Università degli Studi di Camerino

Scuola di Scienze e Tecnologie

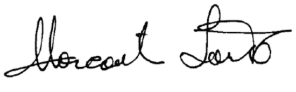
Corso di Laurea in Informatica (Classe L-31)

Studio comparativo dei sistemi di e-voting

Laureando
Alessandro Fraticelli

Matricola 097988

Relatore
Fausto Marcantoni



A.A. 2021/2022

Indice

Elenco delle figure

Elenco delle tabelle

1. Introduzione

Il voto è uno degli strumenti principali con cui le comunità compiono decisioni collettive. Nella maggior parte delle elezioni i votanti devono recarsi fisicamente in un seggio per indicare le loro scelte con carta e penna. Il voto elettronico, ossia l'uso di sistemi elettronici per l'espressione e lo scrutinio dei voti, è parte dei progressi tecnologici che ha il potenziale per rivoluzionare il modo in cui conduciamo le elezioni. Anche se alcuni approcci hanno raggiunto un alto livello di usabilità, le principali sfide del voto elettronico sono tutt'ora aperte: proteggere la privacy dei partecipanti, garantire la segretezza, l'integrità, l'unicità e l'autenticità dei voti, rendendo il voto elettronico affidabile come quello fisico.

A differenza di molte altre applicazioni in cui le tecnologie informatiche sono state ampiamente adottate per rendere le cose più efficienti, questa tipologia di voto non è molto diffusa in molti paesi. Negli Stati Uniti, un paese relativamente aperto all'adozione delle tecnologie informatiche, vengono utilizzate solamente le macchine per il voto *direct-recording electronic* e quelle a scansione ottica durante le elezioni governative. Secondo le analisi sulla situazione attuale dei sistemi di voto a distanza in tutto il mondo, la mancata adozione del voto elettronico è spesso dovuta alla sicurezza: o il sistema è effettivamente poco sicuro o è percepito come poco sicuro dalla popolazione.[Gib+16]

La sicurezza dei sistemi di voto elettronico è stata studiata in letteratura per più di quarant'anni, a partire dall'articolo di David Chaum.[Cha81] Questo argomento può essere considerato uno dei problemi più difficili nell'ambito della sicurezza informatica. Il voto elettronico comporta infatti molti requisiti e la letteratura non è ancora giunta a un insieme comune di requisiti. Ancor peggio, questi requisiti sono a volte in contraddizione tra loro per esempio, verificabilità e assenza di ricevuta.

1.1 Obiettivi

Questa tesi si propone di esplorare lo stato attuale dei sistemi di voto elettronico, i loro punti di forza e di debolezza, e riassumere un elenco di requisiti minimi che ne garantiscono l'integrità offrendo una rassegna dei protocolli più conosciuti e, infine, fornendo una descrizione dell'implementazione di uno di questi.

1.2 Struttura della Tesi

Questo lavoro si divide in cinque capitoli, ognuno dei quali analizza un tema. Nel primo ci concentreremo sulla definizione del voto elettronico, esaminando i suoi aspetti principali e i dettagli legati a questa tecnologia di voto. Successivamente, verranno approfonditi i principi e i requisiti necessari per garantire un corretto utilizzo del voto elettronico, in particolare per quanto riguarda la sicurezza e l'integrità dei sistemi.

Nel terzo capitolo, verranno elencate e descritte le diverse soluzioni di voto elettronico più comuni, esaminandone i vantaggi e gli svantaggi. Infine, nel quarto capitolo, ci concentreremo sull'implementazione pratica del sistema di voto Helios, descrivendo in dettaglio la sua installazione e il suo utilizzo, insieme a una valutazione dei suoi vantaggi e dei suoi limiti.

2. Il Voto Elettronico

2.1 Definizione

Per voto elettronico, o *e-voting* (dall'inglese *electronic voting*), si intende qualsiasi forma di elezione in cui, almeno in un momento, una copia elettronica del voto è memorizzata elettronicamente e il risultato elettorale è calcolato sulla base dei voti elettronici memorizzati.[Vol09] Esistono diverse forme di voto elettronico, tra cui il voto via Internet, il voto tramite terminale elettronico in seggio, o tramite sistemi a scansione ottica.

A seconda della specifica implementazione, il voto elettronico può prevedere l'uso di macchine specifiche per il voto elettronico o di computer connessi a Internet. Può includere una serie di servizi online, dalla trasmissione dei risultati scrutinati fino alla possibilità di votare completamente online attraverso comuni dispositivi domestici collegati in rete. Il grado di automazione può essere limitato alla semplice marcatura di una scheda cartacea o può rappresentare un sistema completo che prevede l'inserimento del voto, la registrazione del voto, la crittografia dei dati e la loro trasmissione ai server, il consolidamento e il conteggio dei risultati elettorali.

Il voto elettronico presenta alcuni vantaggi rispetto al voto tradizionale a schede cartacee, tra cui la maggiore velocità di conteggio dei voti, la riduzione dei costi di stampa e distribuzione delle schede, e la possibilità di accedere ai risultati in tempo reale. Tuttavia, ci sono anche molte preoccupazioni legate alla sicurezza del voto elettronico, tra cui la possibilità di frode informatica, la vulnerabilità ai virus informatici e agli attacchi hacker.

L'idea di sistemi di *e-voting* basati sulle *information and communications technologies* e sulle tecniche crittografiche suscita da tempo un notevole interesse da parte della comunità accademica e delle autorità pubbliche.[CB05][Moy04]

Per garantire la sicurezza del voto elettronico, è necessario utilizzare tecnologie di sicurezza avanzate, come la crittografia, per proteggere i dati dei voti durante la trasmissione e la conservazione. Inoltre, è importante che i sistemi di voto elettronico siano sottoposti a rigorosi test di sicurezza e di verifica della conformità prima dell'utilizzo in un'elezione ufficiale.

2.2 Tipi di voto elettronico

Nel discutere i vantaggi e gli svantaggi dei vari sistemi di voto elettronico è utile distinguere diverse tipologie di sistemi che si sovrappongono. Tutte le tipologie hanno diversi punti di forza e di debolezza, sia se confrontate tra loro sia se confrontate con il voto tradizionale su carta. Non esiste un sistema di voto perfetto e quelli disponibili continuano ad evolversi grazie ai continui progressi tecnologici. È quindi importante

scegliere il sistema giusto per il contesto giusto, valutando attentamente i vantaggi e gli svantaggi di tutte le opzioni.

Le tre dimensioni fondamentali per le categorie di forme elettorali sono il *supporto*, l'*ambiente* e il *momento*.[VK06a][VK06b] Di seguito vengono presentate queste dimensioni e le possibili implementazioni:

- Il **supporto** utilizzato per la scheda elettorale:
  - Il tradizionale supporto cartaceo.
  - Il nuovo supporto elettronico in cui i voti sono memorizzati elettronicamente (in bit e byte).
- L'**ambiente** in cui gli elettori esprimono il proprio voto:
  - In un ambiente supervisionato, in cui gli assistenti elettorali assicurano la correttezza, privacy e integrità del processo di voto. Questa categoria è chiamata anche voto in presenza. Gli ambienti supervisionati sono, ad esempio, i tradizionali seggi elettorali, gli uffici postali o, nel caso di cittadini che vivono all'estero, le ambasciate
  - In un ambiente non supervisionato dove, in assenza di assistenti elettorali, l'elettore stesso deve assicurare la correttezza, privacy e integrità del processo di voto. Questa categoria è chiamata anche voto da remoto. Votare in un ambiente non supervisionato comporta che l'elettore non abbia un contatto diretto con i funzionari elettorali. Pertanto, presenta maggiori rischi di coercizione e di voto di scambio.
- Il **momento** in cui è possibile votare:
  - Il giorno delle elezioni;
  - Prima del giorno delle elezioni (chiamato anche voto anticipato).

Alcuni sistemi elettorali possono appartenere a più categorie della stessa dimensione, a seconda del periodo dell'implementazione.

Sebbene le categorie siano state ampiamente esplorate e molti sistemi siano stati testati in casi reali con vari gradi di successo, le soluzioni esistenti presentano alcuni limiti, rendendo impegnativa la scelta del sistema più adatto a uno scenario specifico.

2.2.1 Categorie di forme elettorali

È possibile individuare le seguenti forme elettorali associabili al voto elettronico:

Sistemi di voto elettronico su carta

Le schede cartacee sono la forma di raccolta di voti più diffusa, ben conosciuta, accettata, economica e facile da usare.

Il principale svantaggio delle tradizionali elezioni che fanno uso di schede cartacee è che il processo di conteggio è tipicamente fatto a mano e quindi lungo e soggetto a errori. Questo è uno dei principali argomenti a favore dell'uso di dispositivi elettronici di voto.

Il processo di verifica di un'elezione con schede cartacee contate a mano è molto semplice. Si verifica che l'urna sia vuota all'inizio della fase di voto, che solo gli elettori idonei abbiano espresso il proprio voto e che ogni elettore abbia espresso al massimo un voto. Durante lo scrutinio, si verifica che le schede siano state contate correttamente. Il problema di questo processo di verifica è che un singolo scrutinatore è in grado di controllare al massimo un seggio elettorale. Un altro svantaggio è che le schede cartacee non offrono alcun feedback all'elettore. In un'elezione complessa in cui l'elettore può applicare il voto disgiunto o esprimere più preferenze, distribuire un gran numero di voti, egli potrebbe rendere il proprio voto non valido, senza che questa irregolarità venga segnalata.

Con il voto elettronico basato su carta, gli elettori esprimono il loro voto tramite supporto cartaceo, ma la scansione (cioè la registrazione e l'archiviazione come voto elettronico) e il conteggio dei voti avvengono elettronicamente. La scansione può essere effettuata direttamente dopo l'espressione del voto o alla fine del periodo elettorale. Il voto può essere espresso in un ambiente supervisionato o non supervisionato, durante il giorno delle elezioni o in anticipo.

Un vantaggio di questi sistemi è che, in caso di problemi tecnici, i voti cartacei possono essere contati a mano e non è necessaria la ripetizione del voto. Lo svantaggio è che l'autorità elettorale responsabile deve decidere e definire nel regolamento elettorale se i voti cartacei o i voti elettronici siano i voti ufficiali. Nel caso in cui i voti cartacei siano quelli legali, il conteggio rapido dei voti elettronici fornisce solo risultati non ufficiali e il risultato deve essere successivamente confermato dal conteggio dei voti cartacei. Nel problematico caso in cui il risultato elettronico e quello cartaceo non fossero identici, è necessario tenere conto dei tassi di errore del lettore. I tipi di sistemi di voto elettronico e cartaceo più diffusi sono:

- **Sistemi a schede perforate**. Gli elettori bucano la scheda (con un dispositivo di punzonatura fornitogli) in corrispondenza della loro scelta. Successivamente, il voto viene letto nel dispositivo di conteggio e l'elettore ripone la scheda in un'urna. In questo caso, oltre al tasso di errore del lettore, occorre considerare anche la possibilità che l'elettore non effettui correttamente l'operazione di perforatura. Per maggiori informazioni si veda, ad esempio.[BGE07] Alcuni esempi di tali sistemi sono: Portapunch, Datavote e Votomatic.

- **Sistemi a scansione ottica**. Gli scanner a scansione ottica sono stati sviluppati per la somministrazione di esami di ammissione all'università e altri test standardizzati e successivamente sono stati applicati alle elezioni.[Jon10] In questi sistemi, l'elettore esprime il proprio voto su carta, di solito riempiendo un cerchio o un ovale. Questi voti cartacei vengono letti da uno scanner ottico. Le schede utilizzate per questo tipo di voto non differiscono enormemente da quelle classiche. Il vantaggio di questi dispositivi è che il processo di conteggio è molto più veloce di quello manuale. Oltre che dagli errori di scansione, i sistemi a scansione ottica sono ostacolati da schede cartacee marcate in modo errato. I segni non conformi alle regole possono causare errori nei conteggi. Esempi di questo sistema sono Prêt à Voter [CRS05] [RS06], il sistema Three-Ballot [Riv06] e il sistema Scantegrety [Cha+08].

Macchine per il voto elettronico stand-alone nei seggi elettorali

Le macchine per il voto elettronico stand-alone sono dispositivi elettronici che consentono di esprimere il voto, spesso utilizzando un *touch screen*, e di memorizzarlo localmente. Questi dispositivi sono dei terminali, simili a bancomat, che si trovano nei seggi elettorali, dunque in un ambiente controllato. Inoltre, non sono collegati a Internet o ad altre reti. L'autorizzazione di voto all'elettore idoneo viene effettuata su carta come nelle elezioni tradizionali. Analogamente a quanto avviene nelle tradizionali elezioni in un seggio elettorale, le macchine per il voto elettronico autonome possono essere utilizzate per votare il giorno delle elezioni o in anticipo nelle ambasciate.

Nel contesto del voto elettronico supervisionato, vengono spesso utilizzati due diversi tipi di sistemi: i sistemi *direct-recording electronic* (DRE) e i sistemi di voto computerizzati.[OP17]

I sistemi DRE dipendono da dispositivi hardware proprietari, spesso costosi. Inoltre, non è raro che tali dispositivi presentino vulnerabilità che mettano a rischio la segretezza, la privacy e il corretto svolgimento delle elezioni.[Fra+11] [Bed+03] [Koh+04] L'alternativa è quella di utilizzare come postazioni di voto comuni PC, laptop portatili o tablet, con un software appositamente progettato. Il vantaggio di utilizzare sistemi personalizzati è che possono essere adattati a esigenze specifiche e soddisfare qualsiasi tipo di richiesta.

L'adozione di tali soluzioni è molto conveniente in scenari di media o grande scala, dove il costo dell'acquisto di nuovo hardware può essere ammortizzato su più elezioni. In un contesto universitario, ad esempio, questa sembra la soluzione più conveniente e il software può essere sviluppato tenendo conto delle caratteristiche peculiari di tale scenario.

Il problema principale, tuttavia, è che senza ulteriori strumenti di verifica del risultato prodotto dalla macchina elettorale al termine della fase di voto, gli elettori e le autorità di voto non hanno modo di controllare la correttezza del conteggio. In America, questo ha portato a richiedere che, oltre alla scheda elettronica, ci fosse una copia fisica della scelta di ciascun elettore. L'implementazione più comune è il VVPAT (Voter-Verified Paper Audit Trail) [VWT09] [Hal06] [CG09].

Macchine per il voto elettronico in rete nei seggi elettorali

Le macchine per il voto elettronico in rete nei seggi elettorali sono macchine per il voto elettronico collegate a Internet. Ciò consente agli elettori di votare in qualsiasi seggio elettorale abilitato, senza essere limitati a uno specifico seggio assegnato. Pertanto, questa forma di elezione offre all'elettore una maggiore flessibilità rispetto ad altre forme di elezione in seggi elettorali. Il diritto di voto dell'elettore viene verificato online su un registro elettorale centrale. Questa verifica può essere effettuata sia senza un token di autenticazione elettronico, necessitando però di assistenti elettorali che operano secondo le procedure delle elezioni tradizionali, sia dalla macchina elettronica di voto in rete.

Macchine per il voto elettronico a chiosco

Le macchine per il voto elettronico a chiosco sono simili alle macchine per il voto elettronico in rete nei seggi elettorali. La differenza sta nell'ubicazione: la versione "*kiosk*" descritta da Volkamer non è disposta in un vero e proprio seggio elettorale, ma

in luoghi centrali come scuole, università, biblioteche o supermercati. Si compie perciò il primo passo verso un ambiente non controllato, poiché non sono più gli operatori a garantire, la privacy e l'integrità del processo di voto. A causa dell'assenza di operatori elettorali, le macchine devono essere protette come i bancomat.

Voto elettronico a distanza

Il voto elettronico a distanza è la controparte elettronica del voto per corrispondenza. In alcuni testi viene chiamato anche *Internet-Voting* o *Online-Voting*. Il voto elettronico a distanza consente agli utenti di votare senza essere fisicamente presenti in ambienti sorvegliati, purché dispongano di un dispositivo collegato alla rete. Tale dispositivo può essere un PC o un telefono cellulare. Il voto elettronico a distanza può essere impiegato solo se il controllo dell'autorizzazione e l'espressione del voto avvengono online e, quindi, per via elettronica. Si tratta quindi della forma elettorale più complessa e difficile da attuare, in quanto combina i problemi di ambienti non controllati con tutti i problemi dei sistemi *direct-recording electronic*. Inoltre, il software di voto lato *client* viene eseguito su una macchina e un sistema operativo al di fuori del controllo dell'autorità elettorale responsabile. È possibile implementare il voto elettronico a distanza per esprimere un voto in anticipo o il giorno delle elezioni e comunque contare i voti elettronici il giorno stesso delle elezioni.

Questa possibilità è generalmente più attraente per gli utenti per via della comodità di votare da casa. Purtroppo, questo approccio presenta i due seguenti problemi principali.

I sistemi di voto elettronico remoto non possono garantire completamente l'incoercibilità degli elettori. Infatti, per un sistema remoto, è impossibile garantire che gli utenti siano soli al momento del voto.[Wan+17]

Un altro svantaggio di questo tipo di voto è che l'elettore deve avere fiducia nel software in utilizzo e nel fatto che il suo voto venga effettivamente conteggiato. Attualmente, questo problema viene talvolta risolto utilizzando procedimenti che consentono agli utenti di verificare che il loro voto sia stato conteggiato correttamente; ad esempio, i sistemi di voto elettronico remoto fanno spesso uso di ricevute virtuali.

Nonostante queste limitazioni, e data la loro grande convenienza, molti ricercatori hanno concentrato i loro sforzi sullo sviluppo di nuovi sistemi di voto elettronico basati su Internet che consentono agli elettori di partecipare a distanza, utilizzando applicazioni web ad hoc.[Wu+14] [RH10] [AP14] [TSP13] [YO13]

Blockchain Recentemente, i principali sforzi di ricerca si sono concentrati sull'uso di blockchain per l'implementazione di sistemi di voto.[HR17] [Hja+18] [KV18] [MW20] [Agg+19].

L'idea chiave dell'utilizzo della tecnologia *blockchain* come base di un sistema di voto è quella di dotare gli elettori di un portafoglio virtuale. Analogamente alla moneta virtuale, ogni elettore riceve una singola moneta, che rappresenta il diritto di voto, che può essere trasferita al portafoglio del candidato per esprimere una preferenza. Tuttavia, l'uso di questa tecnologia solleva spesso preoccupazioni, poiché non è ancora familiare agli utenti. Inoltre, oltre ai problemi di fiducia, le *blockchain* richiedono molta energia per eseguire operazioni essenziali come l'autenticazione e la convalida dei blocchi e sono piuttosto lente. Pertanto, l'utilizzo di tale tecnologia per elezioni su larga scala potrebbe non essere ancora pratico.[KV18]

Dal punto di vista della fattibilità, alcuni sistemi non supervisionati presentano vantaggi rispetto a quelli supervisionati in termini di maggiore comodità nel votare a distanza, risparmio di tempo e riduzione dei costi di hardware. Tuttavia, i sistemi di voto elettronico a distanza non possono garantire le stesse rigorose proprietà di sicurezza necessarie per il corretto svolgimento di elezioni a livello nazionale. Lo stesso Ronald L. Rivest ha affrontato di recente questo argomento, criticando aspramente il voto via Internet e i sistemi basati su *blockchain*.[Par+21]

Soluzioni *cloud based* Zissis e Lekkas[ZL11] hanno analizzato i vantaggi e gli svantaggi dell'adozione di una soluzione cloud per i sistemi informativi di *e-government*. Un mix di *cloud computing* e tecniche crittografiche può essere utilizzato per affrontare una serie di minacce tipiche dei sistemi di voto elettronico. Purtroppo, questa tecnologia sembra essere lontana dall'uso reale, poiché spesso le persone non ripongono abbastanza fiducia nella gestione dei dati in cloud.[XX13]

<table>
<tr><th></th><th>Ambiente supervisionato</th><th>Ambiente non supervisionato</th></tr>
<tr><td rowspan="2">Supporto cartaceo</td><td colspan="2">Sistemi a schede perforate</td></tr>
<tr><td colspan="2">Sistemi a scansione ottica</td></tr>
<tr><td rowspan="4">Supporto elettronico</td><td>Macchine per il voto elettronico in rete nei seggi elettorali</td><td></td></tr>
<tr><td>Macchine per il voto elettronico stand-alone nei seggi elettorali</td><td></td></tr>
<tr><td></td><td>Macchine per il voto elettronico a chiosco</td></tr>
<tr><td></td><td>Voto elettronico a distanza</td></tr>
</table>

Tabella 2.1: Categorie elettorali (classificate secondo le tre dimensioni)

Altri lavori hanno presentato sistemi di voto elettronico che offrono all'utente un'esperienza simile a quella dei sistemi tradizionali, migliorando la soddisfazione degli utenti grazie all'aumento della loro percezione di integrità, sicurezza e fiducia [YM07] [BC08] [Ant+07] [AMP19] [KDS19]. Nel 2014 è stato condotto uno studio sulle caratteristiche che possono migliorare la fiducia degli utenti, come l'usabilità del sistema. Quando un sistema di voto elettronico è troppo diverso da quelli tradizionali, gli utenti possono diffidare del meccanismo sottostante.[Pom+14]

La Tabella 2.2 riassume le caratteristiche più rilevanti di diversi sistemi di voto appartenenti alle categorie precedentemente descritte, da quelli supervisionati, come Prêt-à-Voter, fino a quelli che permettono agli utenti di votare da remoto, come Helios.

La stessa tabella mostra che la maggior parte di questi sistemi non è open source. Questo punto è critico, poiché uno dei motivi per cui gli utenti non ripongono fiducia nei sistemi di voto elettronico è che le applicazioni basate su software proprietario non possono essere analizzate e verificate da esperti e utenti comuni. Un esempio lampante di questo problema è rappresentato da Voatz, le cui principali falle di sicurezza sono state identificate attraverso il *reverse engineering*, data la totale mancanza di codice sorgente e di documentazione adeguata.[SKW20] Nei sistemi open source, al contrario, l'analisi continua del codice sorgente da parte di esperti di sicurezza consentirebbe di individuare molto prima i potenziali problemi, aumentando la fiducia degli utenti.

	Supervisionato	Voti nascosti da elettori nascosti	Assenza di ricevute	Resistenza alla coercizione	Indipendenza del software	Verificabilità individuale	Verificabilità universale	Cast-as-intended	Recorded-as-cast	Tallied-as-recorded	Open Source
Bingo Voting	●	●	●	●	●	●	●	●	●	●	-
Helios	-	-	-	-	●	●	●	●	●	●	●
Prêt à Voter	●	●	●	-	●	●	●	●	●	●	-
Civitas	-	●	●	-	-	●	●	-	●	●	-
Polyas	-	●	●	-	●	●	●	●	●	●	-
SecureBallot	●	●	●	●	●	●	●	●	●	●	●
Sis. Estone	-	●	-	-	●	●	●	-	●	●	◖

● = fornisce la proprietà; ◖ = fornisce parzialmente la proprietà; - = non fornisce la proprietà;

Tabella 2.2: Tabella comparativa dei sistemi di voto elettronico che saranno presi in esame.

2.3 Tipi di attacchi ai sistemi di voto

Ci sono molti modi per influenzare un risultato elettorale. Oltre ai metodi legali, come ad esempio, le campagne elettorali, esistono modi più diretti e illegali per convincere gli elettori a votare in un certo modo o per influenzare o cambiare direttamente il risultato delle elezioni. Questa sezione presenta brevemente diversi attacchi ai sistemi di voto che i progettisti di sistemi elettronici devono tenere in conto. Chiameremo da qui in avanti "avversario" l'entità che tenta di manipolare un'elezione e che conduce gli attacchi.

2.3.1 Ballot Stuffing

Per *ballot stuffing* si intende una forma di frode elettorale in cui viene espresso un numero di schede superiore al numero di persone che hanno legittimamente votato. Il termine si riferisce in generale all'atto di esprimere voti illegali o di presentare più di una scheda per elettore quando ne sia consentita una sola.

2.3.2 Voto di scambio

L'obiettivo di un avversario che intenda manipolare l'esito di un'elezione può essere raggiunto non solo attraverso la diretta manomissione del risultato, ma anche tramite la coercizione degli elettori. Se il sistema di voto consente a un avversario di verificare se l'elettore ha votato in conformità ad un accordo predefinito, ciò potrebbe essere sfruttato per manipolare il comportamento dell'elettore, ricompensando o punendo le azioni desiderate o indesiderate. Questo tipo di attacco è noto come voto di scambio o coercizione e rappresenta una minaccia per la privacy degli elettori, poiché consente all'avversario di comprare voti e alterare il risultato dell'elezione illegalmente.

Il voto di scambio risulta essere il più efficace tra gli attacchi di questo tipo, poiché l'elettore è motivato a deviare dal protocollo di voto dietro compenso.

2.3.3 Chain voting

Un sistema di voto che utilizza schede cartacee e consegna a ciascun elettore una sola scheda è spesso vulnerabile al seguente attacco: l'avversario ottiene una singola scheda, la contrassegna secondo la sua scelta e la consegna a un elettore. L'elettore viene ora prescritto (dietro minaccia o pagamento) a votare utilizzando la scheda già segnata ricevuta dall'avversario e consegnare la scheda bianca che riceve al seggio elettorale all'avversario. Quest'ultimo potrà ripetere l'attacco compilando la nuova scheda bianca e costringendo un altro elettore.

3. Principi e requisiti di un sistema di *e-voting*

I requisiti di un sistema di *e-voting*, sono un elemento fondamentale per garantirne la sicurezza e la segretezza del voto. A contrastare l'introduzione dell'*e-voting* è anche la diffidenza dell'opinione pubblica e delle autorità circa l'affidabilità e sicurezza dei sistemi in uso per offrire questo servizio all'interno del quadro giuridico e costituzionale esistente. Uno degli obiettivi principali degli sviluppatori di sistemi di *e-voting* è quello di garantire un adeguato livello di sicurezza[Cra+96] [Sch99] [But00] [HC01] [FS00] [VM07] [Neu93] [Gri02].

In letteratura, i requisiti sono solitamente identificati come: legali, tecnici e orientati all'utente, questi ultimi sotto forma di condizioni che il sistema dovrebbe soddisfare (ad esempio, il sistema deve consentire il voto online da casa). Di seguito si presenta il modello generico di *e-voting* sviluppato da Dimitris A. Gritzalis in "*Principles and requirements for a secure e-voting system*", illustrando i principi e le pratiche da seguire durante una procedura elettorale.

Altri studi hanno evidenziato criteri standard di sicurezza, privacy, libertà e universalità che tutti i sistemi di voto elettronico dovrebbero garantire. La definizione formale di questi criteri, così come la loro applicazione, sono essenziali per progettare e sviluppare sistemi di voto che possano essere utilizzati in un contesto reale.

3.1 Problemi principali del voto elettronico

Una sfida fondamentale nell'utilizzo di sistemi informatici in questo contesto è quella di migliorare e sviluppare la democrazia rappresentativa e rafforzare i processi che mirano al dare più potere decisionale ai cittadini, rispettando i principi e i valori della democrazia. A questo proposito, un fenomeno da tenere in considerazione è il divario digitale[1]. Al fine di diminuire questo divario, è necessario adottare politiche specifiche. L'Unione Europea, ad esempio, ha adottato tre azioni chiave:

[1] "Divario digitale" è un termine che descrive la distribuzione diseguale delle tecnologie dell'informazione e della comunicazione nella società. Esso comprende le differenze di accesso (divario digitale di primo livello) e di utilizzo (divario digitale di secondo livello) dei computer e di Internet tra

- i Paesi industrializzati e quelli in via di sviluppo (divario globale)
- i vari gruppi socioeconomici all'interno dei singoli Stati nazionali (divario sociale)
- i diversi tipi di utenti in relazione al loro impegno politico su Internet (divario democratico).

In generale, si ritiene che queste differenze rafforzino le disuguaglianze sociali e causino il persistere di un divario di informazioni o conoscenze tra le persone che hanno accesso ai nuovi media e li utilizzano e quelle che non ne hanno.[Sch23]

1. adattare il quadro normativo esistente alle esigenze dell'industria delle comunicazioni in Internet;

2. stimolare la concorrenza nelle reti di accesso locali, in modo da incoraggiare la diffusione di Internet e l'accesso a Internet ad alta velocità in Europa;

3. garantire un elevato standard di diritti degli utenti e di protezione della privacy.[Eur]

Garantire che le tecnologie di voto tradizionali e quelle elettroniche siano equivalenti, per quanto riguarda la facilità e l'opportunità di accesso è una questione di democrazia, uguaglianza ed equità. Le elezioni, perché possano essere considerate democratiche, devono essere libere, uguali e segrete. Allo stesso tempo, la procedura elettorale deve essere trasparente e soggetta al controllo pubblico.

Le costituzioni di molti Paesi richiedono che le elezioni generali rispettino la generalità, la libertà, l'uguaglianza, la segretezza e l'immediatezza. Aggiungendo ad essi il requisito fondamentale della democrazia, si ottiene un insieme di requisiti costituzionali generici per il voto. Questo insieme riflette, a sua volta, l'insieme dei principi essenziali di progettazione del voto (Tabella 3.1).

