VOTARE SU INTERNET È POSSIBILE?

UNO STUDIO DEI PROTOCOLLI CLASSICI

Relazione per il corso di
Metodi Formali 2
Prof. Roberto Gorrieri

Scritta nel febbraio 2001
da
Fabrizio Bisi, bisi@cs.unibo.it

Introduzione

L'opinione pubblica mondiale ha parlato molto di voto elettronico nei mesi febbraio – marzo del 2000, in occasione di due eventi quasi concomitanti. Il 2 febbraio 2000 l'università tedesca di Osnabrück ha fatto uso di un sistema di voto via Internet per eleggere i rappresentanti degli studenti appoggiandosi all'azienda privata TrustCenter [1], mentre il 24 marzo 2000 il partito democratico dell'Arizona [2] ha indetto le primarie permettendo ai propri elettori, previa registrazione, di votare via Internet, anch'esso affidandosi a un'azienda privata, la Election.com [3].

Mentre qui da noi gli echi di queste vicende si sono smorzati nel giro di pochi mesi, cosicché il voto elettronico è tornato ad essere materia di pochi specialisti, i mass-media statunitensi hanno tenuto un riflettore puntato sulla discussione se sia possibile affiancare alle attuali metodologie di voto un sistema di voto elettronico remoto. Il fallimento del sistema di votazione per posta nelle presidenziali del dicembre 2000, fino a quel momento ritenuto dall'opinione pubblica americana un sistema affidabile e collaudato, ha riacceso in quel paese l'attenzione del grande pubblico sulla possibilità che il voto via Internet (ribattezzato i-vote) possa prima o poi sostituire l'attuale sistema di votazione a distanza, con pari o maggiore sicurezza ed eliminandone alcuni aspetti negativi (l'errore umano nei conteggi, le ambiguità nella convalidazione dei voti e l'incertezza sui tempi determinata dall'uso di un mezzo come la posta). La discussione se ciò potrà mai avvenire, tutt'ora in corso, coinvolge commissioni di tecnici, istituzioni politiche e aziende che già oggi forniscono sistemi di voto a società private e ad amministrazioni pubbliche americane.

"Voting Integrity Project" [4], un ente no-profit nato con l'intento di vigilare sull'integrità del voto, ritiene che gli attuali problemi dei sistemi di voto a distanza sarebbero amplificati dall'uso di un mezzo poco controllabile come Internet, che permetterebbe grazie all'automazione sia di effettuare brogli su larga scala sia di impedire agli elettori di esercitare il proprio diritto di voto tramite attacchi di "denial of service". Dello stesso avviso una commissione di esperti incaricata dal Segretario dello Stato della California Bill Jones, nel cui studio di fattibilità sul voto via Internet [5] si caldeggia piuttosto un approccio al voto elettronico per gradi, il cui primo passo consista nell'istituire seggi elettronici in cui l'identificazione degli elettori avvenga in maniera tradizionale.

Dall'altra parte le aziende private che forniscono oggi servizi di voto elettronico (la già citata Election.com, VoteHere.net [6] e molte altre) sostengono che i problemi di sicurezza dei sistemi di voto elettronico remoto non siano dissimili da quelli dei "tradizionali" sistemi di voto a distanza e che problemi come quello del "denial of service" potrebbero essere risolti moltiplicando server e linee telefoniche.

La mia intenzione originale era di presentare un quadro del dibattito in corso negli Stati Uniti sulla fattibilità del voto via Internet nelle elezioni pubbliche, descrivendo e analizzando le ragioni delle parti in causa e possibilmente traendo delle conclusioni personali. Comunque alcuni fattori mi hanno fatto cambiare idea. Innanzitutto una certa iniziale difficoltà a reperire documenti di livello tecnico (fino a pochi mesi fa nessuna delle aziende che forniscono servizi di voto elettronico negli USA aveva ancora divulgato nulla sui protocolli proprietari utilizzati); poi il fatto che il dibattito sia ancora aperto rende prematuro trarre una conclusione definitiva (per esempio una commissione incaricata dalla Casa Bianca [7] dovrebbe a mesi presentare un rapporto sull' Internet Vote). Soprattutto però la vastità della materia mi ha obbligato a ridimensionare i miei propositi originali e a operare una scelta se presentare materiale tecnico recente di elevata complessità senza un adeguato sostegno teorico o se piuttosto fornire una robusta base teorica grazie alla quale rendere agevole a chiunque l'esame del materiale più recente. La mia scelta è caduta su questa seconda opzione per cui in questa tesina, prendendo a pretesto l'idea di tracciare una storia dei protocolli di voto elettronico adatti a elezioni a distanza a larga scala, si analizzeranno in dettaglio alcuni protocolli classici della letteratura, evidenziandone pregi e difetti in vista di questa specifica applicazione.

Per chi volesse buttare uno sguardo sulla diatriba in corso negli Stati Uniti troverà qui di seguito i riferimenti alle fonti usate per scrivere questa introduzione (tutte reperibili su Internet).

Particolarmente utile [8] un articolo reso disponibile dal sito www.voter.com (testata giornalistica on-line che si occupa esclusivamente di elezioni) che permette di farsi un quadro generale del voto elettronico negli Stati Uniti. Per chi volesse esaminare materiale più tecnico sono attualmente due le aziende che hanno pubblicato i loro protocolli proprietari in rete: VoteHere.net [6] e SafeVote [9]. Può inoltre essere utile dare uno sguardo a [10] poiché raccoglie parecchio materiale anche recente.

