige Grundvoraussetzung: Unabhängigkeit der Geräte
- Abgewehrte Malware-Angriffe
  - Blockieren der Stimmabgabe ✓
  - Falsche Stimmabgabe ✓

Verifizierung mit separatem Gerät

- Wichtige Grundvoraussetzung: Unabhängigkeit der Geräte
- Abgewehrte Malware-Angriffe
  - Blockieren der Stimmabgabe ✓
  - Falsche Stimmabgabe ✓
- Nicht abgewehrte Malware-Angriffe
  - Blockieren des QR-Codes ☒
  - Anzeigen eines falschen QR-Codes ☒
  - Stimmgeheimnis ☒
  - Koordinierter Angriff auf beide Geräte ☒

Verifizierung mit separatem Gerät

- Wichtige Grundvoraussetzung: Unabhängigkeit der Geräte
- Abgewehrte Malware-Angriffe
  - Blockieren der Stimmabgabe ✓
  - Falsche Stimmabgabe ✓
- Nicht abgewehrte Malware-Angriffe
  - Blockieren des QR-Codes ☒
  - Anzeigen eines falschen QR-Codes ☒
  - Stimmgeheimnis ☒
  - Koordinierter Angriff auf beide Geräte ☒
- Vorteil: medienbruchfrei

Verifizierung mit separatem Gerät

D. Springall, T. Finkenauer, Z. Durumeric, J. Kitcat, H. Hursti, M. MacAlpine, A. Halderman.
Security Analysis of the Estonian Internet Voting System. *CCS'14, Conference on Computer and Communications Security*, Scottsdale, USA, 2014.

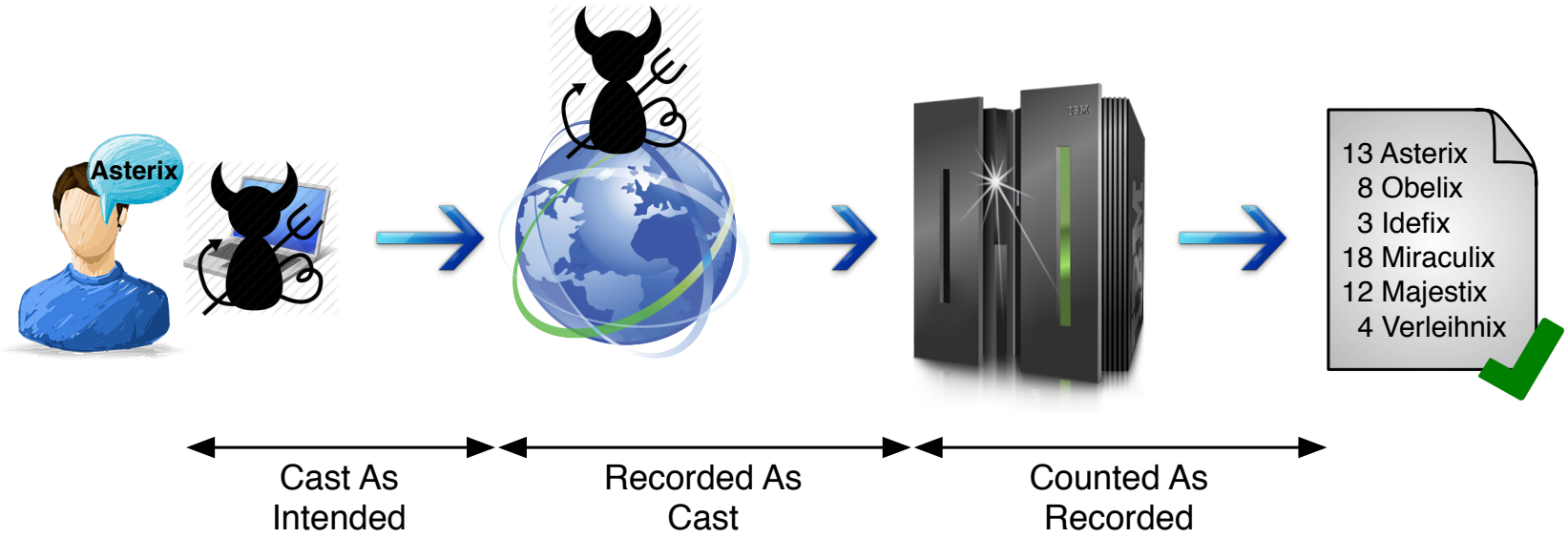
R. E. Koenig, R. Haenni, P. Locher.
Attacking the verification code mechanism in the Norwegian internet voting system. *VoteID'13, International Conference on E-Voting and Identity*, Guildford, U.K., 2013.