Requisiti costituzionali	**Principi di progettazione dei sistemi di voto**
Generalità	Isomorfo al tradizionale
	Idoneità
Libertà	Incoercibilità
	Nessuna propaganda nel sito di voto elettronico
	Possibilità di voto non valido
Uguaglianza	Uguaglianza dei candidati
	Uguaglianza degli elettori
	Un elettore - un voto
Segretezza	Segretezza
	Equilibrio tra sicurezza e trasparenza
Immediatezza	Registrazione e conteggio delle schede non monitorati
Democrazia	Affidabilità e trasparenza
	Verificabilità e responsabilità
	Affidabilità e sicurezza
	Semplicità

Tabella 3.1: Requisiti costituzionali e principi di progettazione proposti da Dimitris Gritzalis

3.2 Principi di progettazione dei sistemi di voto

3.2.1 Generalità

Dal principio democratico del suffragio universale, cioè che ogni elettore avente diritto debba essere incluso nel processo elettorale, deriva la necessità della disponibilità pubblica di infrastrutture adeguate.

Il voto elettronico dà la possibilità di migliorare la generalità delle procedure elettorali fornendo un'ulteriore opzione di partecipazione al processo elettorale. Una questione

che si pone è se la partecipazione alle elezioni attraverso il voto elettronico debba essere soggetta a condizioni speciali, come nel caso del voto per corrispondenza. Nella maggior parte dei Paesi in cui è stato istituito il voto per corrispondenza, solo specifiche categorie di persone sono autorizzate a esercitare questa opzione.[Ide]

L'adozione di una funzionalità di voto elettronico come eccezione alla regola (cioè sulla base della prova di una condizione che impedisce all'elettore idoneo di esprimere fisicamente il proprio voto) è generalmente considerata accettabile. Alla luce della decisione politica di migliorare l'*e-government* e la partecipazione elettronica, l'introduzione di una funzionalità di voto elettronico dovrebbe essere vista come un *isomorfismo* del sistema di voto tradizionale.[2]

L'idoneità può essere garantita, come nel processo di voto tradizionale, attraverso la registrazione degli elettori idonei e la loro identificazione, procedure essenziali per garantire il rispetto del principio del suffragio universale e per prevenire brogli elettorali.

Un'altra questione è se sia necessaria la registrazione nel caso del voto elettronico. Essendo il voto elettronico analogo al voto per corrispondenza, saranno necessarie procedure di registrazione e autorizzazione per le seguenti ragioni:

1. supponendo che non esista un registro nazionale degli elettori online, è necessaria una pre-registrazione per il voto elettronico, per evitare frodi elettorali e sostenere l'integrità delle elezioni. D'altra parte, un sistema di registrazione degli elettori basato su Internet potrebbe essere vulnerabile a brogli su larga scala.

2. nel caso in cui il voto elettronico sia un'alternativa alla procedura tradizionale, la registrazione o la dichiarazione che l'elettore desidera utilizzare l'opzione del voto elettronico non dovrebbe portare all'esclusione o alla discriminazione. Inoltre, si dovrebbe garantire che sia facile per gli elettori elettronici registrarsi, identificarsi e autenticarsi, perché procedure complicate potrebbero essere un peso per loro.[Eur10]

3.2.2 Libertà

Il principio delle libere elezioni richiede che il processo elettorale si svolga senza alcuna violenza, coercizione, pressione, intervento manipolativo o qualsiasi altra influenza, esercitata sia dallo Stato che da uno o più individui.[Res12] Le procedure di voto elettronico possono lasciare spazio a nuove minacce alla libertà e all'integrità della decisione dell'elettore, ad esempio, nell'ipotesi del voto da remoto, il luogo nel quale si esprime la preferenza e i dispositivi usati possono influirne la segretezza, nel caso l'elettore effettui questa operazione dal luogo di lavoro, anche se il datore di lavoro, il supervisore o un collega non si trovano alle spalle del dipendente che vota con l'*e-voting*, gli amministratori di sistema possono monitorare o registrare l'attività di ogni postazione di lavoro e ottenere traccia del voto.

L'incoercibilità e la prevenzione del voto di scambio possono essere garantite da un sistema di voto elettronico progettato in modo che nessun elettore possa dimostrare di aver votato in un determinato modo (non tracciabilità da parte dell'elettore)[Eur10]. Per evitare il rischi causati da squilibri di potere come nell'esempio precedente del luogo di lavoro, Gritzalis suggerisce di evitare il voto elettronico dal luogo di lavoro attraverso

[2] "Si parla di isomorfismo quando due strutture complesse si possono applicare l'una sull'altra, cioè far corrispondere l'una all'altra, in modo tale che per ogni parte di una delle strutture ci sia una parte corrispondente nell'altra struttura; in questo contesto diciamo che due parti sono corrispondenti se hanno un ruolo simile nelle rispettive strutture."[Hof13]

lo sviluppo di un'infrastruttura pubblicamente accessibile, che consenta agli elettori di esercitare i propri diritti senza la coercizione di terzi.[Gri02]

La procedura di voto elettronico dovrebbe anche rendere tecnicamente impossibile la pubblicità di entità politiche sulla piattaforma di voto. Deve essere inoltre garantita la libera espressione delle preferenze dell'elettore, pertanto, anche la possibilità di esprimere una scheda consapevolmente non valida (o scheda bianca).

3.2.3 Uguaglianza

Il principio dell'uguaglianza del suffragio, si compone di due requisiti principali:

1. l'uguaglianza dei partiti politici e dei candidati partecipanti
2. l'uguaglianza dei diritti di voto di ciascun elettore.

Di conseguenza le schede elettroniche dovranno essere impostate e visualizzate in modo analogo a quelle utilizzate per le schede cartacee. L'uguaglianza elettorale richiede che non ci siano deviazioni significative tra la scheda stampata e il suo equivalente elettronico. Pertanto, il *look and feel* della piattaforma di voto e delle schede elettorali non deve favorire o discriminare nessuno dei partiti partecipanti. L'uguaglianza tra i partecipanti prescrive anche che il voto dell'elettore sia trasmesso e conteggiato senza alcuna modifica o interferenza. Un voto valido non deve essere alterato o rimosso nel corso del processo di voto.

Si deve inoltre sostenere la trasparenza. Tutte le parti devono avere la possibilità di accedere in modo paritario agli elementi della procedura di voto, per poterne stabilire il corretto funzionamento.

Il principio di uguaglianza richiede che ogni voto, sia esso fisico o online, abbia lo stesso peso ai fini del risultato elettorale. In una situazione di voto elettronico, alcuni elettori hanno un vantaggio di accesso alla tecnologia in uso e, quindi, alla capacità di voto elettronico. A causa di ciò il diritto alla pari accessibilità al processo di voto dovrebbe diventare il diritto alla pari accessibilità alla tecnologia elettorale.[Goo+18] Pari accessibilità significa anche che il sistema deve essere facile da usare indipendentemente dall'istruzione, dall'età e dalle condizioni fisiche dell'elettore.

Un sistema di voto elettronico deve garantire il rispetto del principio "un elettore - un voto", ovvero che solo gli elettori idonei possano votare, una sola volta, online o non. Pertanto, un sistema di voto elettronico deve essere progettato in modo tale da impedire:

1. la duplicazione del voto (da parte dell'elettore stesso o di qualcun altro).
2. la riutilizzazione del voto (votando online più di una volta o votando sia online che offline).
3. la modifica del voto espresso (dopo che l'elettore ha espresso il suo voto).

3.2.4 Segretezza

La segretezza del voto è ampiamente riconosciuta come un diritto umano fondamentale ed è sancita dall'articolo 21 della Dichiarazione universale dei diritti umani [Uni48]. La logica alla base di questo principio, che risale all'antica Grecia e a Roma [Yak95], è che se soggetti esterni vengono a conoscenza della scelta di un elettore, si apre la porta alla

corruzione e all'intimidazione, finendo per falsare il processo elettorale. L'acquisto di voti non era raro nel mondo occidentale fino al secolo scorso [Sto+13] e persiste ancora in alcuni Paesi in via di sviluppo [JJ14] [GO+12]. Allo stesso modo, gli elettori possono essere intimiditi da criminali, politici locali o persino da membri della famiglia, affinché votino in un certo modo. Questi timori hanno motivato direttamente la nozione di voto segreto, che oggi viene tipicamente attuata mettendo a disposizione dell'elettore una cabina elettorale privata presso il seggio elettorale. La scheda elettorale non reca alcun segno distintivo e il suo voto viene espresso nell'urna dove si mescola a tutti gli altri voti, rendendo molto difficile, se non impossibile, accertare la propria scelta.

La segretezza e la libertà sono principi strettamente correlati. L'elettore ha diritto di isolarsi al momento di votare e deve mantenere il segreto su quanto avviene in quello stesso momento; prima e dopo il momento materiale del voto egli è invece del tutto libero di dichiarare pubblicamente per chi voterà o ha votato.[Res12] Nelle elezioni democratiche il legame tra il voto e l'elettore dovrebbe essere irreversibile per garantire che i voti siano espressi liberamente. Nei sistemi di voto tradizionali la segretezza è protetta fisicamente, ma il voto elettronico può rendere vulnerabili le elezioni alle violazioni della segretezza. Da quanto sopra derivano i seguenti requisiti:

- La segretezza del voto deve essere garantita durante l'espressione, il trasferimento, la ricezione, la raccolta e lo scrutinio dei voti;
- Nessuno degli attori coinvolti nel processo di voto (organizzatori, funzionari elettorali, terze parti fidate, elettori, ecc.) dovrebbe essere in grado di collegare un voto a un elettore identificabile;
- Dovrebbe esserci una chiara separazione tra le procedure di registrazione e di autenticazione, da un lato, e l'espressione e il trasferimento del voto, dall'altro;
- Nessun elettore dovrebbe essere in grado di dimostrare di aver votato in un determinato modo.

Il sistema elettorale deve consentire la verifica dell'autenticità della scheda prima che i voti siano scrutinati o contati. Per proteggere la segretezza, le schede votate dovrebbero essere decriptate e contate solo dopo che le informazioni di autenticazione sono state esaminate e rimosse. Il sistema di voto elettronico deve rendere tecnicamente possibile il controllo e il riconteggio dei voti, garantendo al contempo la non identificabilità degli elettori.[Tel01][Sch00]

3.2.5 Immediatezza

Il principio dell'elezione diretta prevede che non vi siano intermediari nel processo di decisione del voto. Questo principio può essere adattato anche a una procedura di voto elettronico. Il requisito fondamentale è che ogni singolo voto online venga registrato e conteggiato direttamente.

Un problema può sorgere nel caso in cui il periodo di votazione differisca dalla procedura di voto (online o offline) utilizzata per esprimere il voto. I risultati del voto online possono influenzare l'esito dell'intero processo elettorale e limitare l'integrità e la legittimità dell'intero processo. Per evitare ciò, è possibile sviluppare un sistema che consenta la registrazione e il protezione del voto espresso, vietando però il conteggio prima della fine delle elezioni (offline).

3.2.6 Democrazia

Un sistema di voto elettronico democratico dovrebbe soddisfare almeno i requisiti di un sistema elettorale tradizionale e i nuovi derivanti dalla natura del voto elettronico. Questi requisiti riguardano la conservazione di attributi e proprietà come la trasparenza, la responsabilità, la sicurezza, l'accuratezza e la legittimità del sistema. Le procedure di voto tradizionali operano in modo trasparente sia per gli elettori che per gli altri attori delle elezioni. Al contrario, le procedure di voto elettronico non sono trasparenti poiché l'elettore medio non ha le conoscenze necessarie per comprendere il funzionamento il sistema. Pertanto, nel voto elettronico gli elettori devono avere molta più fiducia nella tecnologia utilizzata e nelle persone coinvolte (funzionari elettorali, fornitori di servizi tecnologici, ecc.).

- La verificabilità è un requisito in conflitto con la trasparenza. Un sistema di voto elettronico dovrebbe consentire la sua verifica da parte degli elettori (verificabilità individuale) o da parte di funzionari elettorali, partiti e osservatori indipendenti (verificabilità istituzionale).
- La responsabilità è un ulteriore requisito di un sistema di voto elettronico, inteso come registrazione e monitoraggio di tutte le operazioni relative al voto elettronico.
- I requisiti di affidabilità e sicurezza derivano dall'esigenza democratica di garantire che il risultato delle elezioni rifletta correttamente la volontà degli elettori. La scheda trasmessa all'apparecchiatura di conteggio dei voti deve essere una copia accurata e non modificabile della scelta dell'elettore. Inoltre, dovrebbe essere impossibile escludere un voto valido dal conteggio e, al contrario, convalidare un voto non valido.
- Per sicurezza si intende principalmente al rispetto dell'integrità e della accessibilità del sistema, ma anche a tutta una serie di funzioni e componenti elettorali, come la registrazione, l'idoneità e l'autenticazione. Il sistema di voto elettronico deve essere protetto da interruzioni del servizio accidentali o intenzionali e deve essere disponibile per l'uso ogni volta che si prevede che sia operativo. La mancata disponibilità del sistema (o di uno dei suoi componenti) può comportare la perdita della capacità di un elettore di esercitare i propri diritti politici fondamentali.

I sistemi di voto elettronico sono inevitabilmente complicati. Inoltre, coinvolgono solitamente più attori rispetto a un sistema tradizionale. Dal punto di vista degli elettori, il sistema dovrebbe essere facile da usare e non dovrebbe richiedere particolari competenze. Pertanto, un sistema di voto elettronico dovrebbe essere sviluppato in modo da facilitarne l'usabilità e preservarne la controllabilità. La semplicità e l'accessibilità di un sistema si rivelano quindi questioni (non solo tecniche) di grande importanza.

Sulla base dei principi sopra esposti, Gritzalis ricava i seguenti requisiti:

- Devono esistere procedure di certificazione affidabili per l'hardware e il software del sistema;
- l'intera infrastruttura, così come ogni funzionalità del sistema, deve essere registrata (ad esempio, tutto il software che non sia di interfaccia deve essere open source);

- tutte le operazioni (autenticazione, registrazione dei voti, ecc.) devono essere monitorate, mentre la segretezza deve essere garantita;
- l'infrastruttura deve essere aperta all'ispezione da parte di organismi autorizzati;
- gli elettori, i partiti e i candidati devono avere la certezza che non vi siano state pratiche scorrette;
- deve essere garantita un'adeguata sicurezza del sistema;
- il sistema deve essere semplice e di facile utilizzo.

3.3 Principi di sicurezza

Come emerge della sezione precedente, i requisiti per un'elezione democratica sono molto elevati. Il risultato delle elezioni deve essere corretto e non deve trapelare alcuna informazione sulla scelta di un elettore.

Non è banale decidere se un protocollo sia sicuro. Un grosso problema è quello di specificare il significato di "sicuro". Le prime nozioni di sicurezza erano definite come protezione da attacchi specifici. Un approccio simile è quello di definire la sicurezza come un insieme di proprietà di sicurezza specifiche.

Questa sezione presenta e fornisce le definizioni delle principali proprietà di sicurezza.

3.3.1 Correttezza

È essenziale che uno sistema di voto elettronico produca il conteggio corretto, che è la somma di tutti i voti espressi. Inoltre, è importante che i responsabili del controllo siano in grado di verificare che il conteggio sia corretto. Se il risultato non è il conteggio corretto, il processo di verifica deve indicarlo.

Per molti sistemi di voto è impossibile garantire che il conteggio sia perfettamente corretto, poiché spesso si presuppone che ogni elettore verifichi la corretta inclusione del proprio voto. Questo non risulta essere un problema finché l'errore non cambia il risultato delle elezioni. Per questi motivi è tollerabile che l'avversario possa modificare leggermente il conteggio, a patto che la possibilità di essere scoperti sia elevata per manipolazioni consistenti che cambiano il risultato delle elezioni.

Indipendenza del software

Rivest ha definito per la prima volta l'espressione "indipendenza del software" in [Riv08] come segue:

> Un sistema di voto è indipendente dal software se una modifica o un errore non individuato nel suo software non può causare una modifica o un errore non individuabile nel risultato delle elezioni.

Questa nozione è spesso considerata implicita nella correttezza o nella sicurezza *end-to-end*, come definito nella sezione successiva. L'indipendenza del software è una nozione utile per i sistemi di voto che utilizzano macchine per il voto ma non la crittografia, in quanto spesso non soddisfano la sicurezza *end-to-end* ma offrono comunque la possibilità di rilevare un errore.

Sicurezza *end-to-end*

Il requisito della verificabilità di uno sistema di voto elettronico è una parte importante del principio correttezza. La maggior parte dei sistemi di voto elettronico fornisce a ciascun elettore una sorta di ricevuta della sua scelta che gli consente di verificare che la scheda corrispondente sia stata correttamente inclusa nel conteggio finale. Questo è quello che si intende per **verifica individuale**. Si parla invece di **verifica universale** quando le autorità di voto pubblicano anche delle prove che convincano chi conduce le verifiche che il conteggio è effettivamente la somma di tutte le schede votate. Se il conteggio non è corretto, ci dovrebbe essere una alta probabilità che almeno una delle due fasi di verifica fallisca.

Il concetto che descrive i sistemi di voto che forniscono una verifica sia individuale che universale è la sicurezza *end-to-end*, spesso abbreviata in sicurezza o verificabilità E2E.

La nozione di verificabilità individuale è spesso suddivisa in due proprietà. Poiché l'elettore deve essere in grado di verificare che il voto è stato espresso come previsto, è necessario convincerlo che la ricevuta codifichi correttamente la sua scelta. Questo avviene tipicamente durante il processo di voto all'interno della cabina elettorale. A causa del requisito aggiuntivo di assenza di ricevuta e di resistenza alla coercizione, questa prova non deve essere trasferibile. Dopo la fase di voto, l'elettore deve essere in grado di verificare che la scheda sia stata conteggiata. A tal fine, spesso l'elettore deve verificare che una copia della sua ricevuta sia stata pubblicata dall'autorità di voto.

I sistemi di voto E2E riformulano la nozione di verificabilità in termini di tre fasi fondamentali:

- *Cast-as-intended*: l'elettore può verificare che il sistema di voto abbia segnato correttamente la sua scelta sulla scheda.
- *Recorded-as-cast*: l'elettore può verificare che il suo voto sia stato correttamente registrato dal sistema di voto.
- *Tallied-as-recorded*: l'elettore può verificare che il suo voto sia stato conteggiato come registrato.

Ciò si traduce come segue: l'elettore conferma innanzitutto che il sistema ha codificato correttamente il suo voto. Quindi traccia il suo voto sulla bacheca utilizzando la sua ricevuta e conferma che è stato registrato correttamente. L'integrità del risultato è garantita da una rigorosa verifica del processo di conteggio e dalla richiesta al sistema di pubblicare prove crittografiche del corretto funzionamento. Questa verifica in tre fasi copre quindi l'intero ciclo di vita del voto e l'elettore può essere sicuro che se c'è una manomissione o un guasto nel sistema sarà scoperto in uno dei controlli.

Queste due concezioni di verificabilità sono anche collegate: nella maggior parte dei sistemi di voto E2E, i controlli *cast-as-intended* e *recorded-as-cast* sono verificati dall'elettore, cioè insieme forniscono una verificabilità individuale, mentre la fase *tallied-as-recorded* può essere eseguita sia da elettori che da osservatori, cioè fornisce una verificabilità universale.

3.3.2 Segretezza del voto

Il requisito di segretezza già spiegato in una sezione precedente, si ripresenta qui sotto un aspetto più tecnico.

In passato, i protocolli di voto presenti nella letteratura di ricerca (come [CF85] [BY86] [Ben87] [Cha88]) mantenevano esplicitamente il legame tra elettore e scheda elettorale rilasciando all'elettore una ricevuta che gli consentiva di tenere traccia del suo voto su una bacheca pubblica. Nel 1994, Benaloh e Tuinstra [BT94] (e, indipendentemente, Niemi e Renvall [NR94]) hanno sottolineato che questa strategia permetteva all'elettore di dimostrare il proprio voto a terzi dopo le elezioni, facilitando così la compravendita di voti e la coercizione. Gli autori introdussero la nozione di "*receipt-freeness*" (assenza di ricevute), che si rivelò influente e molti sistemi di voto successivi (come [Cha04] [SK95] [Oka97]) rinunciarono completamente alle ricevute, concentrandosi invece sulla garanzia di trasparenza e integrità delle operazioni di conteggio. Tuttavia, nel 2004 Chaum [Cha04] ha reintrodotto l'uso delle ricevute, con l'importante distinzione che il contenuto della ricevuta viene mascherato crittograficamente, mantenendo così la privacy dell'elettore.

Juels, Catalano e Jakobsson [JCJ05] nel 2005 hanno sostenuto che un avversario può influenzare la scelta di un elettore senza essere esplicitamente a conoscenza del suo voto e hanno descritto tre modalità di coercizione: l'avversario può impedire all'elettore di votare, può appropriarsi delle sue credenziali di voto o costringerlo a votare per un candidato casuale. L'esatta differenza tra la assenza di ricevuta e la resistenza alla coercizione è sottile [DKR06]: nel primo caso, si presume che l'avversario si limiti a osservare le elezioni e a utilizzare le prove fornite da un elettore che collabora, nel secondo, l'avversario può creare voti specifici da far esprimere all'elettore o persino interagire con lui in qualche modo mentre vota.

Nel corso degli anni, quindi, i ricercatori hanno progressivamente affinato le nozioni di segretezza del voto, che sono state definite nelle seguenti proprietà chiave.

Voti nascosti da elettori nascosti

In un qualsiasi sistema di voto segreto, gli elettori devono comunicare privatamente le loro preferenze per lo scrutinio finale e degli scrutinatori sono responsabili di ricevere i voti e di condurre la fase di conteggio. In base al modo in cui gli elettori inviano i voti a questa autorità di conteggio, Sampigethaya e Poovendran[SP06] classificano ulteriormente i sistemi di voto elettronico in tre categorie.

- *Elettore nascosto*: Gli elettori inviano i voti in modo anonimo.
- *Voto nascosto*: I votanti inviano apertamente voti criptati.
- *Elettore nascosto con voto nascosto*: Gli elettori inviano voti criptati in modo anonimo.

Assenza di ricevute

L'espressione "assenza di ricevute" [BT94] [Oka97], [MN06] descrive il fatto che l'elettore non riceve alcuna prova che possa essere utilizzata per dimostrare a terzi come ha votato. I sistemi di voto crittografico spesso rilasciano ricevute per il processo di verifica individuale. Ciò non contraddice questa nozione se le ricevute non convincono terzi della scelta dell'elettore.

Resistenza alla coercizione

Un avversario che tenti di comprare un voto o di costringere un elettore può chiedere all'elettore di deviare dal processo di voto per ottenere una ricevuta con informazioni aggiuntive sulla scelta dell'elettore. Un sistema di voto che non fornisce alcuna informazione sulla scelta dell'elettore anche se quest'ultimo devia dal processo di voto previsto è detto resistente alla coercizione. Questa proprietà è stata definita per la prima volta in [JCJ05] come una garanzia di sicurezza più forte dell'assenza di ricevute.

3.3.3 Requisiti pratici

Oltre ai requisiti di sicurezza per la correttezza e la privacy di un'elezione, ci sono diversi requisiti pratici che sono importanti o almeno vantaggiosi per un sistema di voto.

Assenza di contestazioni

Un sistema elettorale verificabile garantisce che un elettore sia in grado di accorgersi se il suo voto non è stato incluso correttamente nel conteggio o se il conteggio non è corretto. Tuttavia, questo spesso non è sufficiente. Oltre a essere in grado di notare le manipolazioni, l'elettore dovrebbe essere in grado di convincere le autorità di voto che il conteggio è stato effettivamente manipolato.

Questa proprietà è spesso chiamata *dispute freeness*, in quanto un sistema di voto con questa proprietà permette all'elettore di provare un'asserzione valida. Ciò significa che se un sospetto non viene provato, non è valido. Di conseguenza, l'autorità di voto è in grado di risolvere tutte le controversie.

Un'ulteriore sfida è ottenere questa proprietà senza minacciare o indebolire la privacy dell'elettore. In molti casi l'elettore si accorge se il suo voto è stato alterato, ma deve svelare come ha votato per dimostrare la manipolazione. Un suggerimento comune per risolvere questo problema è l'uso di arbitri neutrali fidati che risolvono la controversia ma non riportano il contenuto del voto contestato. Pur essendo una soluzione insoddisfacente, in quanto richiede l'assunzione che esista un numero sufficiente di arbitri di questo tipo e che tutti gli arbitri siano affidabili per mantenere segreto il contenuto dei voti contestati.

Robustezza

Un'elezione non dovrebbe fallire se un singolo elettore corrotto devia dal protocollo o se un singolo membro dell'autorità di voto si rifiuta di partecipare al processo di scrutinio. Un sistema di voto che è in grado di tollerare un certo numero di parti che si discostano dal protocollo è detto robusto. Questa proprietà è importante per le elezioni del mondo reale, ma è difficile da formalizzare.

3.3.4 La conflittualità dei principi

Molte delle proprietà chiave descritte finora sono in conflitto tra loro in modo sottile, dando luogo a una serie di sfide tecniche e legali. Ad esempio, necessità di privacy del voto si scontra con quella la verificabilità. Se un elettore può dimostrare con successo il proprio voto a terzi al di fuori del seggio elettorale, potrebbe facilmente vendere il proprio voto o essere costretto a votare per un determinato candidato.

Analogamente, esiste una tensione tra verificabilità del voto e usabilità. Richiedere agli elettori di verificare il proprio voto ha un impatto negativo sull'usabilità, in quanto aggiunge ulteriori passaggi al processo, che possono confondere l'elettore. [Kar+11] [Ace+14] Inoltre, prove sperimentali hanno rilevato che solo una percentuale molto ridotta di elettori verifica effettivamente il proprio voto [Car+10] [MCE15], il che può compromettere in modo critico le garanzie di sicurezza di questi sistemi di voto.

L'accessibilità è anche in conflitto con la privacy del voto. L'adattamento dei sistemi di voto per fornire ausili audiovisivi o assistenza umana agli elettori disabili può creare situazioni in cui la scelta del candidato dell'elettore viene rivelata a terzi. Allo stesso modo, l'impiego di sistemi di voto a distanza, pur essendo significativamente più conveniente rispetto al voto di persona ai seggi elettorali, comporta un netto peggioramento della privacy del voto.[Ger+] A casa sua l'elettore non ha la garanzia della privacy ed è vulnerabile alla coercizione.

Di fronte a queste situazioni, i progettisti di sistemi di voto di solito incorporano nel sistema rimedi tecnologici o salvaguardie procedurali, danno priorità a certi requisiti di sicurezza rispetto ad altri o magari si concentrano sul soddisfacimento di certe proprietà in forma più debole.

4. Sistemi di e-voting

4.1 Bingo Voting

Nel 2007 i ricercatori Bohli, Muller-Quade e Rohrich[BMQR07] hanno proposto il sistema di voto Bingo Voting, chiamato così per via dell'utilizzo in esso di un generatore di numeri casuali, paragonabile all'estrazione di numeri per il gioco del bingo. È attribuibile alla categoria dei sistemi per il voto elettronico *stand-alone* nei seggi elettorali. Gli obiettivi che questo sistema si prefigge di raggiungere sono:

- la garanzia dell'espressione di voto e della sua verificabilità, cioè la possibilità di ogni elettore di controllare se il proprio voto è stato espresso e contato correttamente senza esporne il contenuto.
- la prevenzione di coercizione, anche se l'elettore fosse oggetto di pressioni esterne o di voto di scambio, non otterrebbe alcuna prova che gli consenta di dimostrare il contenuto del suo voto.

Per garantire la correttezza del suo voto, l'elettore deve solo verificare l'uguaglianza di due numeri generati casualmente e controllare che la sua ricevuta cartacea sia stata affissa in una bacheca. Il sistema di voto rimane sicuro anche se non tutti gli elettori verificano effettivamente la correttezza del processo, purché l'avversario non possa prevedere quale elettore verificherà effettivamente.

Bingo Voting è stato utilizzato nelle elezioni del parlamento studentesco dell'Università di Karlsruhe, in Germania, nel 2008.[Bär+08] Un altro studio esamina le ramificazioni delle macchine da voto corrotte sul Bingo Voting e prescrive soluzioni che possono essere generalizzate per altri sistemi di voto.[Boh+09] Un altro studio ancora [Liu+11] nota che l'affidamento a un generatore di numeri casuali rappresenta una potenziale vulnerabilità per la sicurezza e propongono una soluzione alternativa. Henrich[Hen12] ha analizzato la fattibilità dell'impiego del Bingo Voting per elezioni del mondo reale negli Stati Uniti, in Germania e in India.

4.1.1 Descrizione del sistema di voto

Si assume lo scenario di un'elezione con l candidati e n elettori idonei. Nella cabina elettorale si trovano un generatore di numeri casuali (*random number generator* RNG) affidabile e la macchina per votare.

Fase di pre-voto

Nella fase di preparazione delle elezioni, gli organizzatori dell'elezione generano per m elettori e n candidati $m \cdot n$ stringhe di numeri casuali, raggruppati in modo che a ogni

candidato sia assegnata una serie di n numeri stringhe di numeri casuali. Ognuno di questi "voti fittizi" viene poi criptato utilizzando un "*commitment scheme*"[1]. L'elenco di tutti i vincoli (voti fittizi criptati) viene quindi pubblicato insieme a una prova che i voti fittizi sono distribuiti equamente a tutti i candidati.

È possibile definire inoltre ulteriori "candidati" per rendere possibile la scelta di scheda bianca.