SITI INTERNET CITATI

1. TrustCenter (www.trustcenter.de)
   azienda tedesca che ha fornito il servizio di voto elettronico all'università di Osnabrück nel febbraio 2000
2. Arizona Democratic Party (www.AzDem.org)
   sito Internet del partito democratico dell'Arizona
3. Election.com (http://Election.com)
   un'azienda statunitense che fornisce servizi di voto elettronico a pubblico e privato
4. Voting Integrity Project (www.VotingIntegrity.org)
   sito Internet di una organizzazione no-profit nata per vigilare la sicurezza e l'integrità del voto elettronico
5. California Internet Voting Task Force, A Report on the Feasibility of Internet Voting (gennaio 2000) (www.ss.ca.gov/executive/ivote)
   documento scritto da una commissione di esperti su richiesta del Segretario dello Stato della California sulla fattibilità del voto elettronico nelle elezioni pubbliche americane
6. VoteHere.net (http://voteHere.net)
   un'azienda statunitense che fornisce servizi di voto elettronico a pubblico e privato
7. National Workshop on Internet Voting (www.netvoting.org)
   pagina Web che annuncia l'incontro della commissione incaricata dalla Casa Bianca di redigere uno studio di fattibilità sull'introduzione del voto elettronico nelle elezioni pubbliche americane
8. "Voter.com In-Depth: e-Liminating Paper Ballots?" (1 dicembre 2000) (articolo scritto da Ryan Gillis di www.voter.com)
   articolo di taglio giornalistico che parla delle novità sul voto elettronico all'indomani delle presidenziali di novembre
9. SafeVote (www.safevote.com)
   un'altra azienda che fornisce servizi di voto elettronico
10. Pagina sul voto elettronico di Helger Lipmaa (ricercatore all'università di Helsinki) (www.tml.hut.fi/~helger/crypto/link/protocols/voting.html)

Requisiti di un buon sistema di voto elettronico

Diversi autori hanno indicato diversi insiemi di requisiti che un buon sistema di voto elettronico dovrebbe avere. Nell'appendice A di questa tesina riporto un confronto tra alcuni di questi.
La Cranor, dopo aver esaminato diversi articoli in letteratura (tra cui [Sal91], [FOO92] e [Sch94]), individua nel documento che descrive Sensus [Cra97] quattro requisiti fondamentali:

- **invulnerabilità** dagli attacchi esterni (da alcuni autori chiamata democracy)
- **accuracy**, ossia giustezza, accuratezza, mancanza di errori
- **privacy** (segretezza del voto)
- **verificabilità**, di cui la Cranor individua due definizioni, una debole in cui l'elettore può verificare il suo singolo voto e una più forte in cui tutti possono verificare che tutti i voti siano stati regolarmente conteggiati

Una proprietà non citata in [Cra97] ma che [FOO92] ritiene fondamentale è:

- **fairness**, l'incapacità per chiunque di conoscere i risultati parziali delle elezioni prima che abbiano termine

Oltre a questi requisiti teorici individua inoltre tre requisiti "pratici" che un sistema di voto deve avere per aumentare le proprie chance di essere utilizzato nel mondo reale:

- **facilità d'uso** (la Cranor la chiama convenienza)
- **flessibilità**, ossia capacità di adeguarsi a diversi tipi di elezioni
- **mobilità** (quanta libertà lascia all'utente sul luogo da cui votare)

Vediamo i requisiti citati sopra in dettaglio.

REQUISITI TEORICI

Invulnerability (democracy)

Un sistema è invulnerabile agli attacchi esterni (democratico) se:

(1) permette di votare solo agli aventi diritto e a nessun altro
(2) garantisce che ogni elettore legittimato non possa votare più di una volta

Questo requisito viene chiamato democracy da taluni autori, poiché in tal modo viene soddisfatto il principio base della democrazia "una testa, un voto"

Accuracy

Un sistema è accurato (senza errori) se:

(1) non è possibile che un voto venga alterato
(2) non è possibile che un voto legittimo non sia conteggiato nello spoglio finale
(3) non è possibile che un voto non valido sia conteggiato nello spoglio finale

Un sistema sarà completamente accurato se è impossibile ogni violazione o più realisticamente se ogni possibile violazione può essere trovata e corretta. Un sistema sarà parzialmente accurato se potrà garantire che ogni possibile violazione sia trovata, ma non corretta.

Privacy

Un sistema mantiene la privacy degli elettori se:

(1) né le autorità di voto né chiunque altro può associare il voto all'elettore che lo ha espresso
(2) nessun elettore può provare di aver votato in un certo modo

Il secondo punto, che può sorprendere a prima vista, è necessario per impedire la compravendita dei voti. In un sistema di voto elettronico che faccia uso di seggi elettorali e che si attenga strettamente al secondo requisito della privacy è assolutamente impossibile per chiunque forzare qualcun altro a votare in un certo modo, cosa questa quasi sempre garantita dai sistemi di voto tradizionali.

[FOO92] mette in evidenza come certi protocolli che sembrano rispettare il primo punto non lo facciano poi in caso di contestazione, costringendo l'elettore a mostrare il proprio voto per dimostrare di avere ragione. Analizzando i vari protocolli terremo conto di questo aspetto peculiare.

Verifiability

Secondo alcuni autori un sistema è verificabile se chiunque può verificare che tutti i voti siano stati correttamente conteggiati. Chiameremo questa proprietà verificabilità in senso forte.
Secondo altri [Sal91, FOO92] un sistema è verificabile se ogni elettore può verificare che il suo voto sia stato correttamente conteggiato senza mettere a repentaglio la propria privacy, ossia senza essere costretto a rivelare a nessuno il modo in cui ha votato. Questa definizione è più debole rispetto alla precedente poiché nella prima ogni elettore può verificare tutti i voti conteggiati, mentre nella seconda ciascuno è responsabile di controllare il suo proprio voto (e se non lo fa un possibile broglio può passare inosservato). Per questo motivo chiameremo questa proprietà verificabilità in senso debole.