Universelle Verifizierung

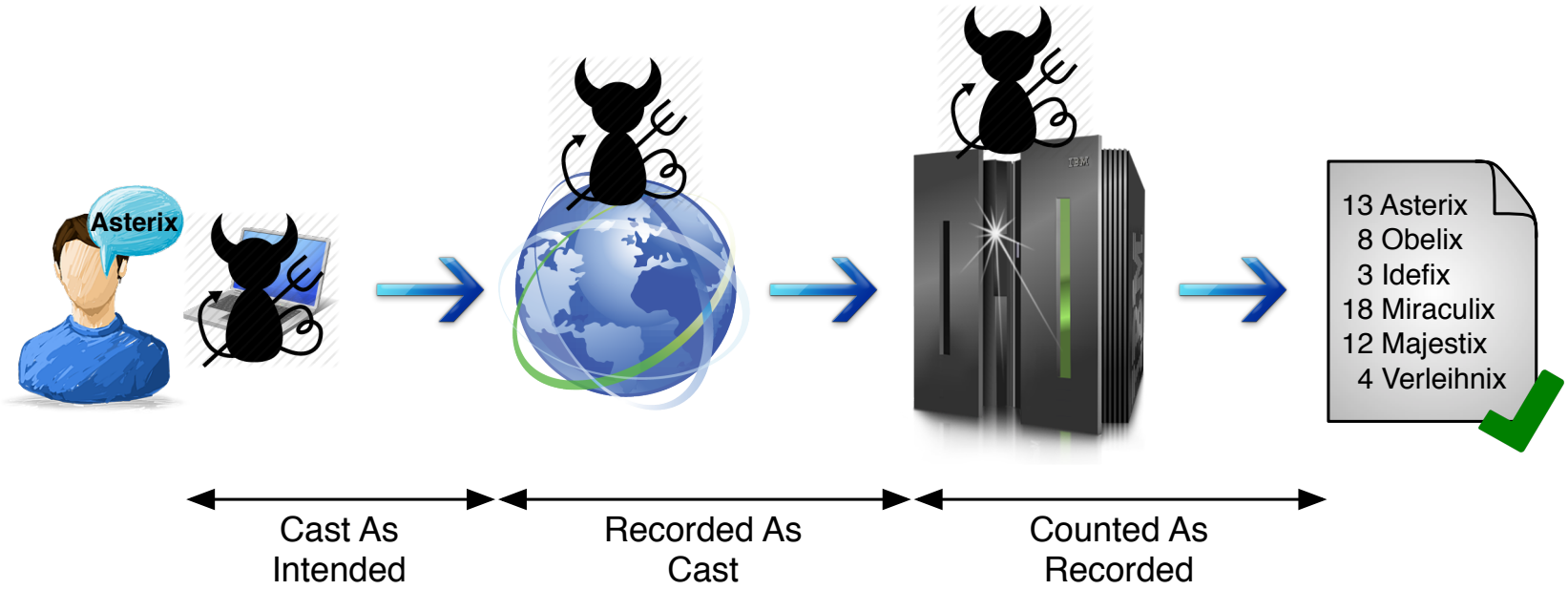
Zur universellen Verifizierung erhalten die Prüferinnen und Prüfer einen Beweis der korrekten Ergebnisermittlung. [...] Dazu müssen sie technische Hilfsmittel verwenden, die vom Rest des Systems unabhängig und isoliert sind.

Verordnung der BK über die elektronische Stimmabgabe, VEleS, 2013

Angriffs- und Vertrauensmodell



Angriffs- und Vertrauensmodell



Elektronisches Anschlagbrett

- Die Wahldaten werden auf ein öffentlich zugängliches elektronisches Anschlagbrett geschrieben
  - System-Setup
  - Kryptographische Parameter
  - Definition der Wahl (Titel, Periode, etc.)
  - Liste der Wahloptionen und Regeln
  - Elektorat mit Zertifikaten
  - Verschlüsselte Stimmen (mit kryptographischen Beweisen)
  - Gemischte verschlüsselte Stimmen (mit kryptographischen Beweisen)
  - Teilentschlüsselungen (mit kryptographischen Beweisen)
  - Entschlüsselte Stimmen
  - Wahlresultat
- Grundsatz: alle Daten sind öffentlich, ausser private Schlüssel