Fase di voto

Prima del processo di voto vero e proprio, l'autorità di voto deve assicurarsi che l'elettore sia effettivamente idoneo a votare.

Oltre alla macchina per votare (compreso il dispositivo di input), l'elettore trova nella cabina elettorale altri due elementi: una stampante e un generatore di numeri casuali affidabile con un display, collegato alla macchina per votare e inattivo prima che l'elettore abbia espresso il proprio voto.

Nella cabina elettorale, un elettore deve eseguire le seguenti operazioni per esprimere un voto:

1. L'elettore deve indicare il proprio voto premendo il pulsante del candidato corrispondente sulla macchina per votare.

2. Il generatore di numeri casuali produce quindi un nuovo numero casuale R.

3. Il numero casuale R viene trasferito alla macchina per votare e assegnato al candidato scelto dall'elettore. Per ogni altro candidato, la macchina per votare estrarrà a caso un numero dal pool di voti fittizi per il rispettivo candidato. La macchina stampa una ricevuta con l'elenco dei candidati e dei numeri: al candidato che è stato votato viene assegnato il nuovo numero casuale R, mentre per gli altri candidati viene indicato il voto fittizio rispettivamente scelto.

4. L'elettore deve verificare che il numero indicato sul generatore di numeri casuali sia assegnato al candidato per cui intendeva votare. In caso contrario, l'elettore deve contestare immediatamente il voto.

5. L'elettore esce dalla cabina e ritira la ricevuta. Per qualsiasi estraneo è impossibile riconoscere il numero casuale assegnato e quindi il voto che la scheda implica.

[1] Un commitment scheme è una primitiva crittografica che consente di vincolarsi a un valore scelto rendendolo segreto, con la possibilità di rivelare il valore vincolato in un secondo momento. I commitment scheme sono progettati in modo che una parte non possa cambiare il valore dopo averla vincolata.

È possibile immaginare un commitment scheme come un mittente che mette un messaggio in una scatola chiusa a chiave e la consegna a un destinatario. Il messaggio nella scatola è nascosto al destinatario, che non può aprire la serratura. Dal momento che il destinatario possiede la scatola, il messaggio al suo interno non può essere alterato, ma solo rivelato se il mittente decide di dargli la chiave in un momento successivo.

Le interazioni in uno commitment scheme si svolgono in due fasi:

- la fase di *commit*, durante la quale si sceglie e si vincola un valore
- la fase di *reveal*, durante la quale il valore viene rivelato dal mittente e il destinatario ne verifica l'autenticità.[Gol01]

Fase post voto

Dopo l'elezione, la macchina per il voto calcola il risultato e lo invia insieme a una prova di correttezza a una bacheca pubblica. I dati pubblicati sono composti da quattro sezioni:

- il risultato finale dello scrutinio;
- un elenco di tutte le ricevute emesse;
- un elenco di tutte le coppie di voti fittizi/candidati non utilizzati con le rispettive informazioni di *commit* e *reveal*;
- prove non interattive per la correttezza, cioè che ogni scheda votata contenga esattamente un voto non fittizio e che ogni voto fittizio non rivelato sia stato utilizzato in una sola scheda.

Con queste informazioni ogni elettore può facilmente verificare che la propria ricevuta sia inclusa nell'elenco e quindi è stata inclusa nel conteggio.

Il numero di voti fittizi non utilizzati per ciascun candidato corrisponde direttamente al numero di voti ricevuti dal candidato durante la fase di voto (e quindi al conteggio).

Figura 4.1: Rappresentazione grafica del voto di Bingo Voting.

4.1.2 Verifica

Una caratteristica molto importante dei sistemi di voto crittografici è che ogni elettore è in grado di verificare che il conteggio dei voti sia corretto. La verifica di Bingo Voting segue il principio di sicurezza end-to-end discusso nella Sezione 3.3.1, dividendosi quindi in verifica individuale e verifica universale.

Per la verifica di un'elezione che utilizza Bingo Voting, l'autorità di voto deve dimostrare che:

- Ogni candidato aveva lo stesso numero di voti fittizi prima della fase di voto.
- Per ogni elettore che ha votato, ogni candidato che non ha ricevuto il suo voto ha perso un voto fittizio.

Questi due elementi fanno parte del processo di verifica individuale e universale.

Verifica individuale

La verifica individuale di Bingo Voting consiste in due fasi. Durante il processo di voto, l'elettore verifica che la ricevuta codifichi correttamente la sua scelta. A tal fine, l'elettore confronta la ricevuta con il display del generatore di numeri casuali e verifica che il numero casuale accanto al candidato per cui ha votato sia lo stesso visualizzato

sul display (cfr. Figura 4.1). Questa operazione viene eseguita all'interno della cabina elettorale direttamente dopo che l'elettore ha espresso il proprio voto. Questa è anche l'unica operazione che l'elettore è tenuto a fare da solo. Se il nuovo numero casuale è associato correttamente al candidato per cui l'elettore intendeva votare, la macchina per votare ha ricevuto e registrato correttamente il voto. Si può considerare il voto come *cast-as-intended*. Questa fase è un semplice confronto tra due numeri casuali ed è l'unica che richiede la conoscenza della scelta rappresentata dalla ricevuta.

Dopo la fase di voto, l'elettore controlla che la sua ricevuta sia stata pubblicata correttamente. Se la ricevuta emessa durante il processo di voto è identica a quella pubblicata successivamente dall'autorità di voto, il voto è stato incluso correttamente nel conteggio. Qui si può considerare il voto come *counted-as-cast*.

La seconda fase, che assicura che certifica il voto ocome *tallied-as-recorded*, può essere delegata dall'elettore consegnando la propria ricevuta (o solo una copia della ricevuta) a uno o più revisori. Questi possono essere familiari, rappresentanti di partiti politici o revisori esterni. Per assicurarsi che il voto corrispondente a una certa ricevuta sia stato inserito correttamente nel conteggio è sufficiente verificare che la ricevuta pubblicata sia identica alla copia cartacea rilasciata durante la fase di voto. Non è necessario sapere per quale candidato è stato espresso il voto.

Sebbene non sia necessario sapere quale voto codifichi la ricevuta per verificare la corretta pubblicazione da parte dell'autorità di voto, è necessario possedere la ricevuta (o almeno una copia identica) poiché l'ipotesi che un elettore umano sia in grado di ricordare il contenuto completo della ricevuta non è realistica. Improbabile l'ipotesi che venga verificata la corretta pubblicazione di ogni ricevuta. È inoltre inutile se l'avversario non è in grado di prevedere quali ricevute vengono controllate e se ne viene controllata una percentuale significativa. In questo caso, l'avversario rischia di essere scoperto per ogni voto alterato, rendendo improbabile la manipolazione non rilevata di un gran numero di voti.

Verifica universale

Bingo Voting utilizza voti fittizi per tracciare internamente i voti espressi e produrre un conteggio verificabile. All'inizio ogni candidato ha lo stesso numero di voti fittizi e durante il processo di voto di ogni elettore ogni candidato, tranne quello per cui ha votato, perde un voto fittizio. Pertanto, il numero di voti fittizi rimanenti si traduce direttamente nel conteggio. La verifica individuale convince l'elettore che il candidato per cui ha votato non ha perso un voto fittizio e che il voto è stato incluso nel conteggio. Per la verifica globale, l'autorità di voto deve dimostrare quattro cose:

- Ogni candidato aveva lo stesso numero di voti fittizi all'inizio dell'elezione.
- Ogni ricevuta è ben formata e non contiene più di un voto fittizio per ogni candidato. Ciò equivale a dimostrare che ogni candidato ha perso al massimo un voto fittizio per ogni elettore.
- Ogni voto fittizio è stato utilizzato, cioè scritto su una ricevuta, oppure non è stato utilizzato, cioè aperto e pubblicato come parte del conteggio.
- Ci sono solo poche coincidenze statisticamente insignificanti, cioè numeri casuali (fittizi) identici per candidati diversi.

Per verificare questi quattro elementi non è necessario essere un elettore o aver partecipato alle elezioni. Sono sufficienti i dati pubblici pubblicati dall'autorità di voto.

La prima prova che dimostra che ogni candidato ha ricevuto lo stesso numero di voti fittizi prima della fase di voto è una semplice prova di un rimescolamento di contenuto noto per un insieme di vincoli. L'autorità di voto agisce come *prover* P, l'input pubblico è l'elenco di tutti i vincoli nei confronti dei candidati che fanno parte dei voti fittizi e la lista M contenente ogni candidato l volte con l come numero di elettori. La prova utilizza un controllo parziale randomizzato per dimostrare che il contenuto dei vincoli è identico a M.

La seconda prova pubblicata dall'autorità di voto dimostra che ogni ricevuta è ben formata. A tal fine, l'autorità di voto prende per ogni ricevuta ogni voto fittizio utilizzato insieme a una nuova coppia di vincoli. Questa nuova coppia contiene il numero casuale generato dal generatore di numeri casuali durante il processo di voto e il candidato per cui l'elettore ha votato. L'autorità di voto dimostra quindi che il contenuto di questi vincoli (i voti fittizi e il nuovo vincolo) è identico al contenuto della ricevuta. Utilizzando queste coppie di vincoli, l'autorità di voto dimostra che tutti i numeri casuali della ricevuta, tranne uno, sono numeri casuali fittizi e che ogni numero casuale fittizio è associato al candidato corretto.

La terza prova è la dimostrazione che ogni voto fittizio è stato utilizzato su una ricevuta o è rimasto inutilizzato e ha contribuito al conteggio. Questo è in realtà abbastanza facile da verificare, poiché ogni voto fittizio pubblicato durante la fase di preparazione deve essere utilizzato per dimostrare che una ricevuta è ben formata o reso pubblico se non è stato utilizzato su una ricevuta. È anche facile verificare che ogni voto fittizio sia stato usato al massimo su una ricevuta e non usato su una ricevuta e segnato come inutilizzato.

L'ultima prova è verificare che non ci siano state più coincidenze di quelle previste se tutti i numeri casuali sono stati estratti uniformemente dall'insieme di tutti i numeri casuali possibili. Ciò è facile da verificare perché tutti i numeri casuali sono pubblici, sia come parte di una ricevuta sia come numeri casuali fittizi pubblici da voti fittizi non utilizzati. Fondamentalmente una macchina per votare manipolata può essere in grado di cambiare un singolo voto quando si verifica una coincidenza. La descrizione originale di Bingo Voting sostiene che le coincidenze sono trascurabili in quanto sono efficacemente prevenute dalla scelta di numeri casuali più lunghi.[Boh+09] Sfortunatamente, numeri casuali più lunghi rendono più impegnativo per l'elettore il confronto della ricevuta con il generatore di numeri casuali nella cabina elettorale.

Si noti che la verifica dell'idoneità di ciascun elettore e del fatto che ogni elettore esprima al massimo un voto non fa parte di Bingo Voting. Sebbene sia una parte integrante della correttezza verificabile, non fa parte della maggior parte degli sistemi di voto crittografici ed è solitamente garantita da autorità fisiche tramite le procedure elettorali consuete.

4.2 Helios

Helios è un sistema di voto elettronico *open source* distribuito al pubblico con licenza GPL v3, appartenente alla categoria dei sistemi di voto a distanza. Una delle caratteristiche principali di questa applicazione è la sua implementazione come una single web-page application utilizzando JavaScript. In particolare, tutti i dati necessari sono precaricati nella memoria del browser e il codice JavaScript aggiorna l'interfaccia utente

HTML renderizzata quando vengono cliccati collegamenti o pulsanti. Di conseguenza, non è necessario alcun accesso a Internet dal momento in cui i dati vengono caricati sul browser web fino a quando si è pronti a esprimere il proprio voto e necessitando solamente di un browser web moderno, è possibile preparare un'elezione, invitare gli elettori a esprimere un voto segreto, eseguire lo scrutinio e generare una prova di validità per l'intero processo.

Helios è, a seconda del suo sviluppatore,[Adi08] volutamente più semplice della maggior parte dei sistemi di voto elettronico, il suo obiettivo non è infatti quello di risolvere il problema della resistenza alla coercizione. Cerca piuttosto di offrire uno strumento a una serie di contesti più piccoli (scuole, università, club locali, gruppi online) che non soffrano dello stesso rischio di coercizione delle elezioni governative ad alto rischio e che tuttavia necessitano della segretezza del voto e di risultati elettorali affidabili, proprietà che non possono ottenere senza un'elezione di persona, fisicamente osservabile e ben orchestrata.

Il voto e la sua preparazione, compresa la verificabilità individuale, si basano sul Simple Verifiable Voting Protocol di Benaloh[Ben06], che si basa su due aspetti: la separazione tra la preparazione/crittografia del voto e il deposito del voto e la "challenge di Benaloh". L'idea della separazione comporta che la scheda possa essere visualizzata, possa essere espresso un voto, la scheda possa essere crittografata e la crittografia può essere verificata senza doversi autenticare e, quindi, senza essere un elettore idoneo. L'elettore deve essere autenticato solo per il voto finale. Un vantaggio di questo approccio è che tutti (compresi gli osservatori elettorali) possono verificare il meccanismo di preparazione della scheda. Nella challenge di Benaloh, il sistema si impegna a registrare il voto criptato e poi gli elettori possono decidere se verificare o esprimere il voto. Il software non può falsificare le informazioni criptando il candidato sbagliato, poiché non sa se l'elettore verificherà o esprimerà il voto criptato. Gli elettori si accorgeranno durante la verifica se il nome del candidato sbagliato è stato criptato. Per garantire che il software fornisca lo stesso testo cifrato per la verificabilità e l'espressione del voto, il software si impegna a cifrare il voto visualizzando un valore hash del testo cifrato, che è lo *smart ballot tracker* del voto. I requisiti di privacy rendono impossibile l'espressione di un voto verificato. Il voto cifrato verificato deve quindi essere nuovamente cifrato. Di conseguenza, viene calcolato e visualizzato un nuovo valore hash. In questo modo, l'elettore non può verificare il voto criptato che ha infine espresso, ma deve fidarsi del sistema grazie alle verifiche precedenti.

Helios si è dimostrato molto influente e viene ancora mantenuto attivamente. Helios è stato oggetto di numerosi studi di usabilità [WH09] [Kar+11] [Ace+14]. Sono state scoperte diverse vulnerabilità nell'implementazione di Helios [ED10] [Hei+11] [BPW12]. Ad esempio, gli avversari possono copiare e rielaborare i voti precedentemente espressi [CS10] e un sistema corrotto può distribuire ricevute identiche a diversi elettori, modificando i loro voti effettivi senza essere individuato [KTV12]. Molti di questi problemi sono stati risolti nelle versioni successive di Helios. Sono stati proposti anche dei miglioramenti: Helios è stato modificato per consentire nuove proprietà come l'autodichiarazione [DL15] e garanzie di privacy più forti [BPW12][DVDGA12].

Helios è stato utilizzato in diverse elezioni vincolanti: l'Université Catholique de Louvain ha utilizzato Helios nel 2009 per eleggere il Presidente dell'Università in un'elezione con circa 5000 elettori registrati [Adi+09]. Helios è stato utilizzato anche per le elezioni studentesche a Princeton [Mor16] e per le elezioni interne dell'International Association for Cryptologic Research (IACR) [Iac]. Helios è stato anche adattato per altri sistemi di voto, come il voto singolo trasferibile (STV). Ne sono un esempio Zeus

[Lou+14], e un'altra variante sviluppata da Bulens Giry e Pereira [BGP11].

4.2.1 Descrizione del sistema

Creazione di un'elezione

Un'istanza pubblica di Helios è ospitata su https://vote.heliosvoting.org/, da cui chiunque può creare e gestire delle elezioni. Questa pagina contiene anche un link al repository del codice di Helios, attualmente ospitato su Github, che consente a chiunque di ospitare un server Helios personale. Il codice è rilasciato sotto licenza Apache 2.0.

La registrazione è gestita come in gran parte dei siti web:

- l'utente inserisce un indirizzo e-mail, un nome e la password desiderata.
- all'indirizzo e-mail indicato viene inviata un'e-mail con un link di conferma incorporato.
- l'utente clicca sul link di conferma per attivare il suo account.

Un utente registrato crea quindi un'elezione con un nome, una data e un'ora in cui si prevede di iniziare a votare e una data e un'ora in cui si prevede di terminare le votazioni. Al momento della creazione, Helios genera e memorizza una nuova coppia di chiavi El-Gamal per l'elezione. Solo la chiave pubblica è disponibile per l'utente registrato: Helios mantiene segreta la chiave privata. L'utente che ha creato l'elezione è considerato l'*amministratore*.

L'elezione si trova quindi in "modalità di impostazione", dove la scheda elettorale può essere preparata, rivista e modificata dall'amministratore. L'utente può accedere nuovamente per più giorni per modificare qualsiasi aspetto della scheda elettorale.

L'amministratore può aggiungere, aggiornare e rimuovere gli elettori a piacimento. Un elettore è identificato da un nome e da un indirizzo e-mail ed è specifico per una determinata elezione. Helios genera automaticamente una password casuale di 10 caratteri per ogni elettore. In qualsiasi momento, l'amministratore può inviare un'e-mail agli elettori utilizzando l'interfaccia amministrativa di Helios. Queste e-mail conterranno automaticamente la password dell'elettore, nonostante ciò l'amministratore non potrà vederla in nessun momento.

Quando è pronto, l'amministratore "congela" l'elezione; a questo punto l'elenco degli elettori, le date di inizio e fine dell'elezione e i dettagli della scheda diventano immutabili e disponibili per il download in formato JSON. L'amministratore riceve quindi un'e-mail da Helios con l'hash SHA1 di questo oggetto JSON. A questo punto l'elezione è pronta per essere votata dagli elettori. L'amministratore di solito invia un'e-mail agli elettori utilizzando l'interfaccia amministrativa di Helios per informarli che le urne sono aperte. Una descrizione dettagliata sulla procedura di creazione dell'elezione è presente nella Sezione 5.2.

Procedura di voto

Per iniziare l'operazione di voto, viene inviata all'elettore un'e-mail di invito contenente la descrizione dell'elezione, il suo nome utente (che può essere ad esempio il suo indirizzo e-mail), la sua password specifica per l'elezione, l'hash SHA1 dei parametri elettorali

e l'URL che la indirizza alla cabina elettorale Helios. È importante notare che questo URL non contiene alcuna informazione identificativa: identifica solo l'elezione.

Sono possibili diverse varianti di questo processo:

- Helios può utilizzare servizi di autenticazione esterni, come quelli forniti da Google, Facebook o GitHub (naturalmente, queste credenziali esterne non sono mai accessibili al server Helios) o, in alcuni casi, servizi di autenticazione personalizzati come un servizio SSO o LDAP dell'università. In questo caso, l'invito al voto non conterrà credenziali.

- Helios può aggiungere un alias di elettore nell'invito al voto. Sebbene l'uso di un ID elettore esplicito sia un modo per migliorare il controllo democratico di chi ha presentato un voto in un'elezione, molte organizzazioni ritengono che il loro elenco di membri sia privato e non vogliono che un'elezione sia un'occasione per pubblicare indirettamente un elenco di membri. In altri casi, anche se l'elenco dei membri può essere reso disponibile agli elettori, il voto è considerato privato, per ovvi motivi di coercizione. In queste situazioni, il server Helios consente di mantenere privato l'ID dell'elettore e di pubblicare solo gli alias anonimi: questo è sufficiente per consentire a un elettore di tenere traccia della propria scheda, ma non di verificare chi ha inviato le altre schede incluse nel conteggio. Questo approccio viene talvolta adottato anche come linea di difesa nel caso di un guasto catastrofico che porterebbe alla decrittazione delle singole schede, o semplicemente della naturale evoluzione della potenza di calcolo che renderebbe possibile la violazione della crittografia: avere schede associate a pseudonimi anonimi e non a identificativi reali può impedire a un avversario di determinare chi ha decrittato la scheda.

Utilizzando il collegamento alla cabina elettorale, gli elettori possono inviare una scheda al server elettorale. Helios ha un'API aperta, il che significa che chiunque (elettore, candidato o attivista) potrebbe programmare un proprio *Ballot Preparing System* (BPS) che può essere utilizzato per presentare le schede in qualsiasi elezione. Tuttavia, la maggior parte degli elettori utilizza il BPS fornito dal sito web di Helios.

Il BPS di Helios consiste in una *Single-Page Web Application*: non appena la cabina elettorale viene caricata in un browser (recente), un elettore può andare *offline*, fare le sue scelte, farle criptare, cancellare tutte le scelte in chiaro e la casualità associata, e tornare *online* solo per inviare la scheda criptata. Il voto con il BPS di Helios si svolge in 3 fasi:

1. L'elettore seleziona le risposte alle diverse quesiti visualizzate. Una scheda può contenere un numero qualsiasi di quesiti. Una scheda può contenere un numero qualsiasi di quesiti, a ciascuna delle quali è possibile rispondere facendo una serie di scelte determinate dalle regole elettorali, ad esempio una sola scelta, un numero qualsiasi di scelte o un numero compreso in un intervallo specifico, ad esempio da 0 al numero di seggi disponibili per l'elezione.

2. Quando l'elettore ha completato le sue scelte, è invitato a rivederle e ad apportare le modifiche che desidera. Nel frattempo, la scheda è stata criptata e un *tracker* della scheda è a disposizione dell'elettore. Questo *tracker* identifica in modo univoco la scheda criptata, preservando la segretezza del voto, e consente all'elettore di contestare il BPS e, successivamente, di verificare che la sua scheda

criptata sia stata correttamente registrata dal server di voto e inclusa nel conteggio. L'elettore è invitato a memorizzare o stampare immediatamente questo *tracker*.

3. L'elettore può quindi decidere di inviare la propria scheda al server di voto Helios, che gli chiederà di fornire le proprie credenziali e di inviare la scheda. Richiedere l'autenticazione dell'elettore proprio alla fine del processo di voto presenta notevoli vantaggi. Permette a chiunque di visualizzare il formato della scheda e il processo di preparazione della stessa, eventualmente in collaborazione con altre persone in caso di dubbi, senza il timore che le proprie credenziali vengano rubate. Inoltre, rende più difficile per un server Helios corrotto servire una versione manomessa della BPS in funzione delle credenziali dell'elettore, poiché l'identità dell'elettore è sconosciuta quando la BPS viene servita.

All'elettore vengono poi presentate due diverse opzioni: inviare semplicemente la scheda al server Helios o verificare che la scheda sia stata criptata correttamente. Ogni elettore può ripetere l'intera procedura di preparazione e invio della scheda un numero qualsiasi di volte: verrà presa in considerazione solo l'ultima scheda ricevuta, che l'elettore potrà verificare grazie al proprio tracker della scheda.

Verifica

La verifica di un'elezione Helios comprende tre modalità:

1. La verifica *cast-as-intended* consente a qualsiasi elettore di ottenere la certezza che la scheda da lui consegnata catturi la sua intenzione di voto.

2. La verifica *recorded-as-cast* consente a qualsiasi elettore di ottenere la certezza che la sua scheda sia stata registrata correttamente sul server Helios.

3. La verifica *tallied-as-recorded* consente a chiunque di verificare che tutti i voti validi registrati siano inclusi nel conteggio.

Insieme, queste fasi di verifica forniscono la verificabilità end-to-end.

Verifica *cast-as-intended* Il tracker della scheda che viene mostrato all'elettore prima della presentazione della scheda dovrebbe essere un'impronta fedele dell'intento dell'elettore. Tuttavia, un elettore può legittimamente mettere in dubbio questo dato e voler scoprire se la scheda cattura davvero la sua intenzione.

Helios offre la possibilità di sfidare il sistema di preparazione della scheda attraverso la challenge di Benaloh. L'ingrediente chiave della verifica risiede nel momento in cui il tracker della scheda viene visualizzato dall'elettore, ossia dopo il completamento di tutte le attività di preparazione della scheda, ma prima che l'elettore si autentichi per inviare la sua scheda. Ciò significa che, quando il sistema di preparazione della scheda visualizza il tracker della scheda, non sa se questa scheda è destinata a essere inviata al server Helios o se la preparazione della scheda fa solo parte di una verifica. La challenge di Benaloh procede offrendo all'elettore due opzioni: autenticare e inviare la scheda, oppure richiedere una verifica della scheda e verificare la crittografia. Se viene scelta la seconda opzione, la scheda viene inviata al server Helios.

Se si sceglie la seconda opzione, il sistema di preparazione della scheda è tenuto a fornire tutti i dati utilizzati per preparare la scheda, compresa la casualità utilizzata

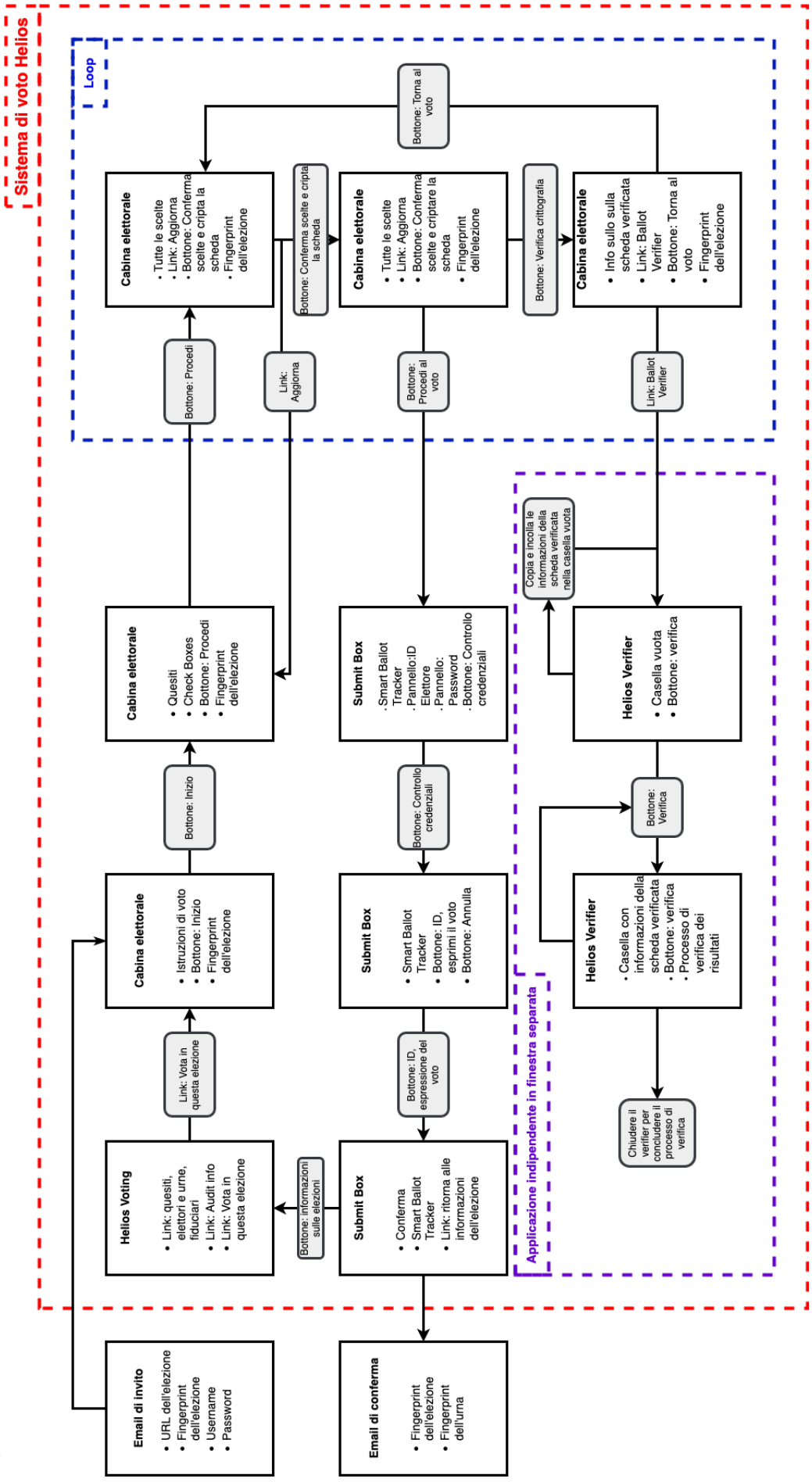


Figura 4.2: Procedura di voto e verifica delle schede di Helios

per la crittografia. Sulla base di questi dati, l'elettore può ora utilizzare il software di verifica della scheda per verificare che questa corrisponda alla sua intenzione di voto, sia valida e corrisponda al tracker fornito in precedenza. Una scheda sottoposta a verifica non può più essere presentata, poiché l'accesso ai dati di verifica potrebbe violare la privacy del voto. L'obiettivo è verificare se il sistema di preparazione delle schede si comporta onestamente su un dispositivo specifico, e non verificare se la scheda specifica che un elettore vuole presentare è stata preparata correttamente.

Verifica *recorded-as-cast* L'elettore può compierla collegandosi alla pagina web di Helios Ballot Tracking Center, che visualizza i tracker delle schede di tutti i voti destinati allo scrutinio, e verificando se la sua scheda è effettivamente visualizzata lì, con il numero di tracking associato.