Fairness

Un sistema è equo se impedisce a chiunque di venire a conoscenza del risultato parziale delle elezioni, in modo che nessuno possa influenzarne l'esito.

REQUISITI PRATICI

Convenience (User Friendliness)

Un sistema è conveniente se è facile da usare, ossia se permette agli elettori di votare velocemente e in una sola sessione. Questa proprietà, che gli autori di E-Vox chiamano più propriamente facilità d'uso (user-friendliness) viene definita nei documenti che descrivono E-Vox [Her97] e Sensus [Cra97] perché come vedremo il protocollo da cui entrambi traggono origine [FOO92] obbliga l'elettore a votare in due sessioni distinte.

Flexibility

Un sistema è flessibile se è possibile utilizzarlo per tipi diversi di elezioni (con e senza liste di candidati, referendum…).

Mobility

Un sistema è tanto più mobile quante meno restrizioni pone all'elettore sulla località da cui deve votare. Ad esempio, un sistema che permetta di votare da casa è massimamente mobile, mentre un sistema a seggi che permetta all'elettore di votare da qualsiasi seggio è più mobile di un sistema che obblighi l'elettore a recarsi a un seggio ben preciso.
La mobilità è uno dei fattori primari di interesse del voto elettronico. Il guaio è che più mobilità si concede, meno è possibile garantire i requisiti che un buon sistema di voto dovrebbe avere. Per esempio se si permette all'elettore di votare da dove vuole allora non importa quanto sia buono il protocollo di voto: chiunque potrà vendere il proprio voto votando in presenza del compratore . In tal caso il requisito 2 della privacy non sarà soddisfatto neppure se il sistema di voto non emette ricevute e non consente a posteriori di verificare che il proprio voto sia stato correttamente conteggiato. Ancora, un protocollo di voto (tradizionale o elettronico che sia) che costringa l'elettore a votare nel seggio vicino a casa soddisfa il requisito di democracy meglio di un protocollo che dia la massima libertà sulla scelta del seggio, poiché il sistema di identificazione ufficiale (carta d'identità in Italia, firma negli USA) verrà rafforzato dal riconoscimento o dal possibile riconoscimento fisico da parte delle persone che lavorano nel seggio.

Alla ricerca di un protocollo adatto a elezioni di larga scala

Tipi di protocolli di voto elettronico

Possiamo suddividere i protocolli di voto elettronico in 3 grandi famiglie:

- protocolli ad auto-arbitraggio (self-adjudicating protocols), in cui non esiste nessuna autorità centrale ma che prevedono interazioni tra gli elettori stessi
- protocolli con una singola autorità centrale che chiameremo Central Tabultating Facility (CTF), responsabile sia della raccolta dei voti che del conteggio finale
- protocolli con due entità istituzionali[1]:
    - una Validation Authority (VA) responsabile della raccolta dei voti, che conosce gli aventi diritto al voto e chi tra loro ha votato ma non può risalire al loro voto
    - una Tabulating Facility (TF), responsabile del conteggio finale dei voti, che conosce i voti ma non può sapere quali elettori li hanno espressi

Ai fini della nostra ricerca scartiamo a priori tutti i protocolli di auto-arbitraggio, utili magari per un consiglio d'amministrazione ma non certo per elezioni a larga scala.
Contrariamente a quanto si può ritenere esistono, come vedremo tra breve, protocolli con una singola autorità centrale che tutelano la privacy anche nei confronti della CTF. Comunque, secondo gli autori di E-Vox [Her97], i protocolli con una singola CTF non sarebbero buoni candidati per elezioni a larga scala per due ordini di motivi:

- tendono a fare uso di tecniche crittografiche computazionalmente più pesanti rispetto a quelli con due entità istituzionali
- la CTF costituisce un singolo punto di rottura: compromessa quella, tutto il sistema è compromesso.

Di seguito vedremo esempi di protocolli di entrambe le famiglie.

Perché scegliere un protocollo di voto crittografico

La storia del voto elettronico è la storia di protocolli di voto crittografici (si veda l'appendice C): senza crittografia non si va molto lontano.
Per esempio in [Cra96] si ipotizza l'uso di un protocollo di voto che non faccia uso di crittografia. Per mantenere l'anonimato dell'elettore si utilizzano due enti istituzionali, la Validation Autorithy e la Tabulating Facility. La VA si occupa di raccogliere i voti direttamente dagli elettori senza guardarli e nel far questo si preoccupa di verificare che votino solo elettori che ne hanno diritto e non più di una volta. Gli elettori potrebbero per esempio identificarsi con un numero identificativo precedentemente fornitogli dalla VA in fase di registrazione. Se il voto è regolare la VA elimina l'associazione tra voto e elettore e invia il voto anonimo alla TF. La TF provvede al conteggio e alla pubblicazione del risultato.
Ecco lo schema del protocollo riassunto brevemente (l'elettore generico lo designiamo con V_i, il suo numero identificativo con ID_i, il voto da lui espresso con m_i):

Protocollo ingenuo che non fa uso di crittografia
1. $V_i \rightarrow$ VA: ID_i, m_i
2. VA controlla se V_i ha diritto al voto e se non ha già votato. Se tutto va bene procede
3. VA $\rightarrow$ TF: m_i

Ovviamente questo protocollo ha una marea di problemi, di cui possiamo citarne alcuni:

[1] In questa tesina mi atterrò a una nomenclatura uniforme, sostituendola quando occorre a quella dei documenti originari. Nel caso specifico la VA e la TF vengono chiamate Validator e Tallier in [Cra97] o Administrator e Counter in [FOO92] e in [Her97].