Elektronisches Anschlagbrett

Candidates	Voters	Encrypted Votes	Signatures	Mixed Votes	Decrypted Votes	Result
Asterix	Bob	E(Asterix)	SigBob	E'(Idefix)	Idefix	13 Asterix
Obelix	Jim	E(Idefix)	SigJim	E'(Asterix)	Asterix	8 Obelix
Idefix	Joe	E(Majestix)	SigJoe	E'(Asterix)	Asterix	3 Idefix
Miraculix	Tom	E(Asterix)	SigTom	E'(Majestix)	Asterix	18 Miraculix
Majestix	:	:	:	:	:	12 Majestix
Verleihnix	:	:	:	:	:	4 Verleihnix
				Shuffle Proof	Decryption Proof	

Wahl-Verifizierung

- Wahl-Verifizierung mittels unabhängiger Software
  - Basierend auf System-Spezifikation
  - Aussenstehende Entwickler (z.B. Hochschulen, NGOs, CCC, Privatpersonen, etc.)
  - Disjunkte Code-Basis
  - Open-Source

Wahl-Verifizierung

- Wahl-Verifizierung mittels unabhängiger Software
  - Basierend auf System-Spezifikation
  - Aussenstehende Entwickler (z.B. Hochschulen, NGOs, CCC, Privatpersonen, etc.)
  - Disjunkte Code-Basis
  - Open-Source
- Voraussetzung: System-Spezifikation von hoher Qualität
  - Vollständigkeit
  - Hoher Detaillierungsgrad
  - Öffentlich
  - Test-Daten