Così come l'accesso al sistema di preparazione delle schede, anche l'accesso al Ballot Tracking Center non richiede alcuna autenticazione. Questo rende più difficile per un eventuale server Helios corrotto adattare l'elenco delle schede a seconda dell'elettore che accede al centro di tracciamento. Inoltre, ciò facilita la delega di questa verifica: un elettore può inviare il suo numero di tracciamento a degli attivisti, o anche diffonderlo sui social network per farlo verificare ad altri. Tuttavia, la verifica della presenza della scheda in un determinato momento delle elezioni non garantisce la sua permanenza, non modificata, e che venga contata durante lo scrutinio. Pertanto, è utile che gli elettori ispezionino il Ballot Tracking Center quando c'è un accordo pubblico sulle schede che saranno utilizzate nel conteggio, cioè quando l'hash del conteggio criptato è reso disponibile ai fiduciari prima della decrittazione.

Verifica *tallied-as-recorded* A differenza dei precedenti processi di verifica, in gran parte individuali, la verifica *tallied-as-recorded* è universale, nel senso che ogni elettore si preoccupa non solo del corretto conteggio del proprio voto, ma anche della validità dei restanti.

Il punto di partenza del processo di verifica *tallied-as-recorded* è il Ballot Tracking Center: da lì chiunque può raccogliere l'elenco delle persone che hanno presentato le schede e i relativi numeri dei tracker. A meno che gli amministratori delle elezioni non decidano di offuscare i nomi degli elettori utilizzando degli alias, questo elenco fornisce le informazioni necessarie per verificare che le schede siano state presentate da elettori reali. In caso di dubbi, è anche possibile contattare gli elettori di persona, e chiedere loro di confermare il loro tracker.

Una volta confermato l'elenco dei tracker, la seconda fase del processo di verifica consiste nello scaricare l'elenco completo delle schede dal server Helios e verificare se tutte le schede corrispondono ai numeri di tracciamento previsti. Sulla base di queste schede, la validità di tutti i voti può essere verificata ispezionando le prove che contengono. Infine, si può controllare anche la correttezza della decrittazione del computo, verificando le prove di decrittazione fornite dai fiduciari. Tutti questi passaggi si basano pesantemente su tecniche crittografiche, riportate nella documentazione online di Helios, rendendo possibile a qualsiasi programmatore implementare un sistema di verifica.

Questo processo di verifica è molto più impegnativo delle altre fasi di verifica: richiede certamente una potenza di calcolo molto maggiore di quella normalmente disponibile in un browser per qualsiasi elezione di dimensioni ragionevoli. Tuttavia, questa fase

di verifica è anche quella che può essere più naturalmente delegata a terzi (attivisti, candidati), poiché si concentra su una proprietà globale dell'elezione.

4.3 Prêt à Voter

Prêt à Voter è una famiglia di sistemi di voto supervisionato verificabile end-to-end ideato da Peter Ryan dell'Università di Lussemburgo che si pone l'obiettivo di fornire garanzie di accuratezza del conteggio e di privacy dell'elettore, garantendo la massima trasparenza del processo, compatibilmente con il rispetto della segretezza del voto.[Rya+10] Prêt à Voter fornisce un approccio pratico alle elezioni verificabili end-to-end con un'esperienza di voto semplice e familiare. Assicura un elevato grado di trasparenza, preservando al contempo la segretezza della scheda elettorale. La garanzia deriva dall'affidabilità di una traccia di prove creata dall'esecuzione dell'elezione, piuttosto che dalla necessità di riporre fiducia nei componenti del sistema. L'idea originale ha subito numerosi miglioramenti dal suo inizio nel 2004, guidati dall'identificazione delle minacce, dalla disponibilità di primitive crittografiche migliorate e dal desiderio di rendere il sistema il più flessibile possibile. Questa evoluzione è culminata nello sviluppo e nell'implementazione del sistema per l'utilizzo nello Stato di Victoria in Australia per le elezioni statali del novembre 2014.

4.3.1 Descrizione del sistema

Il funzionamento del sistema Prêt à Voter è articolato in quattro parti ben distinte: generazione della scheda, acquisizione del voto, elaborazione del voto e verifica.

Generazione della scheda

Una scheda elettorale cartacea Prêt à Voter contiene un elenco rimovibile di nomi dei candidati (di solito la metà sinistra del foglio), dati in ordine casuale, e le corrispondenti caselle in cui segnare la preferenza dell'elettore (la metà destra). Questa metà destra contiene le aree di marcatura per ogni candidato e una stringa chiamata *onion* che contiene le informazioni sulla permutazione in forma criptata e consente al sistema di ricostruire l'ordine dei candidati, in modo tale che nessuna parte sia singolarmente in grado di eseguire la decrittazione. L'ordinamento casuale dei candidati è ciò che garantisce la privacy degli elettori. Il lato sinistro verrà staccato e distrutto prima della scansione del voto, mentre l'elettore conserverà il lato destro (l'originale o una copia) come ricevuta. A condizione che nessuno, tranne l'elettore, conosca l'ordine e che il collegamento tra l'elettore e questa particolare scheda venga perso prima che il voto venga decriptato, nessun altro (compreso lo scanner nella cabina) potrà mai conoscere la preferenza dell'elettore.

Acquisizione del voto

L'acquisizione del voto consiste semplicemente nella lettura digitale del lato destro della scheda elettorale e l'invio al database contenente i voti. L'elettore conserva la parte destra della scheda come ricevuta. Lo scanner nella cabina elettorale contrassegna inoltre la ricevuta come autentica. Il voto criptato (cioè la parte destra) viene pubblicato in una bacheca in modo che chiunque sia in possesso della ricevuta possa verificare che appaia inalterato.

Elaborazione del voto

L'idea dell'elaborazione del voto di Prêt à Voter è quella di trasformare l'insieme dei voti criptati in un insieme di voti non criptati, senza che nessuno (comprese le parti che partecipano alla decrittazione) ricolleghi i voti agli elettori.

Per fare in modo che la fonte di un voto criptato venga anonimizzata, nei sistemi di voto viene solitamente utilizzata una *mixnet*[2] garantendo al contempo che la fonte sia valida e che il voto non sia stato modificato. In generale, un intero insieme di voti criptati viene passato tra una serie di server mix e rimescolato una o più volte. In ogni fase l'insieme di voti criptati viene fatto apparire diverso per nascondere il rimescolamento segreto.

Per verificare una *mixnet* è utile impiegare la pratica del Randomised Partial Checking[JJR02]. In questa pratica, ogni livello della *mixnet* esegue il proprio rimescolamento segreto e la trasformazione dei voti e passa il risultato al livello successivo. L'insieme dei voti viene pubblicato ad ogni fase di questo processo. Per verificare questa pratica, un sottoinsieme casuale di voti criptati viene selezionato pubblicamente. La *mixnet* è quindi obbligata a pubblicare la prova della provenienza di questi voti. I voti da sottoporre a verifica devono essere scelti in modo da non esporre il percorso completo dal voto registrato al voto decriptato, per preservare l'anonimato degli elettori. Le probabilità di riscontro di brogli nella *mixnet* aumentano esponenzialmente con il numero di voti contraffatti.

Verifica

Affinché gli elettori, gli scrutatori e i candidati siano certi che i risultati pubblicati corrispondano ai voti espressi, devono poter verificare che le fasi di mixing, decodifica e conteggio siano state eseguite correttamente. Per raggiungere questo obiettivo, ogni operazione protetta crittograficamente pubblica informazioni sufficienti perché se ne possa verificarne la correttezza. Queste possono consistere in una dimostrazione a conoscenza zero[3]; possono consistere in informazioni che consentano a chiunque di invertire le operazioni eseguite e confermare che l'output può essere ritrasformato nell'input. In Prêt à Voter, la fase di decrittazione porta alla pubblicazione di tutti i voti (anonimizzati) non criptati; in questi casi, l'operazione di conteggio può essere verificata pubblicamente senza ulteriori informazioni.

La verificabilità end-to-end del sistema è data dal fatto che gli elettori possono verificare che le loro espressioni di voto sono state correttamente registrate e gli organi di controllo possono verificare che i voti nel loro complesso sono stati conteggiati in maniera regolare.

[2] Concetto introdotto da David Chaum[Cha81] nel 1981, le *mix networks* sono protocolli di routing che creano comunicazioni difficili da tracciare utilizzando una catena di server proxy noti come *mix* che ricevono messaggi da più mittenti, li mescolano e li rispediscono in ordine casuale alla destinazione successiva (possibilmente un altro nodo mix). In questo modo si interrompe il collegamento tra la fonte della richiesta e la destinazione, rendendo più difficile tracciare le comunicazioni end-to-end. Inoltre, i nodi mix conoscono solo il nodo da cui hanno ricevuto immediatamente il messaggio e la destinazione immediata a cui inviare i messaggi rimescolati, rendendo la rete resistente ai nodi mix maligni.

[3] In crittografia una dimostrazione a conoscenza zero è un metodo utilizzato da un soggetto per dimostrare ad un altro soggetto che una affermazione è vera, senza rivelare nient'altro oltre alla veridicità della stessa.

Alice	
Bob	
Carlo	
Dario	
Edoardo	
	a6q21p

Figura 4.3: Modello di scheda elettorale di Prêt à Voter

4.3.2 Il punto di vista dell'elettore

Esistono diverse implementazioni del sistema di voto Prêt à Voter, che danno luogo a variazioni nei dettagli dell'esperienza di voto, ma hanno caratteristiche comuni.

Il fulcro del processo di voto prevede di ottenere un voto criptato da un elettore. Ciò richiede che le informazioni necessarie per votare (di solito l'elenco dei candidati) siano presentate all'elettore, in modo che il voto possa essere registrato in forma criptata. Le informazioni vengono distrutte una volta che l'elettore le ha utilizzate per esprimere il voto, per cui il voto può essere recepito solo attraverso la decodifica.

Layout della scheda di voto

La scheda elettorale standard di Prêt à Voter, illustrata nella Figura 4.3, è composta da due metà separabili, ciò consente di rimuovere la lista dei candidati e il suo ordine una volta espressa la propria preferenza, lasciando il voto in forma puramente criptata, ovvero la metà rimanente della scheda, insieme all'*onion*.

Tutte le elezioni che prevedono la votazione di candidati possono essere gestite con questa scheda: sia che le votazioni prevedano la selezione di un singolo candidato o una lista, indipendentemente da come verranno conteggiate.

Il sistema originale di Prét à Voter[Rya05][CRS05], proponeva l'uso di una particolare forma di permutazione: spostamenti ciclici dell'elenco dei candidati, in base ai quali l'elenco è sempre nello stesso ordine, ma può iniziare in qualsiasi punto arbitrario e procedere in modo circolare. Questo sistema è appropriato per registrare un voto a favore di un singolo candidato, poiché la posizione del voto non fornisce alcuna informazione sul candidato votato, ma non è lo stesso caso quando i voti sono espressi a favore di più di un candidato, poiché le posizioni relative dei candidati e le probabili combinazioni di voti faranno trapelare alcune informazioni sul voto espresso dal pattern dei voti segnati sul lato destro. In realtà, il sistema può essere generalizzato per consentire permutazioni arbitrarie dell'elenco dei candidati e può quindi essere utilizzato per registrare anche voti per più candidati.

Votazione

L'elettore esprime il proprio voto compilando le caselle apponendo un segno sul lato destro della scheda elettorale, in corrispondenza dei nomi scelti.

Le due metà della scheda elettorale vengono quindi separate. All'elettore verrà quindi richiesto di distruggere la metà sinistra, costituita dall'elenco dei candidati, come parte del processo di voto. Questo passo è necessario per garantire che l'elettore non possa in seguito dimostrare come ha votato, e quindi offre resistenza alla coercizione e al voto di scambio.

La metà destra della scheda elettorale, costituita dal voto da depositare, sarà prima scannerizzata nel sistema di voto Prêt à Voter e poi firmata. Dopodiché, l'elettore potrà conservarla come ricevuta del voto espresso. Si noti che il personale presente nel seggio elettorale dovrà aiutare l'elettore a controllare che la firma apposta sulla ricevuta sia valida. In questo modo ci si protegge da frodi elettorali con ricevute per voti che non sono mai stati espressi. Il sistema di scansione dovrà estrarre la posizione del voto sulla scheda e la codifica dell'ordinamento della lista per poter elaborare il voto.

Una possibile variante della ricevuta è che il sistema Prêt à Voter stampi una propria ricevuta (firmata) delle informazioni scansionate e la fornisca all'elettore. La corrispondenza con il voto originale consentirà all'elettore di confermare che le informazioni contenute nella scheda sono state correttamente comprese dal sistema.

Verifica delle schede elettorali

Gli elettori potrebbero voler verificare che l'ordine dei candidati che si presume criptato sul lato destro corrisponda effettivamente all'elenco stampato sul lato sinistro. Se così non fosse, il voto espresso per un candidato potrebbe essere considerato, dopo la decodifica, come un voto per un altro candidato. Per fornire questa garanzia, gli elettori possono scegliere di "controllare" una scheda elettorale. Ciò comporta la rimozione del lato sinistro della scheda elettorale e la richiesta al sistema di decifrare l'elenco dei candidati dall'*onion* sul lato destro. L'elettore può quindi verificare che l'elenco decriptato corrisponda a quello dei candidati stampato sul lato sinistro. In linea di principio, questa verifica può essere effettuata tutte le volte che l'elettore lo desidera. In questo modo l'elettore ha la certezza che le schede elettorali siano state realizzate correttamente.

Tuttavia, l'elettore non può esprimere un voto su una scheda decriptata. Una volta che la lista dei candidati associata a un *onion* è nota, la privacy del voto, e quindi la resistenza alla coercizione e al voto di scambio, viene meno. Il processo di verifica dà al singolo elettore la certezza che le schede elettorali siano realizzate correttamente, ma non gli consente di controllare la scheda che sta utilizzando per esprimere il proprio voto.

Verifica del voto

I voti espressi vengono pubblicati su una bacheca. Gli elettori possono verificare che le informazioni riportate sulla loro ricevuta (il voto e l'*onion*) appaiano sulla bacheca e questo dà loro la certezza che il loro voto è stato effettivamente accettato nel sistema come voto espresso. Se il voto non compare nella bacheca, o compare in modo errato, l'elettore può utilizzare la ricevuta per contestare il buon andamento dell'elezione, poiché fornisce la prova di un voto che non è stato incluso nel conteggio. Gli elettori devono quindi conservare le ricevute per evitare che il loro voto venga modificato.

Anche il processo di decodifica dei voti è soggetto a processi di verifica, ma l'elettore non sarà direttamente coinvolto. Una volta decriptati, i voti possono essere resi pubblici, consentendo a qualsiasi elettore interessato di verificare il conteggio dei voti e il risultato delle elezioni.

Stampa anticipata/on-demand di schede elettorali

La segretezza della scheda elettorale si basa sul fatto che l'elenco dei candidati non può essere dedotto dall'*onion* senza la possibilità di decifrarla. Tuttavia, la scheda elettorale stessa, quando è intera, fornisce un'associazione tra la lista dei candidati e la sua cifratura presente sul lato destro. Ciò significa che le schede elettorali devono essere gestite con attenzione e la catena di custodia tra la creazione di una scheda elettorale e il suo utilizzo in un seggio elettorale deve essere affidabile.

Un approccio alternativo è quello di stampare le schede elettorali nel momento in cui siano necessarie. Ciò è possibile fornendo una scheda elettorale con la lista dei candidati criptata in due modi diversi: uno che può essere decriptato nel seggio elettorale e uno che può essere decriptato dagli scrutatori come descritto in precedenza. L'elenco dei candidati viene poi stampato su richiesta, quando la scheda elettorale deve essere utilizzata al seggio. Le schede elettorali possono essere verificate come in precedenza, chiedendo di rivelare la lista dei candidati associata al lato destro e controllando che corrisponda al lato sinistro.

Con l'eccezione della stampa della scheda su richiesta dell'elettore, l'esperienza di voto dell'elettore è identica a quella dell'utilizzo di una scheda prestampata.

4.4 Civitas

Civitas [CCM08] è stato il primo sistema di voto elettronico che consente agli elettori di votare in modo relativamente sicuro da un client remoto di loro scelta, garantendo al contempo verificabilità universale, verificabilità individuale, anonimato, integrità e resistenza alla coercizione.[Wan+17] Utilizza un servizio di *log* pubblico, come una bacheca, per registrare tutte le informazioni necessarie alla verificabilità delle elezioni. Per resistere alla coercizione, un elettore deve utilizzare la sua chiave privata designata ed eseguire un algoritmo per generare credenziali false. L'elettore fornisce queste credenziali false in caso di coercizione da parte degli avversari. Tutti i voti inviati tramite credenziali false vengono eliminati e la rivotazione dipende dalla politica specificata dal supervisore.

4.4.1 Descrizione del sistema

Civitas prevede cinque tipi di attori: un supervisore, un *registrar*, gli elettori, gli scrutatori di registrazione e gli scrutatori di conteggio. Gli agenti diversi dagli scrutatori sono le autorità elettorali:

- Il *supervisore* amministra l'elezione. Ciò include la definizione della scheda elettorale e degli scrutatori, nonché l'avvio e l'interruzione delle elezioni.
- L'*registrar* autorizza gli elettori.
- Gli *scrutatori di registrazione* generano le credenziali che gli elettori utilizzano per esprimere il proprio voto.
- Gli *scrutatori di conteggio* conteggiano i voti.

Questi attori utilizzano un servizio di *log* con memoria di sola lettura pubblica. L'integrità dei messaggi nel *log* è garantita dalle firme digitali inserite degli attori. In una

singola elezione vengono utilizzate più istanze del servizio di *log*. Un'istanza, chiamata bacheca, viene utilizzata dalle autorità elettorali per registrare tutte le informazioni necessarie per la verificabilità delle elezioni. Le altre istanze, chiamate urne, sono utilizzate dagli elettori per esprimere il proprio voto.

Fase di impostazione

In primo luogo, il supervisore crea l'elezione pubblicando il formato della scheda su una bacheca vuota e identifica gli scrutatori pubblicando pubblicando le loro chiavi pubbliche individuali.

Successivamente, l'ufficiale di stato civile pubblica la lista elettorale, contenente gli identificativi (nominativi o numeri di registrazione) di tutti gli elettori autorizzati, insieme alle loro chiavi pubbliche. Si presume che ogni elettore abbia due chiavi, una di registrazione e una di designazione, il cui uso è descritto di seguito. Gli scrutatori generano collettivamente una chiave pubblica El Gamal distribuita e la pubblicano sulla bacheca. La decrittazione dei messaggi criptati con questa chiave richiede la partecipazione di tutti gli scrutatori.

Infine, gli scrutatori di registrazione generano le credenziali che verranno utilizzate per autenticare i voti in modo anonimo. Ogni credenziale è associata a un singolo elettore. Come le chiavi di un sistema crittografico asimmetrico, le credenziali sono coppie di un valore pubblico e di un valore privato. Tutte le credenziali pubbliche sono pubblicate in bacheca e ogni scrutatore di registrazione memorizza una parte di ogni credenziale privata. Le credenziali private possono essere falsificate o trapelate solo se tutti gli scrutatori di registrazione colludono.

Fase di voto

Gli elettori si registrano per acquisire le loro credenziali private. Ogni scrutatore di registrazione autentica un elettore utilizzando la sua chiave di registrazione. Lo scrutatore e l'elettore eseguono quindi un protocollo, utilizzando la chiave di designazione dell'elettore, che rilascia la parte dello scrutatore della credenziale privata dell'elettore a quest'ultimo. L'elettore combina entrambe le parti per ottenere una credenziale privata.

Il voto può avvenire immediatamente o molto tempo dopo la registrazione. Per votare, l'elettore invia la credenziale privata e la scelta di un candidato (entrambe criptate), insieme a una prova che il voto è conforme, ad alcune o a tutte le urne.

Resistenza alla coercizione L'idea chiave (dovuta a Juels, Catalano e Jakobsson in *Coercion-resistant electronic elections*[JCJ05]) per permettere agli elettori di resistere alle pressioni esterne e prevenire il voto di scambio è quella di consentire agli elettori di sostituire le loro credenziali reali con credenziali false e agire come richiesto da un avversario.

Voto ripetuto Gli elettori possono presentare più di un voto per ogni credenziale. Il supervisore ha la facoltà di specificare una politica per il conteggio di tali voti ripetuti. Se le rivotazioni non sono consentite, tutti i voti presentati con credenziali duplicate vengono eliminati. Se le rivotazioni sono consentite, l'elettore deve includere nei voti successivi una prova che indichi quali voti precedenti devono essere sostituiti. Questa

Se l'avversario chiede che l'elettore...	**Allora l'elettore...**
Inserisca un voto particolare	Lo fa con una credenziale falsa.
Venda o ceda una credenziale	Fornisce una credenziale falsa.
Si astenga	Fornisce una credenziale falsa all'avversario e vota con una vera.

prova deve dimostrare la conoscenza delle credenziali e della scelta utilizzata in entrambi i voti, impedendo a un avversario di rivotare per conto di un altro elettore.

Design della scheda Civitas è compatibile con l'uso di qualsiasi tipo di scheda per la quale sia possibile ottenere una prova di conformità. Supporta l'uso di schede in cui gli elettori possono scegliere un singolo candidato, un qualsiasi sottoinsieme di candidati o una classifica dei candidati. Anche le votazioni per iscritto potrebbero essere supportate da Civitas, poiché qualsiasi scrittura potrebbe essere considerata conforme.

Fase di scrutinio

Gli scrutatori di conteggio effettuano collettivamente il conteggio delle elezioni:

1. *Recupero dei dati.* Tutti gli scrutatori recuperano i voti da ogni urna e le credenziali pubbliche dalla bacheca.

2. *Verifica delle prove.* Gli scrutatori controllano ogni voto per verificare la prova di conformità. Ogni voto con una prova non valida viene scartato.

3. *Eliminazione dei duplicati.* Per ogni credenziale viene conservato al massimo un voto. I voti con credenziali duplicate vengono eliminati in base alla politica di rivotazione.

4. *Anonimizzazione.* Sia l'elenco dei voti inviati che l'elenco delle credenziali autorizzate vengono resi anonimi applicando una permutazione casuale, implementata con una *mix network*.[Cha81]

5. *Eliminazione dei voti non autorizzati.* Le credenziali dei voti anonimizzati vengono confrontate con le credenziali autorizzate anonimizzate. I voti con credenziali non valide vengono scartati.

6. *Decriptazione.* Le scelte rimanenti, ma non le credenziali, vengono decriptate. Il conteggio finale è pubblicamente calcolabile.

Da notare che Civitas non garantisce l'indipendenza del software, poiché è necessario fidarsi del creatore delle credenziali private nella fase di eliminazione dei voti falsi. Se il sistema elimina un voto valido, non sembra esserci un cambiamento rilevabile nell'esito della votazione.

4.4.2 Modifiche del protocollo JCJ

Civitas è la prima implementazione concreta del protocollo JCJ con importanti modifiche, in particolare il protocollo per la registrazione degli elettori. Nel protocollo JCJ originale, un ufficiale di stato civile corrotto poteva semplicemente rilasciare credenziali false agli elettori senza che questi fossero in grado di rilevare la frode elettorale. La distribuzione della funzionalità di registrazione per JCJ è stata discussa anche in [Kri07]. Koenig [KHF11] suggerisce dei miglioramenti, tra cui la protezione contro un potenziale attacco *denial-of-service* in cui gli aggressori inondano la bacheca di voti con credenziali false, creando così gravi ritardi di elaborazione. Altre proposte includono la formalizzazione del processo di voto in Civitas, un'interfaccia utente ispirata in parte a Helios, e la gestione delle credenziali tramite smartcard [NV12][Neu+13].

L'architettura di base di JCJ ha ispirato altri sistemi di voto *end-to-end* che ne migliorano i vari aspetti di usabilità. Ad esempio, Selections [CH11] offre agli elettori un modo più semplice per registrarsi e gestire le credenziali, incorporando l'uso delle *panic password*, cioè credenziali di voto che l'elettore può creare al volo. Trivitas [BGR12] adatta Civitas per semplificare ulteriormente il processo di verificabilità, non intuitivo e complesso per l'elettore profano, introducendo la nozione di voto di prova. Questi vengono espressi insieme ai voti normali e vengono elaborati più o meno allo stesso modo, tranne per il fatto che i voti di prova vengono decriptati e rivelati in fasi diverse, consentendo all'elettore di trarre una forte fiducia sul fatto che il suo voto effettivo venga gestito correttamente. Caveat Coercitor [Gre+13] estende la proprietà di resistenza alla coercizione di JCJ per scenari più generali, tra cui la perdita di credenziali di voto a causa di malware, *registrar* corrotti e attacchi di impersonificazione.

4.5 Polyas

Polyas è un sistema di voto elettronico da remoto creata dall'azienda tedesca Micromata GmbH, utilizzato sin dal 1996 in diverse elezioni, tra cui quelle per il rinnovo di organi collegiali della Banca d'Italia, l'elezione dell'assemblea nazionale di Unitre e l'elezione del comitato nazionale del Partito Radicale.

Nel 2008, il BSI - Bundesamt für Sicherheit in der Informationstechnik (l'Ufficio federale tedesco per la sicurezza informatica), ha certificato e rilasciato il profilo di protezione Common Criteria Sulla base di questo profilo di protezione, Polyas è attualmente il primo sistema di voto elettronico remoto valutato secondo lo standard di valutazione internazionale Common Criteria (ISO/IEC 15408), che definisce l'"insieme di requisiti di base per i prodotti di voto online".[MR10]

4.5.1 Descrizione del sistema

Il sistema di voto elettronico Polyas può essere classificato in diversi modi. Secondo la classificazione presentata in [Vol09] Polyas è descritto come un sistema di voto elettronico da remoto che utilizza:

- Una tecnica basata sul segreto per l'autenticazione dell'elettore. L'ID dell'elettore è noto a priori, come se fosse un numero di iscrizione o un numero di identificazio-

ne. Il vero token di autenticazione è un TAN[4] di voto che viene inviato all'elettore per posta e viene utilizzato per una sola elezione.

- Il principio della "separazione dei compiti" per garantire la segretezza elettorale nella fase di voto.
- Un browser web senza JavaScript (thin client) come interfaccia dell'elettore.

Componenti e responsabilità

Le componenti principali del sistema Polyas sono le seguenti:

- *Vote Casting Interface*, (VCI): interfaccia web che viene mostrata all'elettore a cui si può accedere da diversi browser web. Facilita l'autenticazione dell'elettore e l'espressione del voto;
- *Electoral Registry Server*, (ERS): server che autentica gli elettori e verifica la loro idoneità a partecipare a un'elezione;
- *Validation Server*, (VS): il server di convalida, controlla l'ERS mediante un'ulteriore autenticazione del materiale di voto da parte degli elettori;
- *Ballot Box Server*, (BBS): server delle urne, memorizza i voti espressi in forma criptata;
- *Tallying Component*, (TC): carica i voti criptati, li decifra e calcola il risultato delle elezioni;
- *Committee Tool* (CT): consente ai funzionari elettorali di avviare e interrompere le elezioni.

Queste componenti sono così collegate:

- VCI - ERS: per verificare il diritto di voto;
- VS - ERS: per controllare la decisione dell'ERS sul diritto di voto dell'elettore e per generare token di autorizzazione anonimi e casuali T;
- VCI - BBS: per esprimere un voto;
- ERS - BBS: per informare il BBS sui token di autorizzazione validi e viceversa per informare l'ERS sui token non autorizzati perché sono stati espressi voti corrispondenti.

Fase di impostazione del voto

In questa fase, i funzionari elettorali eseguono una serie di sei passaggi.

In primo luogo, vengono generati i TAN di voto. Vengono quindi prodotti due elenchi: una *hashlist* e un elenco criptato di TAN di voto, il primo contiene le informazioni sull'identità degli elettori e viene inserito nel registro elettorale, il secondo viene inviato,

[4] Il *transaction authentication number* (TAN) è una forma di password monouso (OTP) che rappresenta un secondo livello di sicurezza oltre alla tradizionale autenticazione con sola password. Agiscono come una forma di autenticazione a due fattori.

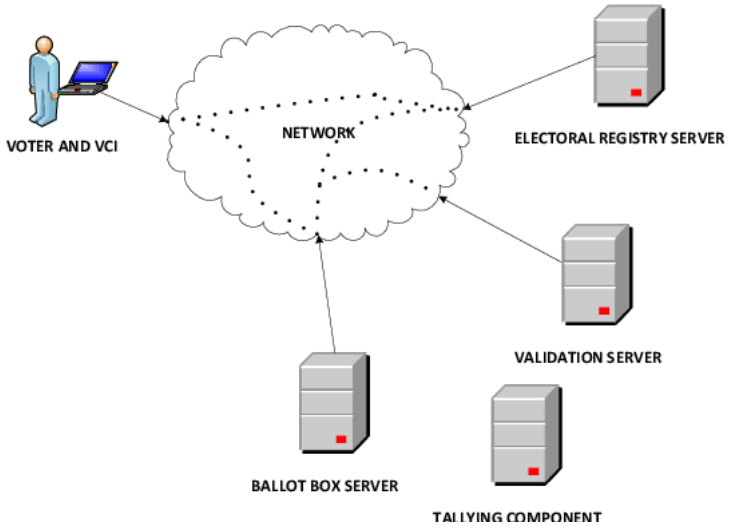


Figura 4.4: Interazione dei componenti di Polyas

insieme ai nomi e agli indirizzi postali degli elettori, a un fornitore di servizi che stampa il materiale elettorale. Questo fornitore di servizi decifra i TAN utilizzando la sua chiave privata e ne applica una versione decifrata a ciascun materiale elettorale. Questi vengono poi spediti agli elettori utilizzando le informazioni sull'indirizzo postale fornito.