- chiunque abbia accesso al meccanismo di voto potrebbe provare a votare usando codici identificativi a caso, avendo buone chance di indovinare codici regolarmente assegnati ad altri elettori
- anche se VA è tenuta a non controllare e registrare i voti che gli arrivano, senza protezione crittografica cosa gli impedisce di farlo? Occorre una fiducia cieca in questa istituzione
- se la VA è disonesta, oltre a controllare i voti può anche impunemente modificarli prima di inviarli a TF
- occorre fiducia cieca anche in TF, poiché potrebbe benissimo scartare i voti che gli arrivano e sostituirli nel conteggio finale con voti di proprio gusto senza che nessuno se ne accorga

Questo protocollo non soddisfa nessuno dei requisiti di un buon voto elettronico. Possiamo però migliorarlo significativamente facendo uso di alcune semplici tecniche crittografiche. Per esempio potremmo far sì che gli elettori utilizzino una firma digitale così da certificare la provenienza dei voti (usando per esempio la chiave privata di un sistema a chiavi asimmetriche, indicata con SK_i nello schemino sotto). Inoltre gli elettori potrebbero criptare i loro voti con la chiave pubblica di TF (PK_{TF}), in modo che VA non possa violare la privacy degli elettori, né modificarne il voto.

Protocollo ingenuo migliorato con tecniche crittografiche

1. $V_i \rightarrow$ VA: $SK_i(ID_i , PK_{TF}(m_i))$
2. VA controlla se V_i ha diritto al voto e se non ha già votato. Se tutto va bene procede
3. VA $\rightarrow$ TF: $PK_{TF}(m_i)$

Questo protocollo ha ancora moltissimi problemi che lo rendono inutilizzabile. Per esempio VA e TF hanno ancora entrambi ampi margini di manovra per poter commettere frodi senza essere scoperti. Il punto è che grazie a una banale applicazione di tecniche crittografiche siamo riusciti a trasformare un protocollo che fa acqua da tutte le parti in un protocollo che, a meno che VA e TF non colludano, riesce perlomeno a garantire la segretezza del voto (il requisito 1 della privacy, anche se nel senso debole specificato in [FOO92]) e l'invulnerabilità da attacchi esterni (ma non l'accuratezza). Il problema più grave di questo protocollo è che se TF e VA colludono allora di nuovo la segretezza del voto viene violata. Di seguito vedremo come alcuni protocolli proposti in letteratura cercano di risolvere questo problema.

Protocolli a uno e due enti di [NSS91] e [Sal91]

Nel 1991 Nurmi, Salomaa e Santean [NSS91] propongono un protocollo a due enti che può essere visto come un miglioramento del protocollo ingenuo descritto nella sezione precedente.

Durante la **prima fase di voto** VA distribuisce ad ogni elettore che ne faccia richiesta un tag di identificazione segreto (chiamiamolo ID_i). Terminata questa fase, VA invia a TF una lista di tutti i tag assegnati, naturalmente senza specificare a chi sono stati assegnati, dopodiché smette di prendere parte al protocollo.

Durante la **seconda fase di voto**, l'elettore generico V_i invia su un canale anonimo a TF il suo tag di identificazione e un file criptato contenente il voto e una copia del tag. In questo modo TF può verificare la correttezza del voto dalla validità del tag e controllare se quell'elettore ha già votato.Se è tutto a posto TF pubblica il file criptato, cosicché l'elettore può avere la prova che il suo voto è stato conteggiato, e solo allora l'elettore risponde inviando le chiavi per decriptare il voto, permettendo così a TF di venire a conoscenza del voto in chiaro.

Al termine delle elezioni TF pubblica una lista dei file criptati insieme ai voti associati, cosicché in fase di verifica (considerata dal documento una **terza fase di voto**) ogni elettore può controllare se il suo voto è stato regolarmente conteggiato. Se non è così può inoltrare un reclamo inviando nuovamente il tag, il file criptato contenente tag e voto e la chiave per decriptarlo. Poiché nella seconda fase di voto il file criptato era stato pubblicato (punto 5 della scaletta sotto), TF non può sostenere di non averlo ricevuto.

Esiste poi una quarta fase in cui è possibile cambiare voto, ma questa caratteristica non ci interessa per cui tralasceremo l'analisi di questa fase.

Protocollo a due enti di Nurmi, Salomaa e Santean [NSS91]
Prima fase di voto
1. $V_i \rightarrow$ VA: richiesta di tag
2. VA controlla se V_i ha diritto al voto e se non ha già votato
3. VA $\rightarrow V_i$: ID_i
Al termine della prima fase
4. VA $\rightarrow$ TF: $\forall i\ ID_i$
Seconda fase di voto
5. $(V_i) \rightarrow$ TF: $ID_i, E_{k_i}(ID_i, m_i)$
6. TF $\rightarrow$ *: $E_{k_i}(ID_i, m_i)$
7. $(V_i) \rightarrow$ TF: ID_i, k_i
Al termine della seconda fase
8. TF $\rightarrow$ *: $\forall i\ E_{k_i}(ID_i, m_i), m_i$
Fase di verifica (terza fase di voto)
9. V_i controlla se $E_{k_i}(ID_i, m_i)$ è stato pubblicato col voto corretto, altrimenti esegue 10
10. $(V_i) \rightarrow$ TF: $ID_i, E_{k_i}(ID_i, m_i), k_i$
Fase di modifica del voto (non ci interessa)