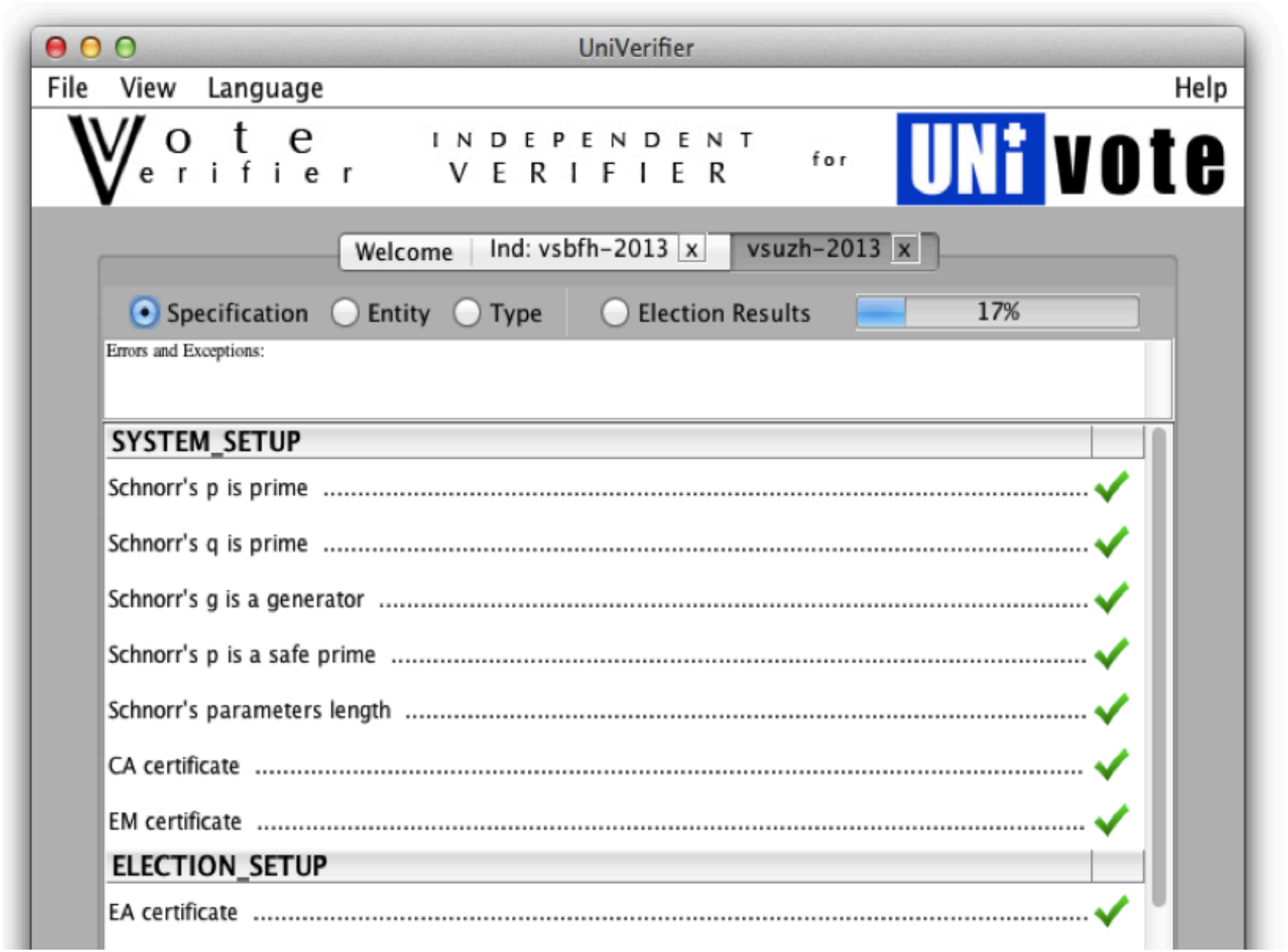
System-Spezifikation: Univote

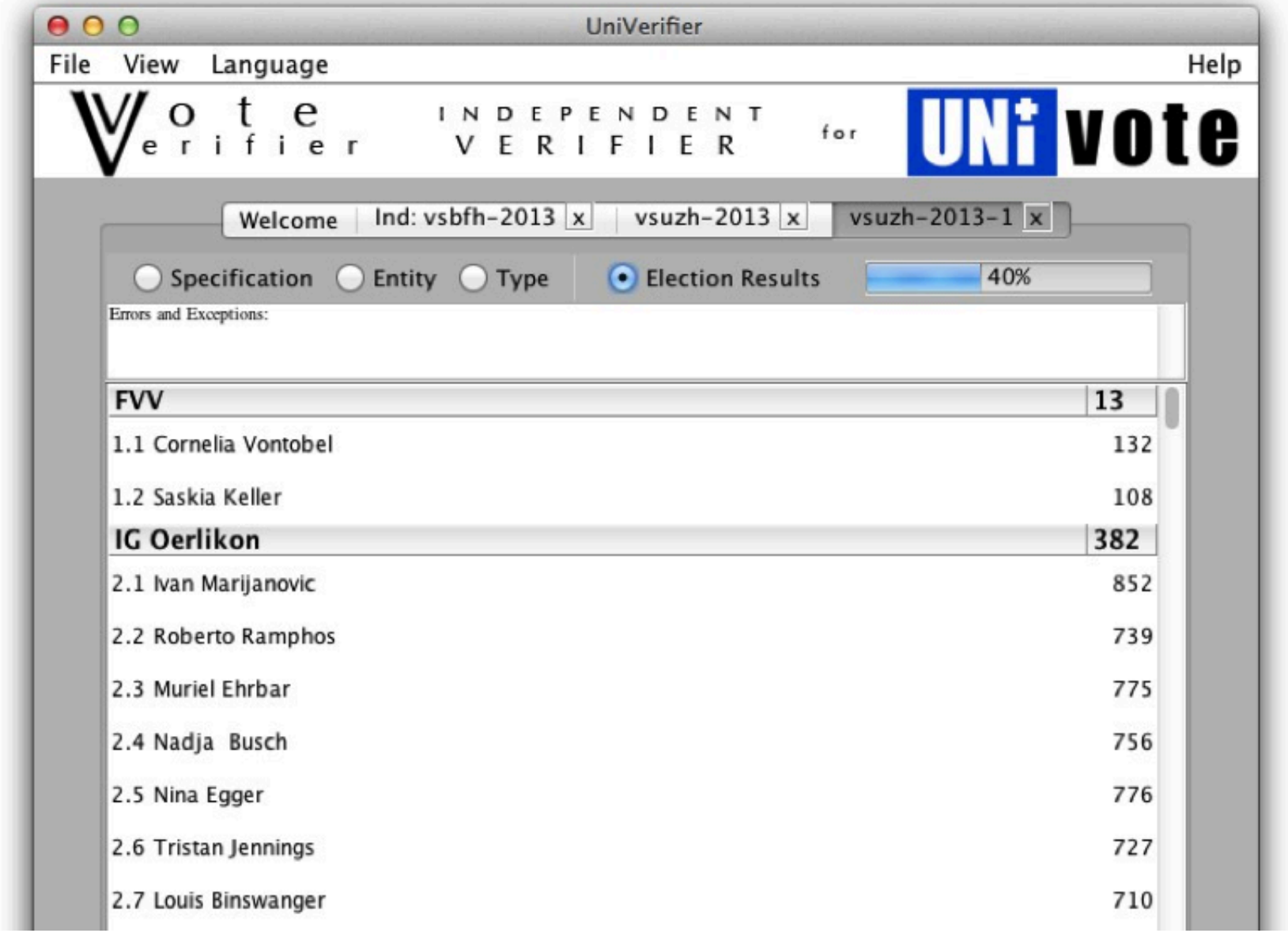
1.3.7. Mixing and Tallying

a) Mixing the Encryptions

Let $\mathcal{E}_0 = \{E_1, \ldots, E_N\}$, $N \leq n$, be the (ordered) set of encrypted votes in $\mathcal{B}$. Repeat the following steps for each $M_k \in M$ (in ascending order for $1 \leq k \leq m$):

1. Shuffle the set encrypted votes $\mathcal{E}_{k-1}$ into $\mathcal{E}_k$:
   a) Choose $\bar{r}_k = (r_{1k}, \ldots, r_{Nk}) \in_R \mathbb{Z}_q^N$ uniformly at random and compute $E'_i = ReEnc_y(E_i, r_{ik})$ for every $E_i \in \mathcal{E}_{k-1}$.
   b) Choose permutation $\tau_k : [1, N] \rightarrow [1, N]$ uniformly at random.
   c) Let $\mathcal{E}_k = \{E'_{\tau_k(i)} : 1 \leq i \leq N\} = Shuffle_{\tau_k}(\mathcal{E}_{k-1}, \bar{r}_k)$ be the new (ordered) set of encrypted votes shuffled according to τ_k.
2. Generate $\pi_{\tau_k} = NIZKP\{(\tau_k, \bar{r}_k) : \mathcal{E}_k = Shuffle_{\tau_k}(\mathcal{E}_{k-1}, \bar{r}_k)\}$ using Wikström's proof of a shuffle (see Section 1.4.7 for details).
3. Generate signature $S_{\mathcal{E}_k} = Sign_{sk_k}(id||\mathcal{E}_k||\pi_{\tau_k})$.
4. Publish $(M_k, id, \mathcal{E}_k, \pi_{\tau_k}, S_{\mathcal{E}_k})$ on EB.


UniVerifier
File View Language Help
Vote Verifier
INDEPENDENT VERIFIER for
UNIvote
Welcome
Ind: vsbfh-2013 x
vsuzh-2013 x
Specification
Entity
Type
Election Results
17%
Errors and Exceptions:
SYSTEM_SETUP
Schnorr's p is prime
Schnorr's q is prime