Nella seconda fase, vengono generate le coppie di chiavi per i tre server e per il TC, ovvero:

- coppie di chiavi HTTPS per tutti e tre i server,
- una coppia di chiavi per la firma crittografica per il BBS,
- una coppia di chiavi del database per il TC.

Le chiavi pubbliche HTTPS per l'ERS e il BBS sono pubblicate sulla pagina web delle elezioni e le *fingerprint* corrispondenti sono stampate sul materiale elettorale. Inoltre, le chiavi pubbliche sono distribuite ai diversi componenti. I componenti contengono le seguenti chiavi:

- L'ERS ha una coppia di chiavi HTTPS per la comunicazione con gli elettori e gli altri componenti. Inoltre, conserva le chiavi pubbliche HTTPS degli altri due server e la chiave pubblica di firma di BBS.
- Il VS contiene una coppia di chiavi HTTPS per la comunicazione con gli altri componenti. Inoltre, memorizza le chiavi pubbliche HTTPS degli altri due server.
- Il BBS ha una coppia di chiavi HTTPS per la comunicazione con gli elettori e gli altri componenti e le chiavi pubbliche HTTPS di ERS e VS. Inoltre, conserva la chiave di firma privata del BBS e la chiave pubblica del database del TC, utilizzata per la crittografia dei voti.
- Il TC contiene la chiave privata della sua coppia di chiavi del database, che utilizza per decifrare i voti al termine delle elezioni. Questa chiave privata è memorizzata in modo criptato e la decodifica è protetta da due *passphrase*, ciascuna nota a un solo funzionario elettorale.

Nella terza fase, il registro elettorale viene installato nell'ERS. Inoltre, l'elenco dei TAN idonei viene caricato nel database del VS e si verifica che il database delle urne presso il BBS sia vuoto. Successivamente, le schede elettorali per l'elezione (compresi i nomi dei candidati e il titolo dell'elezione) e le regole di voto vengono impostate sul BBS. Nella quinta fase, vengono generati due token di accesso per ogni server. Entrambi devono essere inseriti per accedere da remoto al server corrispondente. Questi sei token vengono distribuiti a sei funzionari elettorali indipendenti. Nella fase finale, i tre server vengono configurati e viene installato il software di sicurezza, come ad esempio un software antivirus. Il software Polyas viene quindi installato e le elezioni vengono avviate dai funzionari elettorali.

Fase di voto

La comunicazione tra i diversi componenti è protetta mediante Secure Socket Layer (SSL), illustrata come segue: tra l'elettore e l'ERS, SSL_1; tra l'ERS e il VS, SSL_2; tra il VS e il BBS, SSL_3; tra l'elettore e il BBS, SSL_4; e tra l'ERS e il BBS, SSL_5.

Supponendo che la comunicazione tra i componenti sia garantita dal SSL, il protocollo di voto può essere descritto come segue.

L'elettore visita la pagina web delle elezioni e, idealmente, verifica il certificato SSL dell'ERS. Presenta quindi il suo ID personale e il suo TAN. L'ERS verifica l'idoneità dell'elettore nel registro elettorale. Se l'elettore è idoneo, l'ERS invia il TAN al VS. Il VS verifica l'idoneità dell'elettore controllando se il suo database contiene il TAN. In caso affermativo, il VS genera un token di voto casuale T. Questo token di voto viene generato solo se sia l'ERS che il VS determinano che l'elettore è idoneo. Si noti inoltre che se al VS è stato presentato lo stesso TAN in precedenza, non genererà un nuovo token; piuttosto rimanderà lo stesso token all'ERS (ma non al BBS).

Dopo aver generato un nuovo token T, il VS lo invia al BBS. Il BBS memorizza temporaneamente questo token di voto e invia una conferma. Dopo aver ricevuto questa conferma, il VS invia lo stesso token di voto T all'ERS che lo memorizza temporaneamente e invia una conferma al VS. Per evitare il voto ripetuto, dopo aver ricevuto questa conferma, il VS contrassegna il TAN come non valido e invia una conferma all'ERS.

L'ERS invia T all'elettore, che viene quindi automaticamente inoltrato al BBS per effettuare una richiesta (contenente T) di voto. Idealmente, l'elettore verifica il certificato SSL della connessione HTTPS con il BBS.

Il BBS verifica se T è un token di voto valido. Il BBS considera T valido se è stato ricevuto dal VS e non è ancora stato cancellato (perché ciò significherebbe che è già stato espresso un voto). In tal caso, la scheda viene visualizzata dall'elettore nel suo browser ed egli effettua una scelta tra le opzioni elettorali disponibili. Il token T e la selezione dell'elettore vengono inviati al BBS. Se il token di voto T è valido, il BBS cripta e memorizza la selezione nel database delle urne insieme al token di voto T. Quindi invia di nuovo il token di voto e la selezione memorizzata (in testo in chiaro usando solo HTTPS) e l'informazione se si tratta di un voto valido o non valido. La selezione viene quindi nuovamente mostrata all'elettore. I passaggi qui descritti possono essere ripetuti se l'elettore vuole cambiare la sua selezione, prima di esprimere il suo voto finale. Infine, l'elettore esprime il suo voto confermandolo nel browser. Un messaggio di conferma contenente il token di voto T viene inviato al BBS, che verifica se T è ancora valido. In caso affermativo, il BBS cambia lo stato della voce nel database appartenente al token di voto T da "selezione" a "voto espresso".

Dopodiché, il BBS invia il token di voto T all'ERS, che cancella il token temporaneamente registrato. Inoltre, imposta l'ID elettore corrispondente come non valido. Quindi invia un messaggio di conferma al BBS che cancella il token di voto corrispondente dal database e qualsiasi altra copia temporaneamente memorizzata. L'eliminazione dei token di voto consente di mantenere la privacy dell'elettore, in quanto viene eliminato qualsiasi legame tra l'elettore e il suo voto espresso. L'elettore riceve un messaggio di conferma del voto espresso con successo e questo segna la fine della fase di voto.

I voti espressi sono memorizzati in ordine casuale nel database delle urne, in blocchi di trenta. Questi blocchi sono stati introdotti per implementare un meccanismo che consenta la verifica in caso di problemi durante la fase post-voto. Questo meccanismo funziona nel modo seguente: Si crea una hash chain applicando una funzione di hash a un blocco di trenta voti criptati. Pertanto, non appena la BBS riceve i primi trenta voti, concatena questi voti criptati, aggiunge un valore hash iniziale, calcola l'hash e firma il risultato utilizzando la chiave di firma privata. L'output e la versione firmata vengono inviati all'ERS. La firma viene verificata e, se valida, il messaggio viene memorizzato in un database separato presso l'ERS. Un messaggio di conferma viene inviato alla BBS. Il blocco successivo di trenta voti viene collegato a questo output con hash e la funzione di hash viene applicata nuovamente. Questo processo viene ripetuto per tutti i voti disponibili. Al termine delle elezioni, l'ultimo blocco di voti può contenere meno di trenta voti. In tal caso, i voti non vengono inclusi nella catena di hash.

Fase post-voto

Al termine del periodo elettorale, i funzionari addetti chiudono le elezioni utilizzando i diversi token di accesso per accedere ai tre server coinvolti. Scaricano poi il database delle urne dal BBS, il registro elettorale (compreso lo status di chi ha espresso un voto) dall'ERS e il database dal VS. Il database delle urne viene caricato in TC. Vengono inserite le due *passphrase* e si procede al conteggio. I risultati vengono visualizzati e stampati. Si noti che vengono presi in considerazione solo i voti in cui lo stato è impostato su "espresso" e non quelli in cui lo stato è impostato su "selezione". I voti espressi (decifrati) vengono stampati e conservati per consentire una verifica manuale in caso di controversie. Il numero di voti espressi secondo il VS e l'ERS viene dedotto dai database corrispondenti e confrontato con il numero di voti nel database delle urne. Inoltre, il contenuto dei database dei tre server viene archiviato su un CD.[OSV11]

4.5.2 Verifica

Verifica *cast-as-intended* Per ottenere una verifica *cast as intended*, l'elettore esamina il codice sorgente HTML della pagina web elettorale quando viene visualizzata la scheda. In questo modo, vede i dati trasmessi al sistema quando fa clic su ciascun candidato. Sebbene questo sia generalmente possibile, presenta problemi di usabilità. Non molti elettori sanno come accedere al codice HTML per controllare queste informazioni sulle pagine web. Inoltre, se la scheda elettorale presenta molte opzioni di scelta, questo processo richiederà molto tempo nonostante l'esperienza dell'utente. Pertanto, questa verifica dovrebbe essere effettuata automaticamente e non manualmente. È necessario uno strumento corrispondente, che dovrebbe essere fornito da istituzioni indipendenti e affidabili. Si noti che, anche fornendo questa verificabilità all'elettore, Polyas rimane comunque privo di ricevute.

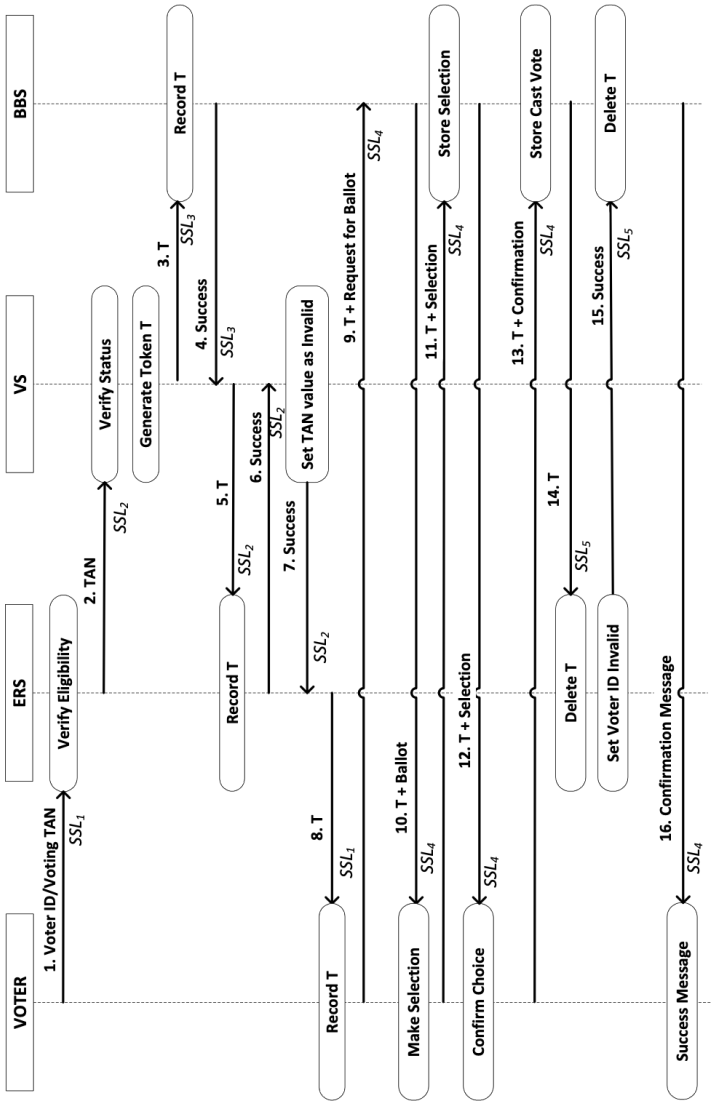


Figura 4.5: Diagramma d'interazione della fase di voto nel protocollo Polyas

Verifica *recorded-as-cast* In [OSV11] viene presentato uno strumento di verificabilità per fornire verificabilità individuale parziale in termini di *recorded-as-cast* e verificabilità universale in Polyas.

Gli input richiesti dai componenti o entità corrispondenti sono:

- File della scheda elettorale dal BBS: questo file contiene i nomi dei candidati e dei voti criptati, gli identificatori dei blocchi, l'ordine dei blocchi e il loro stato.
- File dei valori hash dell'ERS: Sono i valori hash memorizzati.
- File *keystore* del TC: il file *keystore* contiene la chiave privata RSA con cui viene realizzata la crittografia asimmetrica. Il file del *keystore* stesso e la chiave privata sono protetti da *passphrase*.
- *Passphrase* del file del *keystore* da parte dei funzionari elettorali: la *passphrase* è necessaria per accedere al *keystore*.
- Password della chiave privata da parte dei funzionari elettorali: La password deve essere inserita per accedere alla chiave privata.
- Numero massimo di candidati da parte dei funzionari elettorali: questo numero specifica il numero massimo di candidati per esprimere un voto valido. È necessario affinché lo strumento possa distinguere un voto valido da uno non valido.
- Numero di elettori dal registro elettorale: Numero di elettori online secondo il registro elettorale.

Sulla base di questi dati vengono elaborate tre diverse verifiche. In primo luogo, il numero di voti criptati con lo stato "espresso" nell'urna viene confrontato con il numero di elettori. Successivamente, tutti i valori di hash della catena di hash vengono generati dallo strumento in base alle informazioni contenute nel file delle urne. Ogni valore viene confrontato con quello corrispondente nel file dei valori hash. Nella terza fase di verifica i voti criptati vengono decriptati voto per voto. A tal fine, viene estratta la chiave privata dal *keystore* e per ogni voto viene prima decifrata la chiave AES e poi il voto criptato viene decifrato con questa chiave. In base al numero massimo di candidati che possono esprimere un voto valido e ai voti decifrati, il risultato viene conteggiato e visualizzato. A questo punto, il risultato deve essere confrontato manualmente con il risultato emesso dal TC. È necessario verificare con successo tutte e tre le fasi.

4.6 SecureBallot

Avendo come obiettivo quello di creare un sistema di voto elettronico che mantenesse le caratteristiche desiderabili delle votazioni tradizionali, migliorandone la sicurezza e l'usabilità, alcuni ricercatori dell'Università degli Studi di Palermo hanno sviluppato SecureBallot[Aga+21], un sistema di voto elettronico stand-alone *open source* che si prefigge il compito di proteggere la privacy dei partecipanti e garantire la segretezza, l'anonimato, l'integrità, l'unicità e l'autenticità dei voti sfruttando tecniche crittografiche.

Pubblicando l'intero progetto su GitHub[5], i ricercatori pongono l'accento sull'importanza della caratteristica *open source*, attribuendo una delle ragioni principali della

[5] https://github.com/ndslab-unipa/SecureBallot

sfiducia nei confronti dei sistemi di voto elettronico alla presenza maggioritaria di soluzioni esistenti basate su software proprietario, non analizzabili né verificabili. Di conseguenza, secondo i ricercatori, SecureBallot ottiene tre vantaggi principali: maggiore sicurezza, trasparenza e la possibilità di adattare il sistema a esigenze specifiche.

Con l'ottica di mantenere di mantenere le caratteristiche desiderabili delle elezioni tradizionali, il sistema appartiene alla categoria delle macchine per il voto elettronico in rete nei seggi elettorali, che operano attraverso seggi elettorali ripartiti collegati a un'urna virtuale centrale, che raccoglie le preferenze espresse dagli elettori in forma criptata. Questa classe di sistemi di voto digitale prevede l'utilizzo di PC, laptop, tablet e altri dispositivi *off-the-shelf*.

Per preservare la privacy dell'elettore, il sistema separa rigorosamente la fase di identificazione dal voto vero e proprio, in modo che nessuna informazione sugli elettori venga memorizzata o raccolta in alcun modo, adottando un approccio basato sui token ([SP06][Mor+20]), introducendo il concetto di utente virtuale: i seggi elettorali assegnano casualmente delle identità temporanee agli utenti, consentendo loro di votare in una delle qualsiasi cabine di voto disponibili. I token degli utenti virtuali vengono costantemente riutilizzati per diversi elettori, garantendo così la loro privacy.

SecureBallot è stato testato avendo come caso di studio le elezioni universitarie, essendo stato utilizzato per un periodo di sei mesi presso l'Università di Palermo, sia su elezioni fittizie che reali. Permettendo di raccogliere e analizzare il feedback di centinaia di utenti eterogenei in diverse elezioni universitarie, somministrando agli elettori appositi questionari.

4.6.1 Attori del sistema

Per presentare le caratteristiche di SecureBallot, si considerano due diversi tipi di attori che interagiscono con il sistema: gli elettori e i membri dello staff. Le generalità di tutti gli elettori validi vengono preventivamente inserite in un elenco centralizzato, accessibile dal personale autorizzato in qualsiasi seggio elettorale; in questo modo, è sempre possibile verificare se una persona ha effettivamente il diritto di voto e se ha già votato o meno. I membri del personale possono essere suddivisi in tre gruppi, a seconda del loro ruolo e della fase elettorale di cui si occupano:

- membri del personale coinvolti nelle operazioni preliminari, che preparano e controllano l'elenco digitale degli elettori, creano le tessere elettorali digitali e generano tutte le chiavi segrete che saranno necessarie per attivare le postazioni di voto il giorno delle elezioni;
- membri del personale incaricati di supervisionare fisicamente tutte le operazioni di voto; i loro compiti comprendono la verifica del corretto funzionamento dei sistemi di voto, l'allestimento delle postazioni di voto utilizzando le chiavi segrete precedentemente menzionate, la verifica dell'identità degli utenti e la loro registrazione prima che siano autorizzati a votare;
- il supervisore elettorale, che supervisiona le operazioni di conteggio, firma digitalmente i risultati e li pubblica.

SecureBallot prevede che tutte le operazioni di voto siano criptate con schemi asimmetrici e che le chiavi necessarie per ogni elezione siano generate da un pubblico ufficiale (ad esempio un notaio o un agente di polizia) che garantisca il corretto svolgimento delle

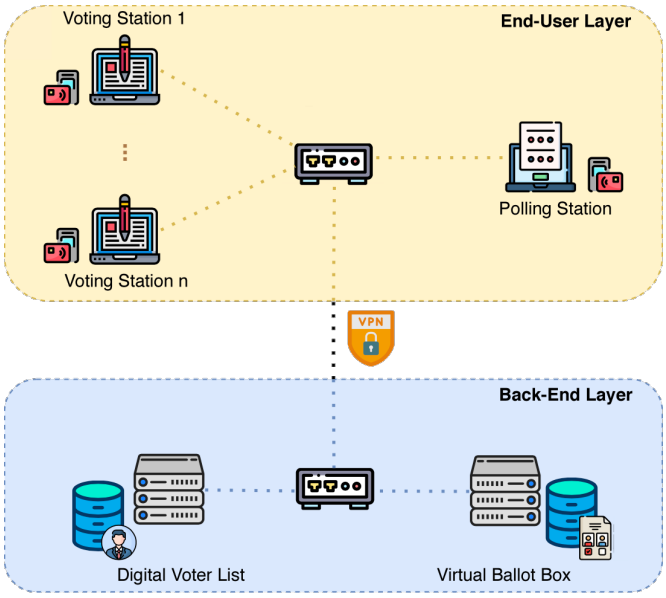


Figura 4.6: Architettura dei componenti di SecureBallot

operazioni di voto. La chiave pubblica dell'elezione in corso viene utilizzata come KEK (*key encryption key*), mentre la corrispondente chiave privata viene tenuta segreta dal pubblico ufficiale fino alla fase di scrutinio, in modo che nessuno possa interferire con il corretto svolgimento delle elezioni.

4.6.2 Componenti hardware e software

SecureBallot può utilizzare PC, laptop e tablet facilmente reperibili, in modo da ridurre i costi organizzativi. Il sistema complessivo comprende diverse applicazioni software che vengono eseguite simultaneamente su più macchine fisiche, l'architettura in questione prevede l'utilizzo di:

- un'applicazione software, chiamata seggio elettorale, utilizzata dal personale per monitorare le operazioni di voto e verificare se gli utenti hanno diritto di voto;
- un'applicazione software installata in ogni cabina elettorale, chiamata postazione di voto, che consente agli utenti di esprimere le proprie preferenze inserendo la scheda di voto digitale in un'urna virtuale;
- un'applicazione software centralizzata che funge da urna elettorale virtuale e raccoglie i voti espressi dagli utenti in forma criptata.

La votazione comporta una continua interazione tra queste applicazioni software; per garantire un adeguato livello di sicurezza a livello di rete tutti i componenti del sistema sono collegati tramite una VPN (*Virtual Private Network*) dedicata. Dopo l'identificazione iniziale, ogni elettore riceve un token di sblocco temporaneo, che gli consente di sbloccare la postazione di voto designata e di votare. In SecureBallot, i token di sblocco prendono la forma di tag NFC, ma il sistema è compatibile con prodotti simili che utilizzano la crittografia simmetrica per garantire l'autenticazione e l'integrità dei dati. Questi token sono completamente anonimi e sono associati solo a utenti virtuali. I token non sono in alcun modo collegati agli utenti reali, per garantire la loro privacy.

Il back-end del sistema è costituito da un'urna virtuale che contiene tutti i voti criptati e dal database contenente l'elenco delle persone autorizzate a votare. L'urna virtuale è il componente più sensibile del sistema, in quanto ha il compito di ricevere tutti i voti criptati inviati dai seggi elettorali e di conservarli in modo sicuro e affidabile fino alla fase di conteggio. Vale la pena notare che gli utenti possono votare solo se la connessione tra la postazione di voto e l'urna è attiva e funzionante, perché l'invio di un voto e la sua memorizzazione nell'urna sono considerati come un'unica operazione atomica, che viene gestita utilizzando transazioni di database distribuiti. Per garantire il completo anonimato di tutti gli elettori, l'urna virtuale non riceve alcuna informazione sugli utenti. Gli unici dati gestiti dall'urna virtuale sono quelli relativi alle schede elettorali anonime, che vengono criptate con una chiave simmetrica generata casualmente a *runtime*, a sua volta criptata con la chiave pubblica dell'elezione che funge da KEK.

4.6.3 Fasi della procedura elettorale

Configurazione

In questa fase, gli organizzatori scelgono il numero e l'ubicazione dei seggi elettorali, e le schede elettorali vengono create e adattate al numero e al tipo di candidati attraverso l'uso di un'interfaccia grafica dedicata, alla quale possono accedere solo i funzionari elettorali. In particolare, è possibile creare e configurare le schede elettorali specificando l'elenco dei candidati e il numero massimo di preferenze esprimibili. SecureBallot consente inoltre di caricare l'elenco degli elettori nel database centrale e di mantenerlo sincronizzato durante la fase di voto, aggiornando cioè lo stato di coloro che hanno già votato.

Altre operazioni svolte durante la configurazione riguardano la procedura di voto stessa, come l'impostazione della data di inizio e fine delle operazioni di voto e la configurazione delle postazioni di voto come nodi di una VPN.

Fase di voto

Durante questa fase gli elettori possono esprimere le loro preferenze. Gli ufficiali elettorali che si occupano del sistema dei seggi sono direttamente responsabili dei seguenti compiti:

- identificare i potenziali elettori secondo le norme della procedura elettorale in vigore (ad esempio, attraverso la carta d'identità);
- verificare se hanno diritto al voto e se non hanno votato in precedenza;
- utilizzare il software del seggio elettorale per attivare una delle postazioni di voto;
- offrire agli elettori le spiegazioni necessarie e fornire loro uno dei token di sblocco;
- assicurarsi che l'operazione di voto sia stata completata senza problemi e farsi restituire il token di sblocco.

Una volta autorizzato da un ufficiale elettorale, l'elettore può entrare nella cabina di voto assegnata, sbloccare la postazione di voto tramite il token di sblocco e indicare le sue preferenze. Il voto viene poi criptato e inviato all'urna centrale, in forma anonima,

garantendo i requisiti di sicurezza, integrità e coerenza di cui sopra. Infine, l'elettore deve restituire il token di sblocco agli ufficiali elettorali per completare l'operazione.

Una volta verificata l'ammissibilità dell'elettore, il sistema non ha più bisogno di informazioni sull'utente che sta votando. SecureBallot introduce l'idea di utenti virtuali per garantire l'anonimato degli elettori e allo stesso tempo assicurare che possano votare solo se autorizzati.

Per consentire solo agli utenti autorizzati di votare, il sistema prevede che l'utente virtuale v_i, a cui è stato assegnato il token t_k, possa votare solo una volta nella stazione di voto VS_j. Ogni utente può votare una sola volta nel seggio elettorale assegnato.

Di solito ci sono più utenti virtuali che seggi elettorali. In questo modo, se molti utenti arrivano al seggio elettorale nello stesso momento, e le cabine elettorali sono tutte occupate, i membri del personale possono comunque assegnare loro un'identità temporanea come utenti virtuali. Quando una cabina elettorale è disponibile, un nuovo utente virtuale può essere associato a quella cabina. Le operazioni di voto operazioni di voto possono quindi avvenire simultaneamente.

Tuttavia, se il numero di utenti virtuali fosse troppo elevato, cioè superiore al numero di elettori, a ogni utente sarebbe associato un valore unico (l'id dell'utente virtuale). Sebbene questo valore non rappresenti di per sé un rischio per la privacy, in quanto non viene memorizzato in alcun modo, sorge il rischio di creare un'associazione involontaria tra voto ed elettore. Il riutilizzo costante di identità virtuali raggiunge questo obiettivo, garantendo l'anonimato degli utenti.

In alcune procedure elettorali, gli elettori sono divisi in gruppi (ad esempio in base all'età o al territorio) e, a seconda del gruppo di cui fanno parte, avranno diritto a votare con una o più delle schede disponibili. In altre parole, il numero e il tipo di schede presentate a ciascun tipo di elettore può variare. L'uso di schede elettorali XML consente alle stazioni di voto di supportare tutte queste opzioni senza dover essere riconfigurate specificamente per ogni elezione. Gli utenti virtuali sono associati al tipo di elettore (e quindi al numero e al tipo di schede). La stazione di voto mostra quindi le schede corrette a ciascun elettore, pur non conoscendo la sua identità.

Per garantire la privacy, i voti vengono espressi in cabine di voto dedicate. Quando gli utenti entrano nella cabina a cui sono stati assegnati, interagiscono con il computer e il lettore di token che vi si trova. Gli utenti possono attivare il software della stazione di voto tenendo il token di sblocco ricevuto (ad esempio un tag NFC) vicino al lettore di token. Se l'utente si trova nella cabina di voto corretta, il suo token di sblocco viene riconosciuto e la stazione di voto viene attivata. A questo punto, le schede di voto vengono mostrate come un modulo che l'utente può compilare e inviare. Le schede compilate sono tenute segrete e memorizzate in un file XML, che contiene anche altri campi di utilità (ad esempio, informazioni sul seggio elettorale o sul tipo di elezione) necessari per fornire agli amministratori statistiche sull'affluenza alle urne.

Scrutinio

Dopo la conclusione della sessione di voto, il supervisore elettorale avvia il processo di conteggio. In SecureBallot, il processo di conteggio è molto semplice, sicuro e immediato.

L'unica persona autorizzata ad avviare la fase di conteggio è il supervisore elettorale, richiedendo l'accesso alla chiave privata della specifica elezione. Infatti, fino a questo momento, la suddetta chiave privata viene mantenuta segreta da un notaio o da un

pubblico ufficiale, che svolge un ruolo di garante. Questa figura, dopo un'accurata verifica della legittimità della richiesta e della corretta chiusura delle elezioni, consegna ufficialmente la chiave privata dell'elezione al supervisore elettorale. Inoltre, poiché ogni pacchetto elettorale è stato firmato digitalmente dall'urna virtuale prima di essere salvato, il sistema controlla, per ogni voto, che il pacchetto non sia stato modificato e che la firma sia ancora valida, prima di procedere all'effettivo conteggio.

Quindi, ogni chiave simmetrica utilizzata per criptare un pacchetto di voti è stata criptata con la chiave pubblica dell'elezione in corso e memorizzata nel file XML, insieme al voto effettivo.

Durante la fase di conteggio, ogni chiave simmetrica viene decifrata utilizzando la chiave privata fornita dal notaio e quindi utilizzata per decifrare il pacchetto di voti corrispondente. I pacchetti di voto decriptati vengono utilizzati per aggiornare il conteggio dei voti, in base alle preferenze specificate sulla scheda di voto. Per garantire l'integrità e l'autenticità dell'intero processo di conteggio, i risultati finali vengono firmati digitalmente dal supervisore elettorale.

Ogni campo della i-esima scheda di voto viene crittografato utilizzando una chiave simmetrica, K_i, generata automaticamente dalla rispettiva stazione di voto. SecureBallot genera ogni K_i e IV_i utilizzando un generatore di numeri casuali crittograficamente sicuro. La chiave risultante, K_iC viene inviata all'urna centrale insieme al pacchetto di voto criptato e a IV_i. Si noti ancora una volta che la chiave privata dell'elezione in corso è tenuta segreta e inaccessibile a tutti fino alla fine dell'elezione. Nemmeno il supervisore può accedervi e quindi non può interferire in alcun modo con le operazioni di voto.

Solo durante la fase di conteggio un'autorità esterna, come un notaio, consegna ufficialmente la chiave privata al supervisore elettorale, che può così iniziare il conteggio dei voti.