Oltre a garantire la democracy come già faceva il protocollo ingenuo visto sopra, [NSS91] rispetta anche la verificabilità debole, ossia ogni elettore può individualmente controllare che il suo voto sia stato correttamente conteggiato. Comunque ha ancora diversi problemi. Il più grave è che se VA e TF colludono, la privacy può ancora essere violata. Per risolvere questo problema Salomaa in [Sal91] propone una variante a questo protocollo. Se due enti lavorano insieme, afferma l'autore, tanto vale considerarli uno solo.
Così propone un protocollo con un ente solo in cui la prima fase di voto, quella dell'assegnamento dei tag, viene risolta usando un protocollo ANDOS (all-or-nothing disclosure of secrets). Questo risolve il problema della collusione. Comunque i protocolli ANDOS sono computazionalmente pesanti e poco adatti in circostanze dove gli utenti siano più di poche decine, onde ragion per cui il protocollo è inadatto a elezioni a larga scala.
Entrambe le varianti del protocollo falliscono a soddisfare il secondo requisito della privacy e parte dei requisiti di accuratezza.
Non soddisfano il secondo requisito della privacy perché il meccanismo che permette all'elettore di dimostrare di aver votato in un certo modo può essere usato per la compravendita dei voti. Come abbiamo già avuto modo di dire questo è un peccato veniale in quanto non esiste alcun modo per soddisfare il secondo requisito della privacy e insieme la verificabilità a meno che non si obblighi l'elettore a votare in un seggio. Anche volendo rinunciare alla verificabilità in favore della mobilità la compravendita dei voti non se sarebbe impedita: basterebbe che chi voglia vendere il proprio voto voti davanti a un testimone.
Un fatto più grave è che non soddisfano parte dell'accuracy visto che TF può votare per tutti gli elettori che si sono astenuti e anche se questi lo scoprono e reclamano non possono dimostrare di avere ragione. Comunque, come vedremo più avanti, questo non è un problema semplice da risolvere ed anche protocolli meglio progettati soffrono dello stesso problema.
Abbiamo visto che solo il secondo può garantire la segretezza del voto anche in caso di collusione di tutte le autorità di voto esistenti, pagando però un prezzo troppo elevato in termini di costo computazionale. Di seguito vedremo un protocollo che grazie all'uso delle firme cieche può soddisfare la privacy anche nel caso che tutti gli organismi istituzionali colludano senza degradare significativamente la performance del sistema.

Un pratico schema di voto segreto per elezioni a larga scala [FOO92]

Nel 1992, Fujioka Okamoto e Ohta pubblicano un articolo [FOO92] in cui illustrano un protocollo di voto per elezioni a larga scala. I protocolli di voto segreti, dicono gli autori, sono di due tipi: quelli che fanno uso di messaggi crittografati e quelli che fanno uso di un canale anonimo. I protocolli del primo tipo hanno due tipi di problemi:

- poiché fanno uso di server se tutti i centri cospirano la privacy viene violata
- sono poco adatti a elezioni di larga scala poiché in caso di un elevato numero di elettori c'è un grosso sovraccarico computazionale e nelle comunicazioni

I protocolli del secondo tipo sono migliori, ma quelli esaminati dagli autori presentano gli stessi punti deboli:

- rispettano la privacy solo finché non ci sono contestazioni, mentre in caso di contestazioni obbligano gli elettori a svelare il voto per dimostrare di avere ragione
- non rispettano il requisito della fariness (equità) in quanto consentono al centro di venire a conoscenza dei risultati parziali delle elezioni, fornendogli un vantaggio e permettendogli di tentare qualcosa per influenzare l'esito delle votazioni

L'articolo allora propone un protocollo che fa uso di un canale anonimo progettato per soddisfare i due requisiti visti sopra (primo requisito della privacy in senso forte e fairness), anche nel caso in cui tutti i centri colludano.

Le tecniche crittografiche utilizzate sono tre:

- un sistema di commitment, che consente di inoltrare un messaggio senza che il destinatario possa leggerlo prima di aver ricevuto l'autorizzazione del mittente ma anche senza che il mittente abbia la possibilità di modificarlo nel frattempo (nello schema sotto indicato con $C_k(\ldots)$ dove k è la chiave da spedire in un secondo momento e che permetterà al destinatario di leggere il contenuto)
- un sistema di firme digitali (digital signatures), che come abbiamo già visto certificano la provenienza di un messaggio (nello schema sotto indicato con SK(…))
- un sistema di firme cieche (blind signatures), che permettono di far firmare un documento da qualcuno senza però consentirgli di leggerne il contenuto. La tecnica consiste nell'usare un blind factor b con cui moltiplicare il messaggio m (nello schema sotto b(m)) e nel spedire il tutto per farselo firmare nel modo ordinario (SK(b(m))); b sarà costruito in modo che sia possibile rimuoverlo in modo da ottenere il messaggio originario firmato dal destinatario (SK(m))

Le parti in causa sono il VA (chiamato nell'articolo Administrator), il TF (chiamato Counter) e gli elettori.

Il protocollo ha due fasi di voto, più una terza fase opzionale di verifica.

Nella **prima fase di voto** l'elettore V_i prepara il voto applicando uno schema di commitment e un blind factor $b(C_{k_i}(m_i))$ e inviando il tutto a VA sia con che senza firma digitale. VA verifica che la firma sia corretta e che appartenga a un elettore che ha diritto al voto e che non ha già votato. Se è tutto a posto rispedisce il voto al mittente firmato con la propria firma digitale ($SK_{VA}(b(C_{k_i}(m_i))$. Si noti che grazie al blind factor VA non sarà in grado in futuro di riconoscere i $C_{k_i}(m_i)$ e di associarli agli elettori. L'elettore controlla l'autenticità della firma, dopodiché se è tutto in regola toglie il blind factor ottenendo quello che è il certificato elettorale vero e proprio $SK_{VA}(C_{k_i}(m_i))$. Dopodiché invia su un canale anonimo il certificato elettorale sia dotato della firma di VA sia senza la firma di VA (per mostrare che è l'autore del voto). Al termine della prima fase di voto VA pubblica una lista di tutti gli elettori che hanno richiesto la certificazione, mostrando per ciascuno di essi il nome e i voti protetti dal blind factor sia con che senza firma digitale dell'elettore; TF pubblica una lista di tutti i voti anonimi ricevuti, pubblicando sia la copia con che quella senza firma digitale della VA.