Schnorr's g is a generator
Schnorr's p is a safe prime
Schnorr's parameters length
CA certificate
EM certificate
ELECTION_SETUP
EA certificate


UniVerifier
File View Language Help
Vote Verifier
INDEPENDENT VERIFIER for UNIvote
Welcome
Ind: vsbfh-2013 x
vsuzh-2013 x
vsuzh-2013-1 x
Specification
Entity
Type
Election Results
40%
Errors and Exceptions:
FVV 13
1.1 Cornelia Vontobel 132
1.2 Saskia Keller 108
IG Oerlikon 382
2.1 Ivan Marijanovic 852
2.2 Roberto Ramphos 739
2.3 Muriel Ehrbar 775
2.4 Nadja Busch 756
2.5 Nina Egger 776
2.6 Tristan Jennings 727
2.7 Louis Binswanger 710

Zusammenfassung

Zusammenfassung

- Verifizierbarkeit ist zentral für die Sicherheit von E-Voting Systemen

Zusammenfassung

- Verifizierbarkeit ist zentral für die Sicherheit von E-Voting Systemen
- Es gibt technische Lösungen und Implementierungen
  - Individuell: Norwegen, Neuchâtel, Genf, Estland, UniVote
  - Universell: Verifizierung: UniVote, Helios, Victoria State (AUS)

Zusammenfassung

- Verifizierbarkeit ist zentral für die Sicherheit von E-Voting Systemen
- Es gibt technische Lösungen und Implementierungen
  - Individuell: Norwegen, Neuchâtel, Genf, Estland, UniVote
  - Universell: Verifizierung: UniVote, Helios, Victoria State (AUS)
- Herausforderungen
  - Komplexität der kryptographischen Protokolle
  - Kryptographie im Web Browser (JavaScript)
  - “Usability” und “Voter Education”
  - Stimmgeheimnis auf unsicherer Plattform
  - Stimmenkauf und Nötigung
  - Everlasting Privacy