4.7 Sistema Estone

Già nel 2001, il Ministero della Giustizia estone aveva annunciato l'intenzione di introdurre il voto elettronico a distanza. Nel 2005, il voto elettronico a distanza è stato implementato come canale di voto aggiuntivo per le elezioni locali. Due anni dopo, il voto elettronico a distanza per le elezioni del Parlamento estone ha rappresentato il primo utilizzo di Internet come canale di voto in un'elezione parlamentare in tutto il Paese. Non c'è stato un processo di registrazione speciale, ma ogni elettore ha potuto votare utilizzando il voto elettronico a distanza. Anche se non c'è stato alcun segno di rifiuto del voto elettronico a distanza nelle elezioni del 2007, solo il 5,4% degli elettori ha votato utilizzando Internet come canale di voto.[AHT09]

Secondo la legislazione, il voto elettronico a distanza è consentito a tre condizioni principali: in primo luogo, l'elettore deve identificarsi e autenticarsi con la sua carta d'identità digitale, in secondo luogo, il voto elettronico a distanza deve essere implementato come voto anticipato (da sei a quattro giorni prima del giorno delle elezioni), e in terzo luogo, deve essere possibile l'aggiornamento del voto (in particolare, dopo aver espresso un voto elettronico, l'elettore può sovrascrivere questo voto esprimendo un voto cartaceo in un seggio elettorale avanzato).

4.7.1 Descrizione del sistema

Classificazione

Il sistema estone appartiene alla categoria dei sistemi di voto elettronico da remoto. Il processo di identificazione e autenticazione è basata sull'utilizzo della carta d'identità digitale estone, che identifica l'elettore via Internet e gli consente di firmare digitalmente documenti, come un voto criptato, utilizzando due codici PIN associati alla carta: uno per l'identificazione e uno per la firma. La segretezza del voto è garantita durante la fase di conteggio da un modulo di sicurezza hardware. Per quanto riguarda il software di voto lato client, il sistema estone segue l'approccio "thick-client"[6] e offre tre tipi di software per i sistemi operativi Windows, UNIX e Apple MacOS.

Componenti

A livello astratto, l'elettore si collega al server di voto e si identifica con la sua carta d'identità (utilizzando il PIN1). Quindi, il server di voto verifica l'identità dell'elettore e gli fornisce la scheda elettorale corrispondente. Dopo aver fatto la sua scelta, l'elettore firma digitalmente il suo voto criptato. Il server di voto verifica la firma dell'elettore. Nella fase di scrutinio, prima viene rimossa la firma digitale, poi i voti criptati vengono rimescolati e infine decriptati dall'HSM e contati. Le parti principali del sistema estone sono le seguenti:

- Server di registrazione (*Registration server*, RS)
- Server di certificazione (*Certification server*, CA)
- Server di archiviazione dei voti (*Vote storage server*, VSS)
- Software di conteggio su un PC separato (*Counting software on a separate PC*, CPC)
- Modulo di sicurezza hardware (*Hardware security module*, HSM)
- Software di voto lato client (*Client-side voting software*, CSS)

Il server di voto è quindi separato in tre server, ovvero il server di registrazione, di archiviazione dei voti e di certificazione, mentre l'ultimo è coinvolto ma non è impostato in particolare per la soluzione di voto elettronico a distanza, bensì per qualsiasi applicazione basata sulla carta d'identità digitale. Il software di conteggio è suddiviso in una parte che gira sul PC di conteggio e in una seconda che gira sul modulo di sicurezza hardware per decriptare i voti. Il sistema estone implementa i seguenti collegamenti di comunicazione:

- CSS - RS: per comunicare con il sistema di voto elettronico remoto
- RS - VSS: per inoltrare i voti all'archiviazione
- VSS - CA: per verificare se il certificato dell'elettore è ancora valido.

[6]Nelle reti informatiche, un rich client (chiamato anche heavy, fat o thick client) è un "client" nell'architettura di rete client-server che in genere fornisce funzionalità complete indipendentemente dal server centrale. Questo tipo di sistema è così chiamato in contrasto con "thin client", che descrive un computer fortemente dipendente dalle applicazioni di un server.

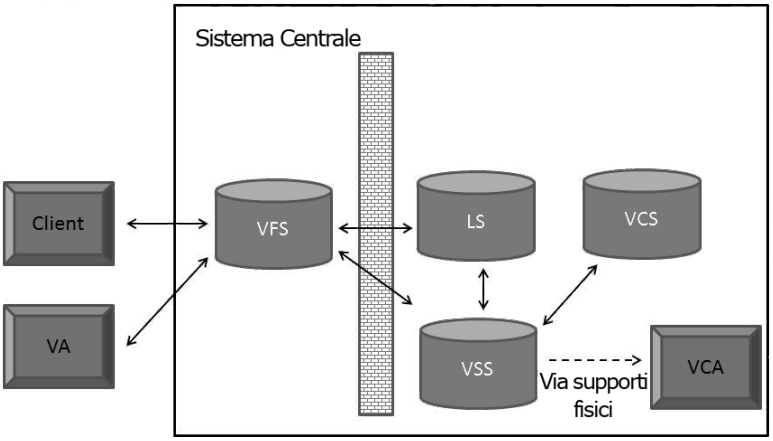


Figura 4.7: Componenti del sistema estone

4.7.2 Fase di impostazione delle elezioni

Preparazione sul lato server

Sul lato server, devono essere eseguite le seguenti fasi: si acquistano e si installano nuovi RS, VSS e CPC con sistema operativo, meccanismi di sicurezza (ad esempio firewall) e il relativo software di votazione/conteggio. Il modulo di sicurezza hardware viene configurato, cioè viene generata una coppia di chiavi. Mentre quella segreta è memorizzata sul dispositivo, la chiave pubblica è integrata nel software di votazione del client. Inoltre, vengono generate le chiavi per abilitare l'HSM: sette chiavi, che vengono distribuite ai membri della Commissione elettorale nazionale, e due per gli amministratori. Queste chiavi sono generate in modo che le due chiavi dell'amministrazione e quattro delle sette chiavi del comitato elettorale nazionale siano necessarie per abilitare l'HSM.

Preparazione da parte dell'elettore

L'elettore deve essere invece preparato a utilizzare il mezzo elettronico: oltre alla sua carta d'identità elettronica, deve essere in possesso di un lettore di smart card corrispondente e conoscere i suoi due PIN. Inoltre, se utilizza MacOS o Linux, l'elettore deve scaricare il software di voto lato client. Nel caso di un utente Windows, deve avere Java abilitato, in modo che il browser web possa caricare l'applet Java corrispondente.

4.7.3 Fase di votazione

Le fasi del protocollo di alto livello durante la fase di votazione sono descritte nella figura 4.8. Questa figura utilizza molte scorciatoie, pertanto vengono fornite alcune spiegazioni:

- *SSL* - Connessione SSL a due vie (con la prima chiave segreta dell'elettore abilitata con il PIN1).
- *elig?* - In questo caso il RS verifica se la persona richiedente è un elettore idoneo.
- *re-vote?* - Qui il VVS controlla se l'elettore richiedente ha già espresso un voto.[7]

[7]Nel caso in cui l'elettore abbia già espresso un voto, gli viene mostrata questa informazione e gli viene chiesto se vuole aggiornare il suo voto.

- *gen ballot* - genera la scheda che appartiene a questo particolare elettore.
- *choose* - l'elettore fa la sua scelta.
- *vote* - il sistema visualizza la scelta dell'elettore e l'elettore stesso verifica se vuole confermare la scelta o se vuole cambiarla di nuovo.
- *sig(m)* per firmare il messaggio m con la chiave segreta dell'elettore abilitata con il PIN2, mentre tale messaggio è implicitamente esteso con il certificato dell'elettore per la corrispondente chiave segreta. dell'elettore per la chiave segreta corrispondente.
- *enc(m)* indica la crittografia di un messaggio m con la chiave pubblica dell'HSM.
- *sig-ID* - il RS verifica se la firma appartiene alla persona che ha iniziato la sessione.
- *sig ok* - il VVS verifica la firma e la validità del certificato.

4.7.4 Fase di scrutinio

Dopo la fase di scrutinio elettronico e la chiusura dei seggi, i voti elettronici memorizzati presso i seggi dove gli elettori hanno espresso anche un voto cartaceo vengono etichettati con "non da contare". Viene quindi masterizzato un CD contenente l'ultimo voto elettronico ricevuto da ciascun elettore in ordine casuale (mentre quelli etichettati con "non da contare" vengono esclusi). Questo CD viene sigillato e consegnato al presidente della commissione elettorale nazionale estone. Il giorno delle elezioni, un'ora prima della chiusura dei seggi, inizia il processo di calcolo dei risultati. I voti elettronici vengono caricati sul CPC, che viene collegato all'HSM (via cavo).

Successivamente, l'HSM viene abilitato inserendo quattro delle sette chiavi del comitato elettorale e le due chiavi di amministrazione. A questo punto i voti criptati vengono inviati all'HSM voto per voto e quest'ultimo invia i voti decriptati corrispondenti. Una volta terminata la decrittazione, i voti vengono conteggiati e il risultato viene masterizzato su un altro CD. Questo CD viene caricato su un altro normale PC per visualizzare il risultato in modo leggibile. Il risultato viene firmato digitalmente dai presidenti del comitato elettorale nazionale. Il risultato firmato è quello legale.[Vol09]

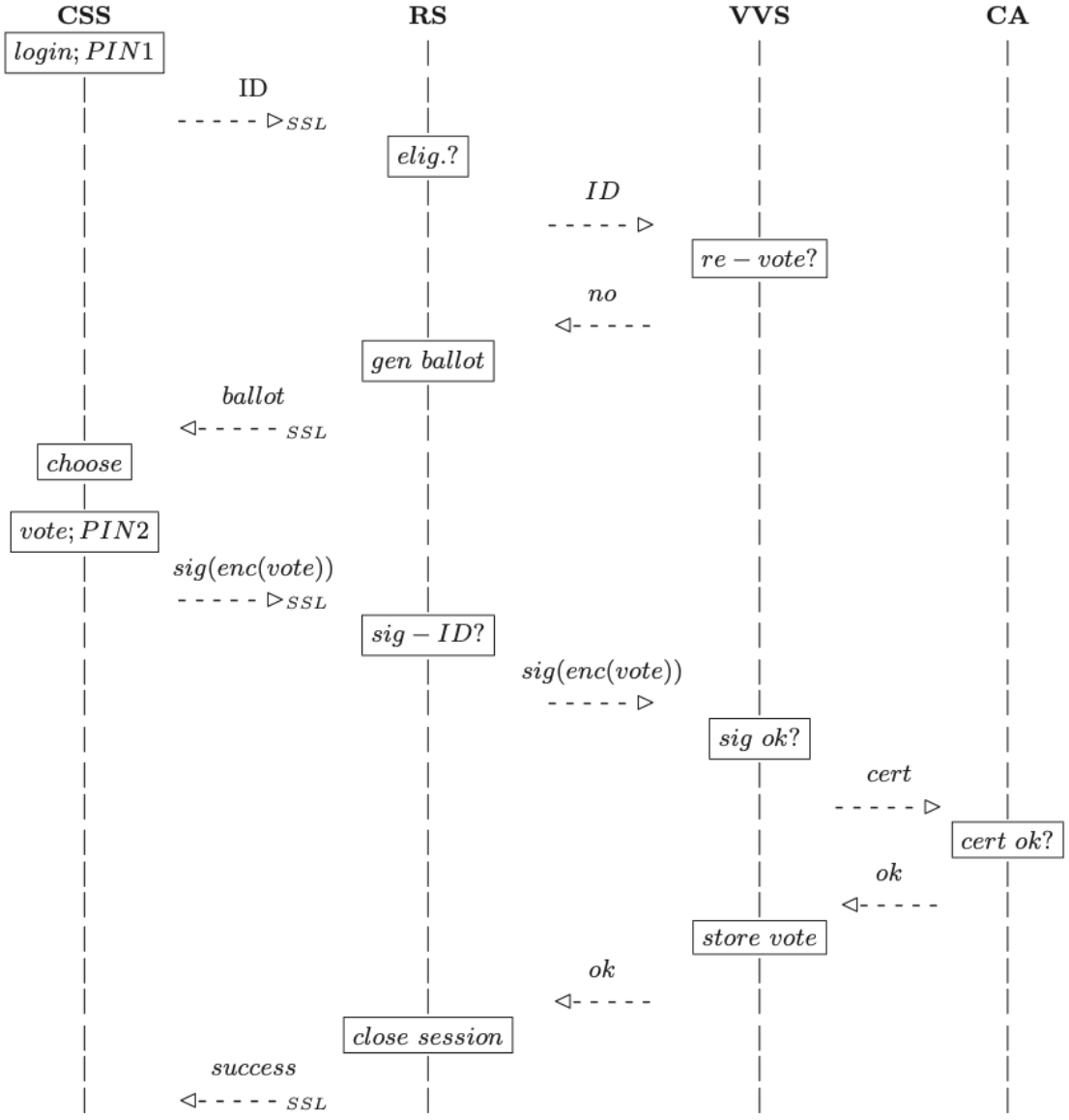


Figura 4.8: Il protocollo di voto implementato nel sistema estone

5. Implementazione

5.1 Installazione

Per dimostrare le funzionalità del sistema Helios, lo abbiamo installato in una macchina virtuale contenente Ubuntu 22.04. Dopo aver aggiornato il sistema e i suoi programmi tramite i comandi `apt update` e `apt upgrade`, e verificato che in esso sia installato Python con il comando

```
python3 --version
```

si procederà con l'installazione del gestore di pacchetti per Python Pip3. Esso consentirà l'installazione e la gestione di pacchetti software di terze parti con caratteristiche e funzionalità non presenti nella libreria standard di Python.

```
sudo apt install python3-pip
```

Non è obbligatorio, ma è una buona pratica creare un ambiente virtuale Python per l'implementazione di Helios, in quanto consente di separare le dipendenze del progetto e di non interferire con altri sistemi sulla stessa macchina.

```
sudo apt install python3-venv
```

5.1.1 Installazione Postgres

Si dovrà poi installare Postgres, un sistema di database relazionale a oggetti che si occuperà della gestione del database associato al progetto:

```
sudo apt update
sudo apt install postgresql
```

e attivarne il servizio con il seguente comando.

```
sudo systemctl start postgresql
```

Per accedere alla command line del database è necessario utilizzare l'utente installato di default dal servizio usando il seguente comando.

```
sudo -i -u postgres
```

Una volta cambiato utente è possibile accedere alla command line, che non è integrata nella bash, ma è di fatto una sua linea di comando dedicata. Il comando per accedervi è:

```
psql postgres
```

Si crea successivamente un utente superuser con nome uguale a quello di login ("ubuntu" nel caso della nostra implementazione, è possibile risalire al proprio tramite il comando `logname`) e un database nominato "helios" di cui è padrone, fatto ciò si esce dalla command line di postgres e si può eseguire il logout come utente di default di postgres.

```
CREATE USER Ubuntu WITH SUPERUSER;
CREATE DATABASE helios WITH OWNER Ubuntu;
\q
```

```
exit
```

Si procederà poi a connettersi al nuovo database "helios" e controllare che sia presente nella lista dei database con il comando `\l`.

```
psql helios
\l
\q
```

Un altro componente richiesto dal sistema di voto Helios è il broker di messaggistica RabbitMQ, installabile tramite il seguente comando.

```
sudo apt install rabbitmq-server
```

5.1.2 Impostazione dell'autenticazione OAuth di Google

Con Helios è possibile fare uso dell'autenticazione OAuth di Google, un sistema di autorizzazione che consente alle applicazioni di accedere ai dati dell'account utente di Google senza dover chiedere all'utente di condividere la propria password.

Per usufruire di questo servizio è necessario disporre di un account Google con la verifica a due fattori attiva e usarlo per accedere a https://console.developers.google.com/apis e selezionare "Credenziali" dalla barra laterale a sinistra. Qui è possibile creare un nuovo progetto, chiamandolo in questo caso "Helios".

Successivamente, si deve creare le credenziali per l'autenticazione OAuth selezionando "Crea Credenziali" e poi "ID client OAuth". È necessario configurare la "Schermata di consenso" selezionando "Esterno" e quindi cliccando su "Crea". Si compilano poi i campi richiesti come il nome dell'applicazione o email di supporto per gli utenti.

Creata la schermata di consenso si selezionerà di nuovo "Credenziali" e poi "Crea Credenziali" per creare le credenziali per il progetto "Helios". Si deve quindi selezionare "Applicazione Web" come tipo di applicazione e dare un nome al ID client OAuth, è possibile riutilizzare il nome "Helios" per semplicità. Nella sezione "Origini JavaScript autorizzate" del ID client OAuth selezionato si inserirà, poiché è nostra intenzione avviare il servizio in locale nella macchina virtuale, "http://localhost:8000", mentre nella sezione "URI di reindirizzamento autorizzati" si deve inserire "http://localhost:8000/auth/after/".

Infine, è necessario copiare il Client ID e il Client Secret che sono stati appena creati per utilizzarli successivamente nelle impostazioni di Helios e abilitare "Google People API" cercandolo nella sezione "API e servizi abilitati".

5.1.3 Impostazione dell'autenticazione OAuth di Github

La versione di Helios disponibile pubblicamente permette diversi metodi di autenticazione, tra cui quello che permette di usare il proprio account di Github. Per usufruire di

questo servizio è necessario disporre di un account Github e accedere alle impostazioni del proprio profilo, selezionando "OAuth apps" è possibile creare un OAuth app. Si dovrà quindi scegliere un nome per l'applicazione e, come per nel procedimento per l'autenticazione OAuth di Google, specificare l'URL dell'applicazione e l'URL di callback (nel caso di implementazione in locale, rispettivamente "http://localhost:8000" "http://localhost:8000/auth/after/").

Dopodiché si dovrà modificare nuovamente il file `settings.py` inserendo il Client ID e il Client Secret nelle rispettive righe (o inserendole nel file `.env`), si dovrà anche modificare la variabile `AUTH_ENABLED_SYSTEMS` aggiungendo "github" così da rendere disponibile la relativa opzione nella schermata di benvenuto.

5.1.4 Preparazione di Helios

È necessario dopodiché scaricare il codice di Helios disponibile su https://github.com/benadida/helios-server.

Una volta estratta e aperta nel terminale si dovrà creare un ambiente virtuale python per il progetto, in questo caso lo si chiamerà "my-venv", il comando sarà quindi:

```
python3 -m venv my-venv
```

Dopo aver creato l'ambiente virtuale, bisogna attivarlo con il comando:

```
source my-venv/bin/activate
```

Prima di procedere all'installazione delle dipendenze, è possibile riscontrare un errore relativo al modulo "wheel". In tal caso, è opportuno installarlo usando il comando:

```
python3 -m pip install wheel
```

Poiché si sta utilizzando PostgreSQL come database, si dovrà installare anche "libpq-dev", una libreria statica per compilare programmi C che si linkano con la libreria libpq per comunicare con un backend per database PostgreSQL, usando il comando:

```
sudo apt install libpq-dev
```

Infine, si installano tutte le dipendenze utilizzando il comando:

```
python3 -m pip install -r requirements.txt
```

Da notare che nel file "requirements.txt" è possibile dover aggiornare alcune dipendenze per garantire la compatibilità con la versione di Python installata. Nel caso di questa installazione il campo "celery" è stato aggiornato alla versione 5.0.0.

Conservazione sicura dei delle credenziali

Questo passaggio, in teoria opzionale, è di grande importanza se si decide di conservare una copia personalizzata del codice copiato dalla repository ufficiale di Helios in una repository GitHub. Il rischio è infatti quello di rendere pubbliche le credenziali di Google, GitHub o del mail server nel file `settings.py`.

Per ovviare a questo problema è possibile inserire le credenziali in un file `.env`. Un file `.env` consente di personalizzare le variabili dell'ambiente di lavoro di Django, esso contiene le variabili d'ambiente personalizzate dell'utente che sovrascrivono le variabili impostate nel file `/etc/environment`. Per fare ciò è necessario l'utilizzo di django-environ, un pacchetto Python che consente di recuperare le variabili d'ambiente dal file `.env`.

```
python3 -m pip install django-environ
```

Creare quindi un file `.env` con le credenziali:

```
GOOGLESECRET=xxxx
GOOGLEID=xxxx
```

In `settings.py` inserire poi:

```
import environ
# Read in secrets from .env file
env = environ.Env()
environ.Env.read_env()
GOOGLESECRET = env('GOOGLESECRET')
GOOGLEID = env('GOOGLEID')
...
# google
GOOGLE_CLIENT_ID = get_from_env('GOOGLE_CLIENT_ID', GOOGLEID)
GOOGLE_CLIENT_SECRET = get_from_env('GOOGLE_CLIENT_SECRET', GOOGLESECRET)
```

Una volta conservate le credenziali in un file diverso da `settings.py`, sarà necessario aggiungere `.env` al file `.gitignore` della repository GitHub così che non venga mai caricato.

Modifica della Timezone

Uno dei problemi riscontrati nell'implementazione del sistema è legato alla gestione dei fusi orari. In particolare, Helios è impostato sull'UTC e non supporta cambiamenti di fuso orario o, in generale, impostazioni che permettano la localizzazione (come ad esempio il cambio di lingua che viene mostrata all'utente). Per risolvere questo problema e applicare il fuso orario del dispositivo in uso sono state apportate le seguenti modifiche.

La modifica della variabile `TIME_ZONE` nel file `settings.py` non produce alcun cambiamento se non accompagnata dall'inserimento della seguente riga:

```
USE_TZ = True
```

In questo modo, si abilita l'utilizzo dei fusi orari nell'applicazione. Successivamente, è necessario modificare il codice in tutti i file che utilizzano la funzione "datetime.datetime.utcnow()" sostituendola con "timezone.now()". Per fare ciò, occorre aggiungere la seguente riga di codice all'inizio dei file interessati (questi sono `helios/stats_views.py`, `helios/models.py`, `helios/management/commands/load_voter_files.py`, `helios_auth/auth_systems/cas.py`, `helios/views.py` e `helios/tests.py`):

```
from django.utils import timezone
```

In questo modo, si garantisce che l'applicazione utilizzi correttamente il fuso orario impostato.

Impostazione del server SMTP

Perché vengano inviate le email che comunichino informazioni e credenziali a elettori e fiduciari, è necessario andare a modificare il file settings.py per personalizzare le configurazioni per l'invio di e-mail tramite Django.

```
# email server
EMAIL_HOST = get_from_env('EMAIL_HOST', 'smtp.gmail.com')
EMAIL_PORT = int(get_from_env('EMAIL_PORT', "587"))
EMAIL_HOST_USER = get_from_env('EMAIL_HOST_USER', 'xx...@gmail.com')
EMAIL_HOST_PASSWORD = get_from_env('EMAIL_HOST_PASSWORD', 'xxxx')
EMAIL_USE_TLS = (get_from_env('EMAIL_USE_TLS', '0') == '0')
```

Modificare il valore di `EMAIL_HOST` in base al proprio client di posta elettronica. I valori accettabili per i client di posta elettronica più comuni sono:

- Gmail: smtp.gmail.com
- Outlook: smtp-mail.outlook.com
- Yahoo: smtp.mail.yahoo.com

Usare un email server di Google Per motivi di sicurezza, dal 30 maggio 2022, Google non supporta più l'uso di app di terze parti o dispositivi che chiedono di accedere all'Account Google utilizzando solo il nome utente e la password, perciò, perché si possa usare il server SMTP di un account Google è necessario generare una password apposita che consenta alle app di terze parti di accedere.

Per accedere alla sezione "Password per le app", si deve prima visitare la pagina Account Google e selezionare la sezione "Sicurezza". Se questa opzione non è disponibile, ci sono diverse possibili ragioni, come la mancata configurazione della verifica in due passaggi, la configurazione della verifica in due passaggi solo per i token di sicurezza, l'utilizzo di un account di lavoro, accademico o di un'altra organizzazione. Per generare una password per l'app, si dovrà quindi selezionare il tipo di applicazione che avrà accesso all'account (in questo caso "Posta") e il dispositivo utilizzato (in questo caso "Altro), e quindi premere "Genera". Verrà quindi mostrata la password apposita da assegnare alla variabile `EMAIL_HOST_PASSWORD`,

5.1.5 Avvio del server

Prima di eseguire l'applicazione Helios, è necessario avviare il servizio Celery in un terminale separato. Celery è un sistema di elaborazione distribuito in background che viene spesso utilizzato in combinazione con Django per eseguire attività asincrone.

Per avviare Celery, aprire un nuovo terminale e accedere alla cartella del progetto Helios e attivare l'ambiente virtuale appena creato per evitare conflitti con le dipendenze di altri progetti. Quindi, eseguire il comando seguente per avviare Celery:

```
cd /percorso/alla/cartella/helios
source my-venv/bin/activate
celery -A helios worker -l INFO
```

Questo comando avvia il processo worker di Celery e lo imposta per registrare i messaggi di log con livello di registrazione INFO.

Bisogna tornare al primo terminale per eseguire l'applicazione Helios. Per prima cosa, eseguire il comando `makemigrations`. Le migrazioni sono il modo in cui Django propaga le modifiche apportate ai modelli (aggiunta di un campo, cancellazione di un modello, ecc.) nello schema del database. `makemigrations` è il comando responsabile della creazione di nuove migrazioni basate sulle modifiche apportate ai modelli.

```
python3 manage.py makemigrations
```

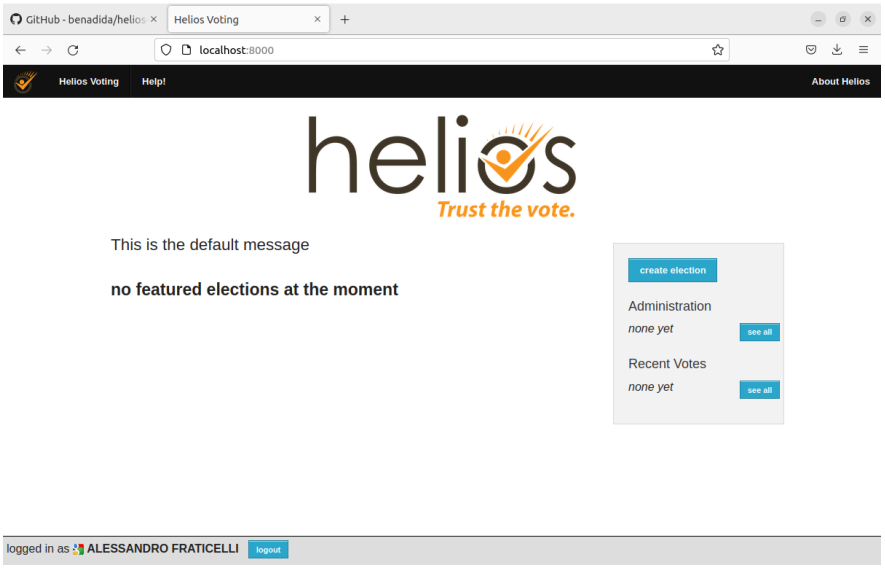


Figura 5.1: Homepage di Helios

Successivamente, eseguire lo script reset.sh per inizializzare il database:

```
./reset.sh
```

Infine, eseguire il comando seguente per avviare il server web di sviluppo di Django:

```
python3 manage.py runserver
```

Questo comando avvia il server web di sviluppo di Django sulla porta predefinita 8000. Ora è possibile accedere all'applicazione Helios tramite il browser visitando l'indirizzo http://localhost:8000.

5.2 Creazione di un'elezione

Per creare e amministrare un'elezione, è necessario effettuare il login utilizzando l'autenticazione di terze parti come Google o GitHub. Helios richiede e conserva solo le informazioni di base necessarie per autenticare l'utente durante l'accesso successivo, tra cui: il nome, l'indirizzo e-mail e l'ID utente. L'amministratore potrà utilizzare queste informazioni per inviare messaggi all'utente quando l'elezione richiederà attenzione.

Come amministratore dell'elezione, si ha il potere di designare chi può votare, quando l'elezione inizia, quando l'elezione termina e quando i risultati vengono pubblicati. Oltre a questo, l'amministratore non ha alcun potere al di là di quello degli elettori dell'elezione.

Sulla homepage di Helios, dopo aver effettuato il login, il link "Create Election" porta a una pagina contenente un modulo che definisce le caratteristiche principali dell'elezione.

Una volta completati i campi del modulo, premendo "Next" l'elezione viene aggiunta, sarà possibile tornare indietro e modificarla fino a quando non si sceglie di congelarla, e viene mostrata la pagina principale dell'elezione.

Oltre al nome dell'elezione e l'username del creatore dell'elezione, viene mostrata un'informazione riguardante la presenza dell'elezione sulla pagina principale del servizio. Perché un utente possa presentare delle elezioni sulla homepage, il campo utente

Campo	Descrizione
Nome breve	Non può contenere spazi, farà parte dell'URL dell'elezione, ad esempio my-club-2010
Nome	Il nome dell'elezione
Descrizione	Appare sulla pagina principale per questa elezione / referendum
Tipo	Elezione / referendum;
Uso di alias per gli elettori	Se selezionato, le identità degli elettori saranno sostituite con alias, ad esempio "V12", nel centro di monitoraggio dei voti
Randomizza l'ordine delle risposte	Abilitare questa opzione se si desidera che le risposte alle domande appaiano in ordine casuale per ogni elettore
Privato?	Un'elezione privata è visibile solo agli elettori registrati.
Indirizzo email di supporto	Un indirizzo email a cui gli elettori possono rivolgersi in caso di problemi, di default è l'indirizzo associato all'email dell'admin.
Inizio Voto	Data e ora dell'inizio della votazione
Termine Voto	Data e ora della fine della votazione

`admin_p` nella tabella del database `helios_auth_user` deve essere impostato su True. Per apportare questa modifica occorre connettersi al suddetto database in un terminale ed eseguire la query SQL:

```
UPDATE helios_auth_user SET admin_p = 't' WHERE user_id = [user_id];
```

La pagina dell'elezione mostra, oltre il resto delle informazioni inserite dal creatore dell'elezione, l'indicazione contenente il passo successivo che l'amministratore deve eseguire per condurre l'elezione. Da questa pagina si possono quindi gestire:

- I quesiti per cui votare
- Gli elettori e le schede elettorali
- I fiduciari dell'elezione

5.2.1 Quesiti

Questa sezione permette di aggiungere i quesiti da sottoporre agli elettori. Oltre al quesito stesso e alle possibili scelte, a ciascuna di queste ultime è possibile associare un link (utile ad esempio a fornire ulteriori informazioni su un candidato). Per ogni domanda inserita è possibile specificare quante preferenze è lecito esprimere, selezionando 0 come minimo di "risposte" si dà la possibilità di scheda bianca. È infine possibile stabilire se al termine dell'elezione i risultati debbano essere espressi in numero assoluto o in numero relativo (in percentuale) di voti.