Nella **seconda fase di voto** l'elettore controlla che le liste abbiano lo stesso numero di elementi. Se non è questo il caso e si sospetta una irregolarità della VA basta che ogni elettore invii il blind

factor a chi è preposto a fare chiarezza. Se invece le liste hanno lo stesso numero di elementi ma il voto non compare nella lista pubblicata da TF allora l'elettore può inviare come reclamo $C_{k_i}(m_i)$ e $SK_{VA}(C_{k_i}(m_i))$, mostrando di essere in possesso di una scheda di voto regolarmente certificata dalla VA. Se tutto va bene, l'elettore invia a TF sul canale anonimo la chiave k_i per togliere il commitment, insieme alla posizione nella tabella. In tal modo TF può aggiungere per ogni riga della tabella il voto in chiaro e la chiave dello schema di commitment.
In un'eventuale **fase di verifica** l'elettore può controllare che alla riga assegnata a lui sia stato associato il voto corretto. Se non è così basta che faccia di nuovo uso del canale anonimo inviando nuovamente il certificato elettorale insieme stavolta alla chiave. In tal modo potrà mostrare di essere in possesso di una scheda elettorale regolarmente certificata e di aver votato in un certo modo.

[FOO92]

Prima fase

Preparazione del voto

1. $V_i \rightarrow$ VA: V_i, $SK_i(b(C_{k_i}(m_i)))$, $b(C_{k_i}(m_i))$

Amministrazione

2. VA controlla se Vi ha diritto al voto e se non ha già votato. Se va tutto bene procede
3. VA controlla la validità della firma eseguendo $PK_i(SK_i(b(C_{k_i}(mi)))) = b(C_{k_i}(mi))$. Se va tutto bene procede
4. VA $\rightarrow V_i$: $SK_{VA}(b(C_{k_i}(m_i)))$
5. VA $\rightarrow$ *: V_i, $SK_i(b(C_{k_i}(m_i)))$, $b(C_{k_i}(m_i))$

Votazione

6. V_i toglie il blind factor, ottenendo il certificato di voto vero e proprio: $SK_{VA}(C_{k_i}(m_i))$
7. V_i verifica che la chiave di VA sia corretta. Se va tutto bene procede
8. (Vi) $\rightarrow$ TF: $SK_{VA}(C_{k_i}(m_i))$, $C_{k_i}(m_i)$

Raccolta dei voti (al termine della prima sessione)

9. TF $\rightarrow$ *: $\forall i\ SK_{VA}(C_{k_i}(m_i))$, $C_{k_i}(m_i)$

Seconda fase

Apertura dei voti

10. V_i controlla se il numero di voti conteggiati da TF è identico al numero di voti certificati da VA. Se va tutto bene procede
11. V_i controlla se il suo voto compare nella lista preparata da TF e in che posizione (pos). Se va tutto bene procede
12. $(V_i) \rightarrow$ TF: pos, k_i

Conteggio

13. TF $\rightarrow$ *: $\forall i\ SK_{VA}(C_{k_i}(m_i))$, $C_{k_i}(m_i)$, k_i, m_i

Verifica (opzionale)

14. Se V_i riscontra un'irregolarità esegue 15
15. $(V_i) \rightarrow$ TF: $SK_{VA}(C_{k_i}(m_i))$, $C_{k_i}(m_i)$, pos, k_i

Questo protocollo soddisfa i due punti della democracy grazie al sistema delle firme digitali e a quello delle firme cieche. Soddisfa il primo punto della privacy anche se i due centri colludono e anche in caso di contestazioni. Soddisfa la verificabilità in senso debole e soddisfa la fairness.
Comunque, malgrado la protezione del commitment, ha nei confronti dell'accuracy gli stessi problemi del protocollo precedente: se alcuni elettori si astengono (cioè se non inviano neppure un voto di astensione), la VA può decidere di barare e di votare per loro.
Gli autori mettono inoltre in evidenza un altro pericolo per l'accuratezza: se le chiavi spedite dall'elettore per aprire il voto non funzionano, è impossibile capire se è l'elettore (legittimo) che sta facendo il furbo o se è la TF a essere disonesta. L'elettore avrebbe poco interesse a non mostrare il

proprio voto ma proprio per questo la TF sarebbe maggiormente sospettata di essere corrotta, per cui si può pensare a uno scenario in cui un gruppo di elettori si mettono d'accordo per gettare discredito sulla TF e sull'esito dell'elezione.

Sistemi di voto elettronico

Abbiamo trovato un protocollo [FOO92] che soddisfa quasi tutti i requisiti e che si dichiara adatto per elezioni a larga scala. Questo protocollo ha avuto grande successo e su di esso sono state tentate molte implementazioni. Nel seguito ne analizzeremo due: Sensus, sviluppato da Lorrie Cranor [Cra97] ed E-Vox, sviluppato come tesi di laurea al MIT da Mark A. Herschberg [Her97] e utilizzato al MIT stesso per eleggere i rappresentanti degli studenti nel 1998. Entrambi utilizzano tecnologie Internet, dimostrando la validità del protocollo teorico anche in applicazioni remote. Entrambi, come vedremo, falliscono a soddisfare tutti i requisiti del protocollo originale.