5.2.2 Registrazione dell'elettore

Nella sezione "Votanti e schede" l'amministratore dell'elezione definisce il gruppo di elettori idoneo al voto. Helios offre due opzioni principali:

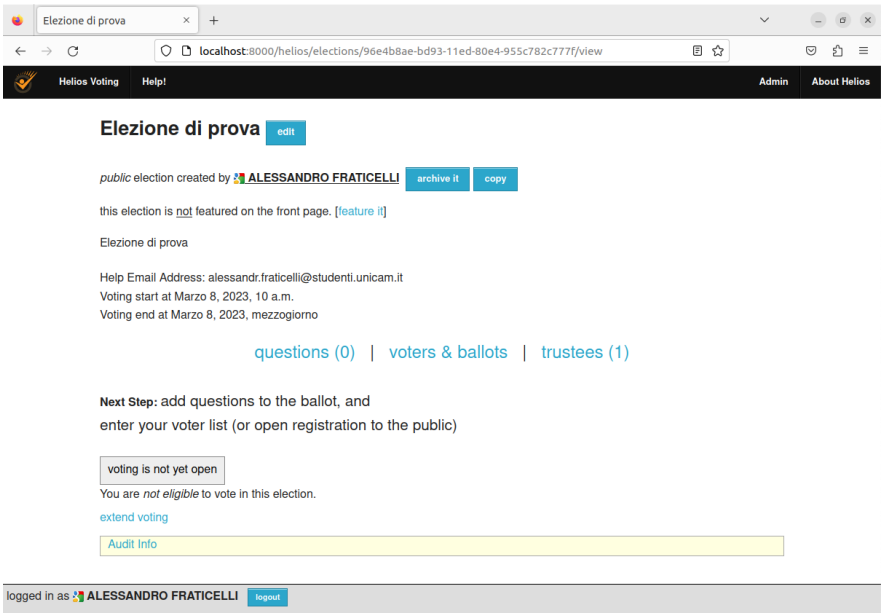


Figura 5.2: Pagina principale dell'elezione

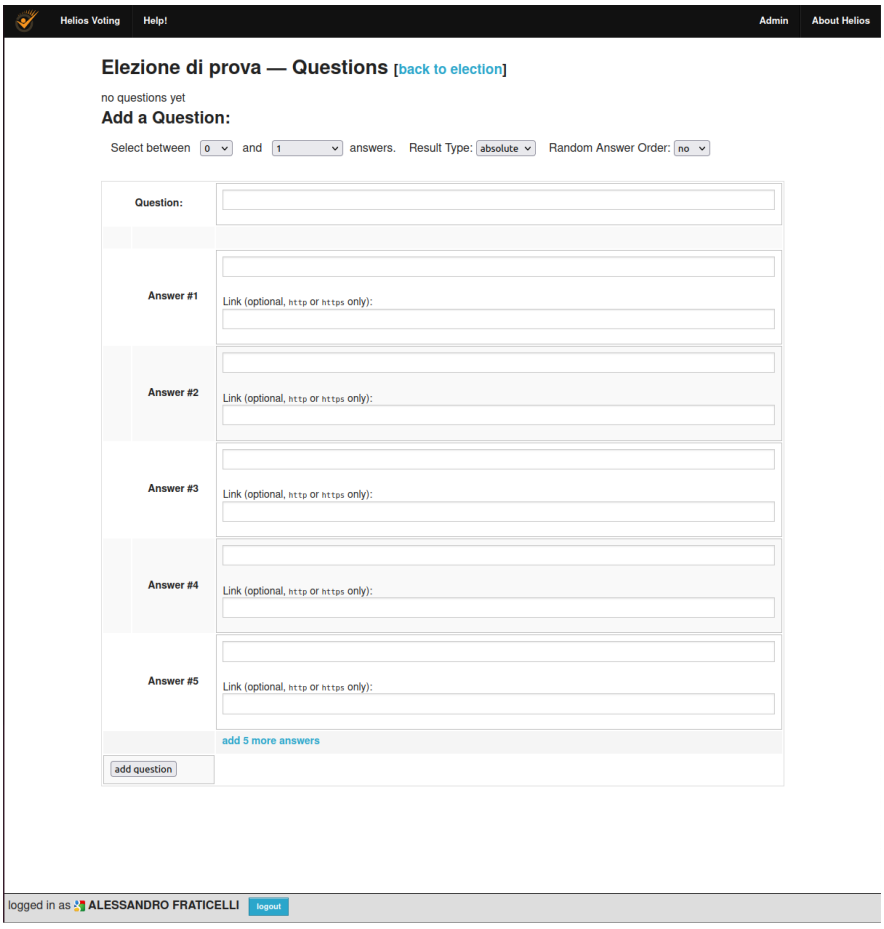


Figura 5.3: Pagina dei quesiti dell'elezione

- consentire a qualsiasi membro del pubblico di votare;
- caricare un elenco di elettori idonei

La prima opzione permette a qualsiasi utente di partecipare all'elezione accedendo servendosi di una delle opzioni di autenticazione di Helios (Google, Facebook, ecc.) prima di esprimere il proprio voto. Questa soluzione permette però a una singola persona di votare più volte utilizzando altrettanti account.

Se l'amministratore dell'elezione sceglie di definire l'idoneità al voto utilizzando un elenco di elettori idonei contenuto in un file csv, questo deve essere caricato al momento della creazione dell'elezione e non può essere modificato una volta avviata. Il file csv va preparato seguendo i template:

`password, <id_univoco>, <email>, <nome completo>`

oppure

`github,<nome_utente>`

Questa opzione non permette infatti di immettere nell'elenco degli account di Google, qualora lo si facesse, il campo riguardante gli indirizzi email di questi ultimi risulterebbe vuoto, rendendo impossibile l'invio delle mail che permettono la partecipazione all'elezione.

L'applicazione Helios non fornisce la possibilità di registrarsi come elettore idoneo. Questo significa che solo gli elettori inclusi nell'elenco fornito dall'amministratore dell'elezione possono votare. Questa opzione è utile per le elezioni limitate solo ad un gruppo specifico di persone, come i membri di un'organizzazione o un'associazione.

È da notare che gli utenti con password non possono partecipare alle elezioni a registrazione aperta. Se si prova infatti a impostare "password" nelle variabili di ambiente `AUTH_ENABLED_SYSTEMS` e `AUTH_DEFAULT_SYSTEM` in `settings.py`, nella fase finale del processo di voto di un'elezione aperta, all'elettore vengono richiesti il suo ID e password, tuttavia, la vista responsabile cerca solo di confrontare le credenziali fornite con l'elenco degli elettori, che non è stato creato.

Sembra sia una scelta progettuale che gli utenti non autenticati tramite un servizio di terze parti, cioè utenti con password memorizzate nel database, non possano partecipare alle elezioni a registrazione aperta.

5.2.3 I fiduciari

Nella sezione "Trustees" è possibile gestire i fiduciari dell'elezione, essi sono le figure alle quali ci si affida per mantenere la riservatezza dei voti e per partecipare al processo di decodifica del conteggio. Per impostazione predefinita, il server Helios è configurato per fungere da fiduciario, ma può essere revocato e può essere aggiunto un numero qualsiasi di nuovi fiduciari. I fiduciari hanno un ruolo molto delicato. Ognuno di loro deve generare una coppia di chiavi, composta da una componente pubblica e una segreta. La chiave pubblica deve essere caricata sul server Helios, mentre la chiave segreta deve essere mantenuta segreta. Al momento del conteggio, tutti i fiduciari devono partecipare alla decodifica dei risultati elettorali, utilizzando la propria chiave segreta. Come corollario di questa procedura di conteggio, se manca un fiduciario, non ci sarà modo di ottenere il risultato dell'elezione ed è probabile che l'elezione debba essere riavviata. In letteratura sono state proposte procedure più robuste, in grado di tollerare un numero limitato di fiduciari mancanti[Ped91],[Gen+07], ma sono molto più difficili da utilizzare, in quanto richiedono più cicli di interazioni tra i fiduciari,

motivo per cui Helios si affida a questa procedura a passaggio singolo che massimizza la riservatezza.

In pratica, i fiduciari vengono registrati attraverso questa pagina, in modo simile al processo di registrazione degli elettori. Ricevono un'e-mail con un URL, dal quale accedono a una pagina web che consente loro di generare la propria coppia di chiavi, caricare la componente pubblica, salvare quella segreta e verificare se l'intero processo è andato a buon fine.

5.2.4 Apertura dell'elezione

Dopo aver impostato le domande, definito il gruppo degli elettori idonei e scelto i fiduciari, è possibile "congelare" l'elezione e renderla disponibile alla votazione. Una volta completata questa azione, i quesiti, le opzioni di voto e i fiduciari non possono più essere modificati. Se l'elezione che sta per essere avviata è di tipo chiuso ed è stato caricato un elenco di elettori, è possibile, nella sezione Elettori e schede, inviare un'e-mail a tutti gli elettori idonei fornendo a ciascuno il proprio ID di voto e la propria password.

5.2.5 Informazioni per la verifica

Sotto la pagina principale dell'elezione si trova un riquadro contenente le informazioni utili per la verifica dell'elezione. Esso contiene il link diretto all'elezione stessa, la stringa fingerprint univoca dell'elezione, il link al tracking ballot center e il link alla lista di schede verificate. Il ballot tracking center è la pagina dove è possibile monitorare i voti espressi e i loro relativi smart ballot tracker. La sezione delle schede verificate comprende tutte le schede che gli elettori hanno inserito nel single ballot verifier, verificando la loro correttezza e pubblicandole.

5.3 Verifica del singolo voto

Nel caso in cui un elettore desideri verificare che la propria intenzione di voto sia stata registrata correttamente e garantirsi che né il browser utilizzato per votare né il server che ospita l'elezione siano compromessi, egli può decriptare il proprio voto dopo averlo espresso, al fine di verificarne la correttezza.

Dopo aver scelto l'opzione per un quesito, nella sezione di revisione del proprio voto, cliccando sul bottone "Spoil & Audit" il codice JavaScript rivela il testo di un file JSON che identifica la scheda, questa struttura dati contiene una scheda in chiaro, il suo testo cifrato, la casualità usata per cifrarla e l'ID dell'elezione. L'elettore può salvare questi dati su disco ed eseguire il proprio codice per assicurarsi che la crittografia sia corretta, oppure può utilizzare il programma Python Ballot Encryption Verification (BEV) fornito da Helios.

Il BEV ottiene i parametri dell'elezione in base all'ID e produce:

- l'hash dell'elezione, che l'elettore può verificare con quello visualizzato nella "voting booth",
- l'hash del testo cifrato, che l'elettore può verificare rispetto alla ricevuta ottenuta prima di richiedere la verifica,
- il testo in chiaro verificato della scheda.

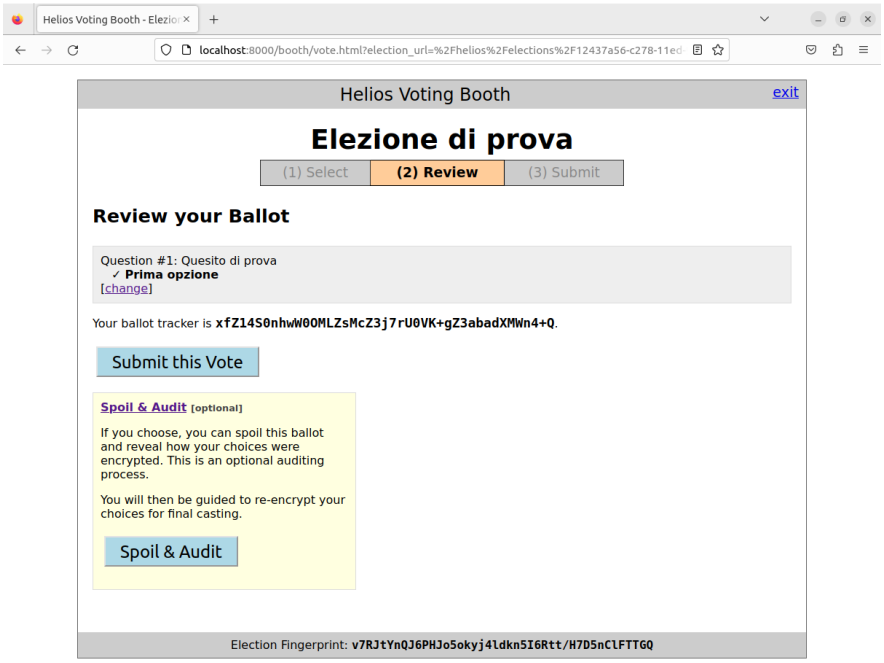


Figura 5.4: Schermata del Voting Booth di Helios che offre la scelta di votare o verificare il proprio voto.

Una volta che l'elettore sceglie di verificare la propria scheda e le informazioni di verifica vengono visualizzate, il codice JavaScript cancella le strutture dati crittografate della scheda e riporta l'elettore alla schermata di conferma, dove può aggiornare le sue scelte o scegliere di sigillare nuovamente le sue opzioni con una diversa casualità e quindi un diverso testo cifrato.

L'elettore dovrà quindi copiare le informazioni visualizzate, cliccare sul link "Ballot Verifier" e incollare le informazioni copiate nella pagina "Helios Single-Ballot Verifier" che si apre. Facendo clic sul pulsante "Verifica", si ottengono i risultati in fondo alla pagina. Poiché "Helios Single-Ballot Verifier" è un'applicazione indipendente con una propria finestra, l'elettore può tornare all'applicazione principale di scrutinio in qualsiasi momento (ad esempio chiudendo l'applicazione di verifica). Per continuare il processo di voto, l'elettore fa clic sul pulsante "torna al voto". Dopodiché dovrà criptare nuovamente la scheda. L'elettore può quindi decidere se esprimere il voto criptato o verificarlo nuovamente.

5.4 Scrutinio

Una volta terminato il turno di votazione, il server Helios calcola una crittografia del conteggio elettorale, aggregando l'ultima scheda valida ricevuta da ciascun elettore, utilizzando la proprietà omomorfica dello schema di crittografia utilizzato per proteggere i voti.

I fiduciari sono quindi invitati a decifrare questo conteggio. Collegandosi all'interfaccia dei fiduciari di Helios tramite il proprio browser, questi possono scaricare il conteggio criptato, vederne l'impronta digitale, caricare la propria chiave privata nel browser, ese-

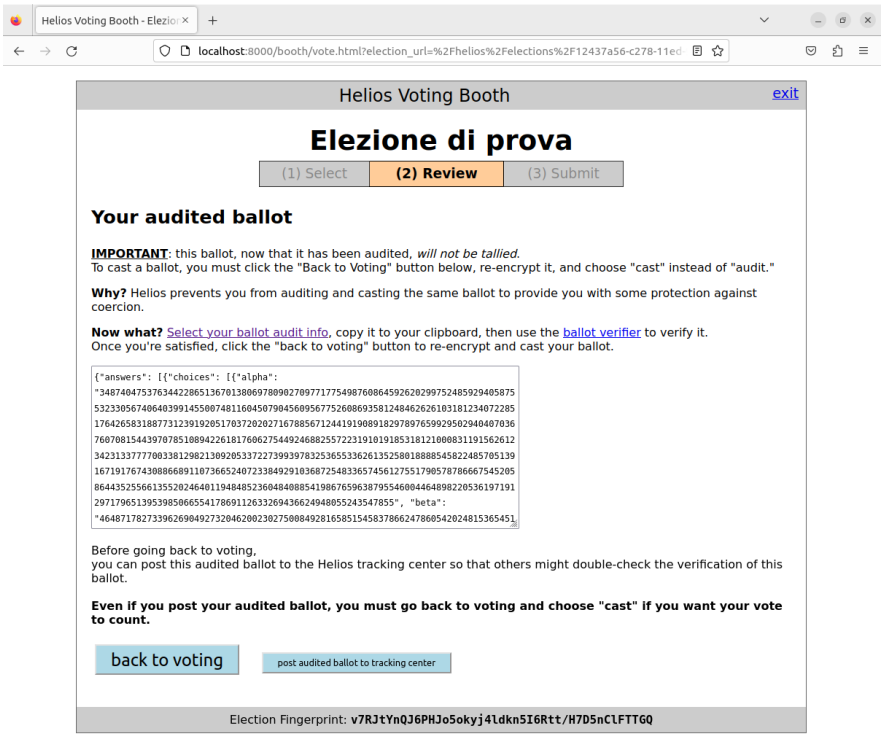


Figura 5.5: Schermata del Voting Booth di Helios che mostra il file JSON associato alla scheda da verificare.

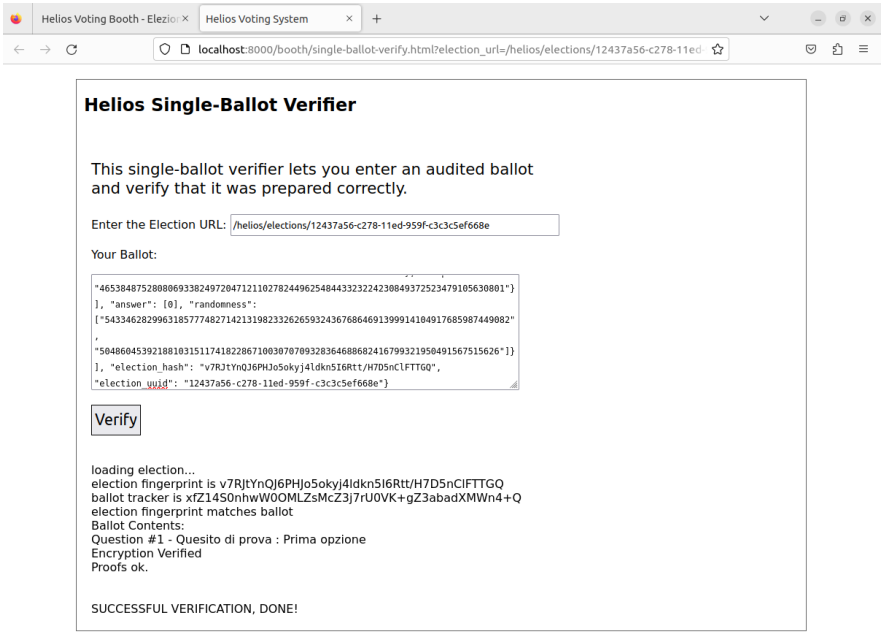


Figura 5.6: Schermata di successo del Helios Single-Ballot Verifier.

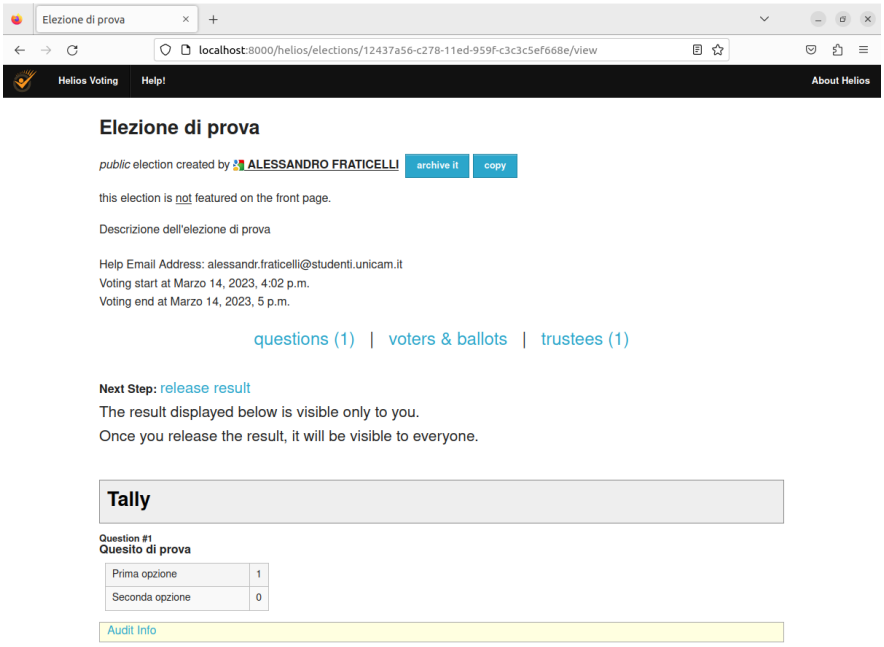


Figura 5.7: Schermata dell'amministratore a elezione conclusa e scrutinio completato

guire la decodifica parziale del conteggio e caricare il risultato di questa decodifica sul server Helios. Come per la preparazione della scheda elettorale, non è necessario utilizzare l'interfaccia web di Helios per questo scopo: un altro software, gestito in modo indipendente, potrebbe eseguire le stesse operazioni e inviare la decodifica parziale del conteggio.

Non è possibile ottenere informazioni sul conteggio finché uno dei fiduciari non ha inviato la decodifica parziale del conteggio. Ma non appena tutti i fiduciari hanno completato il loro compito, i server Helios combinano le decifrazioni parziali nel conteggio elettorale completo e lo rendono disponibile. Anche in questo caso, la combinazione delle decifrazioni parziali è un'operazione pubblica.

Quando il conteggio è completo, tutte le informazioni necessarie per la verifica delle elezioni sono disponibili sul server Helios. Inoltre, le chiavi segrete conservate dai fiduciari possono essere distrutte: non sono più utili per la verifica e questa distruzione riduce i rischi di compromissione delle chiavi in futuro.

6. Conclusioni

In questa tesi si è presentato il tema del voto elettronico, focalizzandosi sugli standard che essi devono raggiungere in tema di sicurezza, sulla descrizione di alcuni esempi di sistemi esistenti e sulla valutazione dell'implementazione di uno di essi.

Nonostante gli sforzi per sviluppare sistemi di voto elettronico sicuri e affidabili, tuttavia, la fiducia degli elettori in questi sistemi rimane un aspetto critico e la mancanza di standard comuni a livello globale limita la diffusione e l'adozione di soluzioni di voto elettronico. Il diritto di voto è una parte del processo democratico, che rimane profondamente radicata nelle moderne costituzioni. Il voto elettronico, a differenza di altre transazioni elettroniche, sarà accettabile solo se garantirà il rispetto di tutti i principi costituzionali pertinenti. Inoltre, un sistema di voto elettronico dovrebbe essere implementato in un contesto che garantisca parità di accesso all'infrastruttura tecnologica sottostante, che dovrebbe essere aperta, facile da usare, interattiva e sicura, al fine di consentire ai cittadini di partecipare alla vita politica e di avere un impatto diretto su di essa [MGK02][Mer00].

L'analisi condotta ha messo in evidenza l'importanza della sicurezza informatica per la realizzazione di un sistema di voto elettronico affidabile e ha sottolineato l'importanza degli standard di sicurezza per garantire l'integrità del voto. Le misure di sicurezza impiegate nei sistemi presentati risultano difficilmente scalabili e rischiano di non garantire la correttezza dell'elezione e la segretezza del voto.

Ad esempio di ciò, nel maggio 2020, diverse agenzie governative statunitensi hanno comunicato agli Stati che, in relazione al voto elettronico impiegato nel paese, "le tecnologie per la votazione elettronica sono ad alto rischio anche con i controlli [di mitigazione del rischio] in atto" e che gli attacchi "potrebbero essere condotti da qualsiasi parte del mondo, ad alti volumi, e potrebbero compromettere la riservatezza, l'integrità della scheda e/o interrompere la disponibilità della scheda" [Vol20].

Tuttavia, l'implementazione del sistema di voto Helios ha dimostrato che in contesti di dimensioni medio-piccole, dove il pericolo di brogli è relativamente basso, è possibile utilizzare i sistemi di voto elettronico, a patto che siano adottate misure di sicurezza adeguate e sia effettuata un'attenta valutazione delle esigenze specifiche del contesto.

In conclusione, il voto elettronico rappresenta un'opportunità per migliorare i processi democratici, ma richiede un approccio oculato e una continua attenzione alla sicurezza.

Ringraziamenti

In primo luogo, vorrei ringraziare il Prof. Fausto Marcantoni per avermi concesso l'opportunità di svolgere la mia tesi con lui.

Un ringraziamento particolare va alla mia famiglia, in particolare ai miei genitori, che mi hanno sempre spronato e sostenuto.

Desidero poi ringraziare i miei amici e compagni di corso per aver condiviso con me momenti di studio, di confronto e di svago.

Infine, desidero ringraziare tutti coloro che, in modo diretto o indiretto, hanno contribuito alla realizzazione della mia tesi.

Bibliografia

[Ace+14] Claudia Z. Acemyan et al. «Usability of Voter Verifiable, End-to-end Voting Systems: Baseline Data for Helios, Prêt à Voter, and Scantegrity II». In: *2014 Electronic Voting Technology Workshop/Workshop on Trustworthy Elections (EVT/WOTE 14)*. San Diego, CA: USENIX Association, ago. 2014.

[Adi08] Ben Adida. «Helios: Web-based Open-Audit Voting». In: *USENIX Security Symposium*. 2008.

[Adi+09] Ben Adida et al. «Electing a University President Using Open-Audit Voting: Analysis of Real-World Use of Helios». In: *Proceedings of the 2009 Conference on Electronic Voting Technology/Workshop on Trustworthy Elections*. EVT/WOTE'09. USA: USENIX Association, 2009, p. 10.

[Aga+21] Vincenzo Agate et al. «SecureBallot: A secure open source e-Voting system». In: *Journal of Network and Computer Applications* 191 (2021). DOI: https://doi.org/10.1016/j.jnca.2021.103165. URL: https://www.sciencedirect.com/science/article/pii/S1084804521001776.

[Agg+19] Shubhani Aggarwal et al. «Blockchain for smart communities: Applications, challenges and opportunities». In: *Journal of Network and Computer Applications* 144 (2019), pp. 13–48. ISSN: 1084-8045. DOI: https://doi.org/10.1016/j.jnca.2019.06.018. URL: https://www.sciencedirect.com/science/article/pii/S1084804519302231.

[AHT09] R. Michael Alvarez, Thad E. Hall e Alexander H. Trechsel. «Internet Voting in Comparative Perspective: The Case of Estonia». In: *PS: Political Science & Politics* 42.3 (2009), 497–505. DOI: 10.1017/S1049096509090787.

[AMP19] Chrisanthi Avgerou, Silvia Masiero e Angeliki Poulymenakou. «Trusting e-voting amid experiences of electoral malpractice: The case of Indian elections». In: *Journal of Information Technology* 34.3 (2019), pp. 263–289. DOI: 10.1177/0268396218816199.

[Ant+07] A. Antoniou et al. «A Trust-Centered Approach for Building E-Voting Systems». In: *Electronic Government*. A cura di Maria A. Wimmer, Jochen Scholl e Åke Grönlund. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 366–377. ISBN: 978-3-540-74444-3.

[AP14] Himanshu Agarwal e G. Pandey. «A Secure E-Election System». In: mag. 2014, pp. 1–4. ISBN: 978-1-4799-4441-5. DOI: 10.1109/ICISA.2014.6847335.

[Bär+08] Michael Bär et al. «Real world experiences with Bingo Voting and a comparison of usability». In: *Workshop on Trustworthy Elections (WOTE), volume 2008*. 2008.

[BC08] France Bélanger e Lemuria Carter. «Trust and risk in e-government adoption». In: *The Journal of Strategic Information Systems* 17.2 (2008). eGovernment Strategies: ICT innovation in international public sector contexts, pp. 165–176. ISSN: 0963-8687. DOI: https://doi.org/10.1016/j.jsis.2007.12.002. URL: https://www.sciencedirect.com/science/article/pii/S0963868707000637.

[Bed+03] Ben Bederson et al. «Electronic Voting System Usability Issues». In: gen. 2003, pp. 145–152. DOI: 10.1145/642611.642638.

[Ben06] Josh Benaloh. «Simple Verifiable Elections». In: *Proceedings of the USENIX/Accurate Electronic Voting Technology Workshop 2006 on Electronic Voting Technology Workshop*. EVT'06. Vancouver, B.C., Canada: USENIX Association, 2006, p. 5.

[Ben87] Josh Benaloh. «Verifiable Secret-Ballot Elections». Tesi di dott. Set. 1987.

[BGE07] Michael D. Byrne, Kristen K. Greene e Sarah P. Everett. «Usability of Voting Systems: Baseline Data for Paper, Punch Cards, and Lever Machines». In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '07. Association for Computing Machinery, 2007, 171–180. ISBN: 9781595935939. DOI: 10.1145/1240624.1240653.

[BGP11] Philippe Bulens, Damien Giry e Olivier Pereira. «Running Mixnet-Based Elections with Helios». In: *2011 Electronic Voting Technology Workshop/Workshop on Trustworthy Elections (EVT/WOTE 11)*. San Francisco, CA: USENIX Association, ago. 2011. URL: https://www.usenix.org/conference/evtwote-11/running-mixnet-based-elections-helios.

[BGR12] Sergiu Bursuc, Gurchetan S. Grewal e Mark D. Ryan. «Trivitas: Voters Directly Verifying Votes». In: *E-Voting and Identity*. A cura di Aggelos Kiayias e Helger Lipmaa. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 190–207. ISBN: 978-3-642-32747-6.

[BMQR07] Jens-Matthias Bohli, Jörn Müller-Quade e Stefan Röhrich. «Bingo Voting: Secure and Coercion-Free Voting Using a Trusted Random Number Generator». In: *IACR Cryptol. ePrint Arch.* 2007 (2007), p. 162.

[Boh+09] Jens-Matthias Bohli et al. «Enhancing Electronic Voting Machines on the Example of Bingo Voting». In: *IEEE Transactions on Information Forensics and Security* 4.4 (2009), pp. 745–750. DOI: 10.1109/TIFS.2009.2033755.

[BPW12] David Bernhard, Olivier Pereira e Bogdan Warinschi. «How Not to Prove Yourself: Pitfalls of the Fiat-Shamir Heuristic and Applications to Helios». In: *Proceedings of the 18th International Conference on The Theory and Application of Cryptology and Information Security*. ASIACRYPT'12. Berlin, Heidelberg: Springer-Verlag, 2012, 626–643. ISBN: 9783642349607. DOI: 10.1007/978-3-642-34961-4_38. URL: https://doi.org/10.1007/978-3-642-34961-4_38.

[BT94] Josh Benaloh e Dwight Tuinstra. ≪Receipt-Free Secret-Ballot Elections (Extended Abstract)≫. In: *Proceedings of the Twenty-Sixth Annual ACM Symposium on Theory of Computing*. STOC '94. New York, NY, USA: Association for Computing Machinery, 1994, 544–553. ISBN: 0897916638. DOI: 10.1145/195058.195407.

[But00] R. et al. Buttler. ≪A national-scale authentication infrastructure≫. In: *Computer* 33.2 (feb. 2000), pp. 60–65.

[BY86] Josh C Benaloh e Moti Yung. ≪Distributing the Power of a Government to Enhance the Privacy of Voters≫. In: *Proceedings of the Fifth Annual ACM Symposium on Principles of Distributed Computing*. PODC '86. New York, NY, USA: Association for Computing Machinery, 1986, 52–62. ISBN: 0897911989. DOI: 10.1145/10590.10595.

[Car+10] Richard Carback et al. ≪Scantegrity II Municipal Election at Takoma Park: The First E2E Binding Governmental Election with Ballot Privacy≫. In: *19th USENIX Security Symposium (USENIX Security 10)*. Washington, DC: USENIX Association, ago. 2010. URL: https://www.usenix.org/conference/usenixsecurity10/scantegrity-ii-municipal-election-takoma-park-first-e2e-binding.

[CB05] Lemuria Carter e France Belanger. ≪The Utilization of E-Government Services: Citizen Trust, Innovation and Acceptance Factors≫. In: *Inf. Syst. J.* 15 (gen. 2005), pp. 5–25. DOI: 10.1111/j.1365-2575.2005.00183.x.

[CCM08] Michael R. Clarkson, Stephen Chong e Andrew C. Myers. ≪Civitas: Toward a Secure Voting System≫. In: *2008 IEEE Symposium on Security and Privacy (sp 2008)*. 2008, pp. 354–368. DOI: 10.1109/SP.2008.32.

[CF85] Josh D. Cohen e Michael J. Fischer. ≪A robust and verifiable cryptographically secure election scheme≫. In: *26th Annual Symposium on Foundations of Computer Science (sfcs 1985)*. 1985, pp. 372–382. DOI: 10.1109/SFCS.1985.2.

[CG09] Thomas E. Carroll e Daniel Grosu. ≪A secure and anonymous voter-controlled election scheme≫. In: *Journal of Network and Computer Applications* 32.3 (2009), pp. 599–606. ISSN: 1084-8045. DOI: https://doi.org/10.1016/j.jnca.2008.07.010. URL: https://www.sciencedirect.com/science/article/pii/S1084804508000830.

[CH11] Jeremy Clark e Urs Hengartner. *Selections: Internet Voting with Over-the-Shoulder Coercion-Resistance*. Cryptology ePrint Archive, Paper 2011/166. https://eprint.iacr.org/2011/166. 2011. URL: https://eprint.iacr.org/2011/166.

[Cha04] David Chaum. ≪Secret-ballot receipts: True voter-verifiable elections≫. In: *IEEE Security & Privacy* 2.1 (2004), pp. 38–47. DOI: 10.1109/MSECP.2004.1264852.

[Cha+08] David Chaum et al. ≪Scantegrity II: End-to-End Verifiability for Optical Scan Election Systems Using Invisible Ink Confirmation Codes≫. In: EVT'08. USENIX Association, 2008.

[Cha81] David L. Chaum. «Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms». In: *Commun. ACM* 24.2 (feb. 1981), 84–90. ISSN: 0001-0782. DOI: 10.1145/358549.358563. URL: https://doi.org/10.1145/358549.358563.

[Cha88] David Chaum. «Elections with Unconditionally-Secret Ballots and Disruption Equivalent to Breaking RSA». In: *Advances in Cryptology - EUROCRYPT '88, Workshop on the Theory and Application of of Cryptographic Techniques, Davos, Switzerland, May 25-27, 1988, Proceedings*. Vol. 330. Lecture Notes in Computer Science. Springer, 1988, pp. 177–182. DOI: 10.1007/3-540-45961-8_15.

[Cra+96] Ronald Cramer et al. «Multi-Autority Secret-Ballot Elections with Linear Work». In: *International Conference on the Theory and Application of Cryptographic Techniques*. Vol. 1070. Berlin, 1996, pp. 72–83.

[CRS05] David Chaum, Peter Y. A. Ryan e Steve Schneider. «A Practical Voter-Verifiable Election Scheme». In: ESORICS'05. Berlin, Heidelberg: Springer-Verlag, 2005, 118–139. ISBN: 3540289631. DOI: 10.1007/11555827_8. URL: https://doi.org/10.1007/11555827_8.

[CS10] Veronique Cortier e Ben Smyth. *Attacking and fixing Helios: An analysis of ballot secrecy*. Cryptology ePrint Archive, Paper 2010/625. https://eprint.iacr.org/2010/625. 2010. URL: https://eprint.iacr.org/2010/625.

[DKR06] Stephanie Delaune, Steve Kremer e Mark D. Ryan. «Coercion-Resistance and Receipt-Freeness in Electronic Voting». In: *Proceedings of the 19th IEEE Computer Security Foundations Workshop (CSFW'06)*. IEEE Computer Society Press, lug. 2006, pp. 28–39. DOI: 10.1109/CSFW.2006.8.

[DL15] Jérôme Dossogne e Frédéric Lafitte. «Blinded Additively Homomorphic Encryption Schemes for Self-Tallying Voting». In: *J. Inf. Secur. Appl.* 22.C (giu. 2015), 40–53. ISSN: 2214-2126. DOI: 10.1016/j.jisa.2014.07.002. URL: https://doi.org/10.1016/j.jisa.2014.07.002.

[DVDGA12] Denise Demirel, Jeroen Van De Graaf e Roberto Araújo. «Improving Helios with Everlasting Privacy towards the Public». In: *Proceedings of the 2012 International Conference on Electronic Voting Technology/-Workshop on Trustworthy Elections*. EVT/WOTE'12. USA: USENIX Association, 2012, p. 8.

[ED10] Saghar Estehghari e Yvo Desmedt. «Exploiting the Client Vulnerabilities in Internet E-voting Systems: Hacking Helios 2.0 as an Example». In: *2010 Electronic Voting Technology Workshop/ Workshop on Trustworthy Elections (EVT/WOTE 10)*. Washington, DC: USENIX Association, ago. 2010. URL: https://www.usenix.org/conference/evtwote-10/exploiting-client-vulnerabilities-internet-e-voting-systems-hacking-helios-20.

[Eur] *Lex - L24226A - en - EUR-Lex*. 2022. URL: https://eur-lex.europa.eu/IT/legal-content/summary/eeurope-2002.html.

[Eur10] Directorate Council of Europe. *E-voting handbook - key steps in the implementation of E-enabled elections*. Strasbourg, France: Council of Europe, nov. 2010.

[Fra+11] Richard Frankland et al. «Side-channels and eVoting machine security: Identifying vulnerabilities and defining requirements». In: *2011 International Workshop on Requirements Engineering for Electronic Voting Systems*. 2011, pp. 37–46. DOI: 10.1109/REVOTE.2011.6045914.

[FS00] California Internet Voting Task Force e California. Secretary of State. *A Report on the Feasibility of Internet Voting*. Bill Jones, Secretary of States, 2000. URL: https://books.google.it/books?id=wKitoQEACAAJ.

[Gen+07] Rosario Gennaro et al. «Secure Distributed Key Generation for Discrete-Log Based Cryptosystems». In: *Journal of Cryptology* 20 (mag. 2007), pp. 51–83. DOI: 10.1007/s00145-006-0347-3.

[Ger+] Ed Gerck et al. «The Business of Electronic Voting». In: *Proceedings Fifth International Conference on Financial Cryptography 2001*. A cura di Paul F. Syverson. Vol. 2339. Lecture Notes in Computer Science. Springer, pp. 243–268. DOI: 10.1007/3-540-46088-8_21.

[Gib+16] John Paul Gibson et al. «A review of E-voting: the past, present and future». In: *Annals of Telecommunications - annales des télécommunications* 71.7 (ago. 2016), pp. 279–286. DOI: 10.1007/s12243-016-0525-8. URL: https://hal.science/hal-01364859.

[GO+12] Ezequiel Gonzalez-Ocantos et al. «Vote Buying and Social Desirability Bias: Experimental Evidence from Nicaragua». In: *American Journal of Political Science* 56.1 (2012), pp. 202–217. ISSN: 00925853, 15405907.

[Gol01] Oded Goldreich. *Foundations of cryptography: Basic tools volume 1*. Cambridge, England: Cambridge University Press, ago. 2001.

[Goo+18] Nicole J. Goodman et al. «Another Digital Divide? Evidence That Elimination of Paper Voting Could Lead to Digital Disenfranchisement». In: *Policy & Internet* 10 (2018), pp. 164–184.

[Gre+13] Gurchetan S. Grewal et al. «Caveat Coercitor: Coercion-Evidence in Electronic Voting». In: *Proceedings of the 2013 IEEE Symposium on Security and Privacy*. SP '13. USA: IEEE Computer Society, 2013, 367–381. ISBN: 9780769549774. DOI: 10.1109/SP.2013.32. URL: https://doi.org/10.1109/SP.2013.32.

[Gri02] Dimitris Gritzalis. «Principles and requirements for a secure e-voting system». In: *Computers & Security* 21 (2002), pp. 539–556.

[Hal06] J.L. Hall. «Design and the support of transparency in vvpat systems in the US voting systems market». In: (2006).

[HC01] L. Hoffman e L. Cranor. «Internet voting for public officials». In: *Communications of the ACM* 44.1 (gen. 2001), pp. 69–71.

[Hei+11] Mario Heiderich et al. «The Bug That Made Me President a Browser- and Web-Security Case Study on Helios Voting». In: *International Conference on E-Voting and Identity*. 2011.

[Hen12] Christian Henrich. ≪Improving and Analysing Bingo Voting≫. Diss. Karlsruhe: Karlsruher Institut für Technologie (KIT), 2012.

[Hja+18] Friorik P. Hjalmarsson et al. ≪Blockchain-Based E-Voting System≫. In: *2018 IEEE 11th International Conference on Cloud Computing (CLOUD)* (2018), pp. 983–986.

[Hof13] Douglas R. Hofstadter. *Gödel, Escher, Bach: un'Eterna Ghirlanda Brillante*. Milano: Adelphi, 2013. ISBN: 9788845907555.

[HR17] Rifa Hanifatunnisa e Budi Rahardjo. ≪Blockchain based e-voting recording system design≫. In: *2017 11th International Conference on Telecommunication Systems Services and Applications (TSSA)* (2017), pp. 1–6.

[Iac] *Should the IACR use e-voting for its elections?* URL: https://www.iacr.org/elections/eVoting/.

[Ide] *Global in-country postal voting data*. URL: https://www.idea.int/data-tools/data/special-voting-arrangements/postal-in-country.

[JCJ05] Ari Juels, Dario Catalano e Markus Jakobsson. ≪Coercion-Resistant Electronic Elections≫. In: *Proceedings of the 2005 ACM Workshop on Privacy in the Electronic Society*. WPES '05. New York, NY, USA: Association for Computing Machinery, 2005, 61–70. ISBN: 1595932283. DOI: 10.1145/1102199.1102213. URL: https://doi.org/10.1145/1102199.1102213.

[JJ14] Peter Sandholt Jensen e Mogens K. Justesen. ≪Poverty and vote buying: Survey-based evidence from Africa≫. In: *Electoral Studies* 33 (2014), pp. 220–232. ISSN: 0261-3794. DOI: https://doi.org/10.1016/j.electstud.2013.07.020. URL: https://www.sciencedirect.com/science/article/pii/S0261379413001200.

[JJR02] Markus Jakobsson, Ari Juels e Ronald L. Rivest. ≪Making Mix Nets Robust for Electronic Voting by Randomized Partial Checking≫. In: *Proceedings of the 11th USENIX Security Symposium*. USA: USENIX Association, 2002, 339–353. ISBN: 1931971005.

[Jon10] Douglas W. Jones. ≪On Optical Mark-Sense Scanning≫. In: *Towards Trustworthy Elections: New Directions in Electronic Voting*. A cura di David Chaum et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 175–190. ISBN: 978-3-642-12980-3. DOI: 10.1007/978-3-642-12980-3_10. URL: https://doi.org/10.1007/978-3-642-12980-3_10.

[Kar+11] Fatih Karayumak et al. ≪Usability Analysis of Helios: An Open Source Verifiable Remote Electronic Voting System≫. In: *Proceedings of the 2011 Conference on Electronic Voting Technology/Workshop on Trustworthy Elections*. EVT/WOTE'11. USENIX Association, 2011, p. 5.

[KDS19] Ildar M. Khamitov, Victor Dostov e Pavel Shoust. ≪Secret Voting: Knowledge vs Trust≫. In: *Computational Science and Its Applications – ICCSA 2019*. Springer International Publishing, 2019, pp. 577–586. ISBN: 978-3-030-24296-1.

[KHF11] Reto Koenig, Rolf Haenni e Stephan Fischli. «Preventing Board Flooding Attacks in Coercion-Resistant Electronic Voting Schemes». In: *Future Challenges in Security and Privacy for Academia and Industry*. A cura di Jan Camenisch et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 116–127. ISBN: 978-3-642-21424-0.

[Koh+04] T. Kohno et al. «Analysis of an electronic voting system». In: *IEEE Symposium on Security and Privacy, 2004. Proceedings. 2004.* 2004, pp. 27–40. DOI: 10.1109/SECPRI.2004.1301313.

[Kri07] Taisya Krivoruchko. *Robust coercion-resistant registration for remote electronic voting*. 2007. DOI: 10.11575/PRISM/10182. URL: https://prism.ucalgary.ca/handle/1880/102243.

[KTV12] Ralf Küsters, Tomasz Truderung e Andreas Vogt. «Clash Attacks on the Verifiability of E-Voting Systems». In: *Proceedings of the 2012 IEEE Symposium on Security and Privacy*. SP '12. USA: IEEE Computer Society, 2012, 395–409. ISBN: 9780769546810. DOI: 10.1109/SP.2012.32. URL: https://doi.org/10.1109/SP.2012.32.

[KV18] Nir Kshetri e Jeffrey Voas. «Blockchain-Enabled E-Voting». In: *IEEE Software* 35 (lug. 2018), pp. 95–99. DOI: 10.1109/MS.2018.2801546.

[Liu+11] Yining Liu et al. «An Improved Electronic Voting Scheme without a Trusted Random Number Generator». In: nov. 2011, pp. 93–101. ISBN: 978-3-642-34703-0. DOI: 10.1007/978-3-642-34704-7_8.

[Lou+14] Panos Louridas et al. «Zeus: Bringing Internet Voting to Greece». In: *E-Democracy, Security, Privacy and Trust in a Digital World*. A cura di Alexander B. Sideridis et al. Springer International Publishing, 2014, pp. 213–223. ISBN: 978-3-319-11710-2.

[MCE15] Ester Moher, Jeremy Clark e Aleksander Essex. «Diffusion of Voter Responsibility: Potential Failings in E2E Voter Receipt Checking». In: *USENIX Journal of Election Technology and Systems (JETS)* (ago. 2015). URL: https://www.usenix.org/conference/jets15/workshop-program/presentation/moher.

[Mer00] Rebecca Mercuri. «Inside Risks: Voting Automation (Early and Often?)» In: *Commun. ACM* 43.11 (nov. 2000), p. 176. ISSN: 0001-0782. DOI: 10.1145/353360.353378. URL: https://doi.org/10.1145/353360.353378.

[MGK02] Lilian Mitrou, Dimitris Gritzalis e Sokratis Katsikas. «Revisiting Legal and Regulatory Requirements for Secure E-Voting». In: *Security in the Information Society: Visions and Perspectives*. A cura di M. Adeeb Ghonaimy, Mahmoud T. El-Hadidi e Heba K. Aslan. Boston, MA: Springer US, 2002, pp. 469–480. ISBN: 978-0-387-35586-3. DOI: 10.1007/978-0-387-35586-3_37. URL: https://doi.org/10.1007/978-0-387-35586-3_37.

[MN06] Tal Moran e Moni Naor. «Receipt-Free Universally-Verifiable Voting with Everlasting Privacy». In: *Annual International Cryptology Conference*. 2006.

[Mor16] Dan Morrell. *Secret ballots, verifiable votes*. Set. 2016. URL: https://www.harvardmagazine.com/2010/05/secret-ballots-verifiable-votes.

[Mor+20] Marco Morana et al. «Secure e-Voting in Smart Communities». In: *Proceedings of the Fourth Italian Conference on Cyber Security (ITASEC 2020)*. 2020.

[Moy04] Donald Moynihan. «Building Secure Elections: E-Voting, Security, and Systems Theory». In: *Public Administration Review* 64 (set. 2004), pp. 515–528. DOI: 10.1111/j.1540-6210.2004.00400.x.

[MR10] Niels Menke e Kai Reinhard. «Compliance of POLYAS with the common criteria protection profile - a 2010 Outlook on certified remote electronic voting». In: *4th International Conference on Electronic Voting 2010*. A cura di Robert Krimmer e Rüdiger Grimm. Bonn: Gesellschaft für Informatik e.V., 2010, pp. 109–118.

[MW20] Mohammed Khaled Mustafa e Sajjad Waheed. «An E-Voting Framework with Enterprise Blockchain». In: 2020.

[Neu+13] Stephan Neumann et al. *Towards A Practical JCJ / Civitas Implementation*. Cryptology ePrint Archive, Paper 2013/464. https://eprint.iacr.org/2013/464. 2013. URL: https://eprint.iacr.org/2013/464.

[Neu93] P.G. Neumann. «Security criteria for electronic voting». In: *16th National Computer Security Conference*. 1993.

[NR94] Valtteri Niemi e Ari Renvall. «How to Prevent Buying of Votes in Computer Elections». In: *Advances in Cryptology - ASIACRYPT '94, 4th International Conference on the Theory and Applications of Cryptology, Wollongong, Australia, November 28 - December 1, 1994, Proceedings*. Vol. 917. Lecture Notes in Computer Science. Springer, 1994, pp. 164–170. DOI: 10.1007/BFb0000432.

[NV12] Stephan Neumann e Melanie Volkamer. «Civitas and the Real World: Problems and Solutions from a Practical Point of View». In: *Proceedings of the 2012 Seventh International Conference on Availability, Reliability and Security*. ARES '12. USA: IEEE Computer Society, 2012, 180–185. ISBN: 9780769547756. DOI: 10.1109/ARES.2012.75. URL: https://doi.org/10.1109/ARES.2012.75.

[Oka97] Tatsuaki Okamoto. «Receipt-Free Electronic Voting Schemes for Large Scale Elections». In: *Proceedings of the 5th International Workshop on Security Protocols*. Berlin, Heidelberg: Springer-Verlag, 1997, 25–35. ISBN: 3540640401.

[OP17] Xavier Ochoa e C. Pelaez. «Affordable and secure electronic voting for university elections: The SAVE case study». In: apr. 2017, pp. 110–117. DOI: 10.1109/ICEDEG.2017.7962520.

[OSV11] Maina M. Olembo, Patrick Schmidt e Melanie Volkamer. «Introducing Verifiability in the POLYAS Remote Electronic Voting System». In: *2011 Sixth International Conference on Availability, Reliability and Security*. 2011, pp. 127–134. DOI: 10.1109/ARES.2011.26.

[Par+21] Sunoo Park et al. «Going from bad to worse: from Internet voting to blockchain voting». In: *Journal of Cybersecurity* 7.1 (feb. 2021). ISSN: 2057-2085. DOI: 10.1093/cybsec/tyaa025. eprint: https://academic.oup.com/cybersecurity/article-pdf/7/1/tyaa025/42533672/tyaa025.pdf. URL: https://doi.org/10.1093/cybsec/tyaa025.

[Ped91] Torben Pryds Pedersen. «A Threshold Cryptosystem without a Trusted Party». In: *Advances in Cryptology — EUROCRYPT '91*. A cura di Donald W. Davies. Berlin, Heidelberg: Springer Berlin Heidelberg, 1991, pp. 522–526. ISBN: 978-3-540-46416-7.

[Pom+14] Julia Pomares et al. «From piloting to roll-out: voting experience and trust in the first full e-election in Argentina». In: *2014 6th International Conference on Electronic Voting: Verifying the Vote (EVOTE)* (2014), pp. 1–10.

[Res12] Giuseppe Ugo Rescigno. *Corso di diritto pubblico*. 14ª. Bologna: Zanichelli, 2012. ISBN: 978-88-08-17614-1.

[RH10] Mohsen Rezvani e S. M. Hossein Hamidi. «MIZAN: A secure E-voting schema with vote changeability». In: *2010 International Conference on Information Society*. 2010, pp. 548–552. DOI: 10.1109/i-Society16502.2010.6018774.

[Riv06] Ronald Rivest. «The ThreeBallot Voting System». In: (nov. 2006). URL: mit.edu/rivest/Rivest-TheThreeBallotVotingSystem.pdf.

[Riv08] Ronald L. Rivest. «On the notion of 'software independence' in voting systems». In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 366.1881 (ago. 2008), pp. 3759–3767. DOI: 10.1098/rsta.2008.0149.

[RS06] Peter Y. A. Ryan e Steve A. Schneider. «Prêt à Voter with Re-encryption Mixes». In: *European Symposium on Research in Computer Security (ESORICS) 2006*. A cura di Dieter Gollmann, Jan Meier e Andrei Sabelfeld. Vol. 4189. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 313–326. ISBN: 978-3-540-44605-7.

[Rya05] Peter Y. A. Ryan. «A Variant of the Chaum Voter-Verifiable Scheme». In: WITS '05. New York, NY, USA: Association for Computing Machinery, 2005, 81–88. ISBN: 1581139802. DOI: 10.1145/1045405.1045414. URL: https://doi.org/10.1145/1045405.1045414.

[Rya+10] Peter Ryan et al. «PrÊt À Voter: a Voter-Verifiable Voting System». In: *Information Forensics and Security, IEEE Transactions on* 4 (gen. 2010), pp. 662 –673. DOI: 10.1109/TIFS.2009.2033233.

[Sch00] B. Schoenmakers. *Compensating for a lack of transparency*. 2000. URL: http://citeseer.nj.nec.com/schoenmakers00compensating.html.

[Sch23] Eva Johanna Schweitzer. *digital divide*. 2023. URL: https://www.britannica.com/topic/digital-divide.

[Sch99] B. Schoenmakers. ≪A simple publicly verifiable secret sharing scheme and its application to electronic voting≫. In: *Lecture Notes in Computer Science* 1666 (1999), pp. 148–164.

[SK95] Kazue Sako e Joe Kilian. ≪Receipt-Free Mix-Type Voting Scheme - A Practical Solution to the Implementation of a Voting Booth≫. In: *International Conference on the Theory and Application of Cryptographic Techniques*. 1995.

[SKW20] M.A. Specter, J. Koppel e D. Weitzner. ≪The ballot is busted before the blockchain: A security analysis of Voatz, the first internet voting application used in U.S. federal elections≫. In: *29th USENIX Security Symposium (USENIX Security 20)* (2020), 1535–1553. URL: https://www.usenix.org/conference/usenixsecurity20/presentation/specter.

[SP06] Krishna Sampigethaya e Radha Poovendran. ≪A framework and taxonomy for comparison of electronic voting schemes≫. In: *Computers & Security* 25 (mar. 2006), pp. 137–153. DOI: 10.1016/j.cose.2005.11.003.

[Sto+13] Susan Stokes et al. ≪Brokers, Voters, and Clientelism: What Killed Vote Buying in Britain and the United States?≫ In: (gen. 2013). DOI: 10.1017/CBO9781107324909.

[Tel01] International Working Group for Data Protection in Telecommunications. *Common Position on the Use of the Internet in the Conduct of Elections*. Berlin, set. 2001.

[TSP13] José Luis Tornos, José Luis Salazar e Joan Josep Piles. ≪An eVoting platform for QoE evaluation≫. In: *2013 IFIP/IEEE International Symposium on Integrated Network Management (IM 2013)*. 2013, pp. 1346–1351.

[Uni48] United Nations. *Universal Declaration of Human Rights*. Dic. 1948.

[VK06a] Melanie Volkamer e Robert Krimmer. ≪Die Online-Wahl auf dem Weg zum Durchbruch≫. In: *Informatik-Spektrum* 29.2 (2006), pp. 98–113. DOI: 10.1007/s00287-006-0064-1.

[VK06b] Melanie Volkamer e Robert Krimmer. ≪Overview on Online Voting≫. In: gen. 2006, pp. 339–350.

[VM07] Melanie Volkamer e Margaret McGaley. ≪Requirements and Evaluation Procedures for eVoting≫. In: *The Second International Conference on Availability, Reliability and Security (ARES'07)* (2007), pp. 895–902.

[Vol09] Melanie Volkamer. *Evaluation of electronic voting*. en. 2009[a] ed. Lecture notes in business information processing. Berlin, Germany: Springer, mag. 2009.

[Vol20] Dustin Volz. *Agencies warn states that internet voting poses widespread security risks*. Mag. 2020. URL: https://www.wsj.com/articles/agencies-warn-states-that-internet-voting-poses-widespread-security-risks-11588975848.

[VWT09] Adolfo Villafiorita, Komminist Weldemariam e Roberto Tiella. «Development, Formal Verification, and Evaluation of an E-Voting System With VVPAT». In: *IEEE Transactions on Information Forensics and Security* 4.4 (2009), pp. 651–661. DOI: 10.1109/TIFS.2009.2034903.

[Wan+17] King-Hang Wang et al. «A Review of Contemporary E-voting: Requirements, Technology, Systems and Usability». In: 2017.

[WH09] Janna-Lynn Weber e Urs Hengartner. «Usability Study of the Open Audit Voting System Helios». In: 2009.

[Wu+14] Zhen-Yu Wu et al. «An electronic voting mechanism for fighting bribery and coercion». In: *Journal of Network and Computer Applications* 40 (2014), pp. 139–150. ISSN: 1084-8045. DOI: https://doi.org/10.1016/j.jnca.2013.09.011. URL: https://www.sciencedirect.com/science/article/pii/S1084804513001999.

[XX13] Zhifeng Xiao e Yang Xiao. «Security and Privacy in Cloud Computing». In: *Communications Surveys & Tutorials, IEEE* 15 (gen. 2013), pp. 843–859. DOI: 10.1109/SURV.2012.060912.00182.

[Yak95] Alexander Yakobson. «Secret Ballot and Its Effects in the Late Roman Republic». In: *Hermes* 123.4 (1995), pp. 426–442. ISSN: 00180777. URL: http://www.jstor.org/stable/4477105.

[YM07] Yurong Yao e Lisa Murphy. «Remote electronic voting systems: an exploration of voters' perceptions and intention to use». In: *European Journal of Information Systems* 16.2 (2007), pp. 106–120. DOI: https://doi.org/10.1057/palgrave.ejis.3000672.

[YO13] Xun Yi e Eiji Okamoto. «Practical Internet voting system». In: *Journal of Network and Computer Applications* 36.1 (2013), pp. 378–387. ISSN: 1084-8045. DOI: https://doi.org/10.1016/j.jnca.2012.05.005. URL: https://www.sciencedirect.com/science/article/pii/S108480451200135X.

[ZL11] Dimitrios Zissis e Dimitrios Lekkas. «Securing e-Government and e-Voting with an open cloud computing architecture». In: *Government Information Quarterly* 28.2 (2011), pp. 239–251. ISSN: 0740-624X. DOI: https://doi.org/10.1016/j.giq.2010.05.010. URL: https://www.sciencedirect.com/science/article/pii/S0740624X10001383.