Sensus

Nel paragrafo precedente abbiamo visto quali requisiti teorici soddisfa il protocollo descritto in [FOO92], ma abbiamo tralasciato di parlare dei requisiti pratici, in parte perché tali requisiti (facilità d'uso, flessibilità, mobilità) sono propri di un sistema di voto elettronico completo, piuttosto che di un protocollo teorico lasciato non specificato negli aspetti implementativi.
Comunque se c'è un requisito pratico che [FOO92] non possiede indipendentemente dall'implementazione è la facilità d'uso. Tale protocollo costringe infatti l'elettore a due sessioni di voto obbligatorie distinte, più un'eventuale terza fase di verifica. Questa procedura di voto scoraggerebbe senza dubbio molti elettori, spingendoli a non votare o a non completare la procedura di voto. Così, sia gli autori di Sensus che quelli di E-Vox hanno modificato il protocollo originario per renderlo più conveniente riducendo le fasi obbligatorie di voto a una sola, come nel caso dei sistemi di voto tradizionali.
La principale differenza tra Sensus e [FOO92] è che mentre in quest'ultimo l'elettore deve aspettare che le elezioni siano finite prima di spedire la chiave di apertura del commitment, in Sensus il programma che fa le veci dell'elettore effettua in una sola sessione le seguenti operazioni:

- dialoga con la VA spedendole il voto protetto da blind factor e ricevendolo firmato (proprio come in [FOO92])
- manda il certificato elettorale alla TF (come n [FOO92])
- ottiene dalla TF una ricevuta di voto
- se tutto va bene spedisce immediatamente a TF la chiave per aprire il voto

Si noti che in questo modo Sensus rinuncia alla fairness, poiché la TF conosce i risultati parziali delle elezioni, in favore della convenienza.
Altri piccoli dettagli per cui si differenziano i due protocolli sono:

- lo schema di commitment viene rimpiazzato da un semplice sistema a chiavi asimmetriche. Nello schema che segue le chiavi pubbliche e private dell'elettore V_i usate per proteggere il voto sono chiamate rispettivamente PK_s e SK_s. Entrambe sono tenute segrete dall'elettore fino alle ultime fasi del protocollo. Ricordo che V_i ha anche un paio di chiavi PK_i e SK_i che gli servono per la firma digitale
- Per spedire messaggi si utilizza sempre la chiave pubblica del destinatario, dettaglio forse dato per scontato in [FOO92] ma comunque mai citato nell'articolo. Questa è una tecnica crittografica standard che impedisce a chiunque altro sul canale di leggere o modificare i dati trasmessi.
- I formati delle liste sono un po' diversi

Ecco lo schema del protocollo

Sensus
Fase di voto

Preparazione del voto (praticamente uguale a prima)

1. V_i prepara $c_i = b(PK_s(m_i))$
2. $V_i \rightarrow$ VA: $PK_{VA}(V_i, SK_i(c_i), c_i)$

Amministrazione (praticamente uguale a prima)

3. VA controlla se Vi ha diritto al voto e se non ha già votato. Se va tutto bene procede
4. VA controlla la validità della firma eseguendo $PKi(SKi(c_i)) = c_i$. Se va tutto bene procede
5. VA $\rightarrow$ V_i: $PK_{V_i}(SK_{VA}(c_i))$
6. VA $\rightarrow$ *: V_i, PK_i, flag "has voted"

Votazione, raccolta e apertura voti

7. V_i toglie il blind factor, ottenendo il certificato di voto vero e proprio: $SK_{VA}(d_i)$ dove $d_i = PK_s(m_i)$
8. V_i verifica che la chiave di VA sia corretta. Se va tutto bene procede
9. (Vi) $\rightarrow$ TF: $PK_{TF}(SK_{VA}(d_i), d_i)$
10 TF verifica che la chiave di VA sia corretta. Se va tutto bene procede
11. TF $\rightarrow$ *: $SK_{TF}(d_i)$, d_i (in posizione pos)
12. TF $\rightarrow$ (V_i): $SK_{TF}(d_i)$, pos
13 V_i verifica che la chiave di TF sia corretta. Se va tutto bene procede
14. (V_i) $\rightarrow$ TF: pos, SK_s
15. TF $\rightarrow$ *: $SK_{TF}(d_i)$, d_i, SK_{TF}, m_i (in posizione pos)

Verifica (opzionale)

16. Se V_i riscontra un'irregolarità invia in maniera anonima: $SK_{VA}(d_i)$, d_i, pos, SK_s

L'autrice dichiara che Sensus soddisfa tutti i requisiti teorici tranne il secondo requisito della privacy, per il quale afferma (giustamente) che non è possibile farci nulla senza sacrificare mobilità o convenienza. Comunque, per eventuali problemi di intercettazione o di ritardo dei messaggi in transito aggira l'ostacolo assumendo che l'utilizzatore del sistema fornisca dei canali sicuri e con buone garanzie di consegna in tempo. Altri assunti di Sensus sono:

- si assume che venga utilizzato un canale anonimo nelle comunicazioni tra elettore e TF, canale la cui implementazione è lasciata all'utente del sistema Sensus. La Cranor suggerisce l'uso di una catena di server WWW anonimizzatori
- si assume che il computer utilizzato dall'elettore sia sicuro e che non sia possibile per nessuno vedere o loggare quello che sta facendo l'utente
- si assume che i messaggi degli elettori a TF e VA non arrivino nello stesso ordine, assunzione valida in caso di un gran numero di elettori. Un elettore ha la facoltà di attendere un lasso di tempo prima di inviare il proprio certificato elettorale alla TF
- si assume che tutti gli algoritmi crittografici reggano

Solo con queste premesse il protocollo soddisfa i requisiti teorici coi limiti visti sopra.
Due osservazioni:

1) la scelta di sostituire un sistema di chiavi pubbliche a un vero e proprio sistema di commitment, probabilmente dovuta a motivi implementativi, introduce un potenziale problema di accuracy: l'elettore potrebbe sostenere di aver votato diversamente esibendo una seconda chiave pubblica che riesce ad aprire il voto mostrando un voto differente. Comunque, se questo trucco può riuscire agevolmente quando si inoltra un bit, dovrebbe essere più complicato se a essere "committed" è una lunga stringa di bit che codifica il voto (la Cranor comunque non giustifica questa scelta).
2) Si noti anche che la forma della lista pubblicata da VA è cambiata, poiché viene a mancare la necessità di una verifica pubblica dell'operato di VA: se l'elettore non riesce ad ottenere un regolare certificato elettorale può immediatamente inoltrare un reclamo, mentre in [FOO92] proprio perché la validazione del voto e l'apertura del voto erano in due sessioni distinte c'era bisogno durante la seconda sessione di un meccanismo che mantenesse traccia

pubblicamente delle operazioni effettuate da VA (in maniera che l'elettore potesse protestare a giochi fatti)

Per quanto riguarda i requisiti pratici Sensus è facile da usarsi, poiché ha una gradevole interfaccia e completa la sessione di voto in pochi minuti, è flessibile, permettendo anche la programmazione della scheda di voto con un linguaggio apposito (BLT = ballot description language) ed è mobile, in quanto è adatto a essere utilizzato su Internet.

E-Vox

Come anticipato anche E-Vox è un'implementazione del protocollo descritto in [FOO92] e anche E-Vox rimette mano al protocollo originario per aumentarne la convenienza. Contrariamente a Sensus, che lascia all'utilizzatore l'onere di decidere diverse strategie implementative, E-Vox si sforza di specificare ogni aspetto implementativo. Vediamo tali aspetti nel dettaglio

PECULIARITÀ DI E-VOX

Autenticazione

Mentre [FOO92] basa l'autenticazione degli elettori su un sistema a chiavi pubbliche, E-Vox fa uso di un sistema a password. I motivi sono due:

- per facilità d'uso, in quanto le password sono conosciute e ben accette anche da utenti inesperti
- per aggirare il problema della distribuzione delle chiavi. Il momento più sicuro in cui distribuire le chiavi sarebbe la registrazione, ma questo obbligherebbe gli utenti a ricordarsi chiavi lunghe almeno un centinaio di bit (mentre le password sono di pochi caratteri e soprattutto sono scelte dagli utenti stessi)

Distribuzione delle chiavi

Prima di partire con le elezioni tutti i server devono aver generato le proprie chiavi e devono essere venuti in possesso delle chiavi degli altri server. Il modo in cui questo avviene è volutamente non specificato e potrebbe essere fatto facendo uso di corrieri fidati.

Canali sicuri

Come abbiamo visto [FOO92] prevede due tipi di comunicazioni, quelle normali e quelle su canale anonimo. Mentre per le prime non specifica nessun tipo di protezione, per le seconde non suggerisce quale tecnologia adottare. Tutte le comunicazioni in E-Vox avvengono su canali sicuri, ossia canali protetti da un protocollo crittografico.
Se A vuole comunicare con B, dopo aver aperto una connessione fisica (per esempio un socket TCP), genera una chiave di sessione S e la usa per criptare il messaggio m e un MAC (Message Authentication Code). Dopodiché invia il messaggio ottenuto insieme alla chiave stessa protetta dalla chiave pubblica di B

$A \rightarrow B$: $S(m, MAC), PK_B(S)$

Il MAC è un hash del messaggio e serve per assicurare l'integrità del messaggio. Per assicurarsi che il messaggio sia integro il destinatario dovrà semplicemente riapplicare la stessa funzione di hashing su m e verificare che il risultato sia identico al MAC spedito.
Se il protocollo richiede che B risponda ad A potrà farlo usando la stessa chiave di sessione

$B \rightarrow A$: $S(m'), PK_B(S)$

In questo modo non c'è nessun bisogno che A abbia una coppia di chiavi asimmetriche. Questa caratteristica viene comoda perché in E-Vox è sempre l'elettore, che non ha chiavi asimmetriche, ad aprire connessioni sicure con la VA e la TF (che invece le hanno).

Canali anonimi

La soluzione adottata è quella di un singolo server anonimizzatore (anonymizer) che lavora utilizzando le connessioni sicure viste sopra.
L'anonymizer (che chiamerò AN) si pone tra l'elettore V e la TF. L'elettore non invia il certificato elettorale direttamente a TF perché altrimenti quest'ultima potrebbe risalire alla sua identità dall'indirizzo IP. Prepara comunque il certificato elettorale nel modo descritto sopra per le connessioni sicure in modo che solo TF possa leggerlo.

$$S(\text{certificato elettorale,MAC}),\ PK_{TF}(S)$$

A questo punto V prende questi due oggetti criptati e li invia all'anonymizer premurandosi di usare una connessione sicura.

$$V \rightarrow AN: \quad S'(\ S(\text{certificato elettorale,}\ldots),\ PK_{TF}(S),\ldots\ ),\ PK_{AN}(S')$$

L'anonymizer raccoglie per tutto il tempo delle elezioni i voti criptati in questa maniera, salvandoli in un database dove non registra informazioni su chi li ha mandati. Dopo la fine delle elezioni li manda in blocco a TF in ordine sparso.
Si noti che l'anonymizer o chiunque ascolti sul canale tra V e AN può risalire all'IP del votante ma non può conoscere il suo voto, mentre TF o chiunque ascolti le comunicazioni tra AN e TF ha in mano i certificati elettorali ma non può associarli al votante. Solo se anonymizer e TF colludono la privacy dell'elettore può essere violata.

Gestione degli errori

Rispetto al protocollo FOO viene aggiunto un server Commissioner (supervisore) a cui l'applet con cui gli elettori votano e gli altri server mandano i messaggi di errore. Il Commissioner provvede a memorizzarli in un file di log, mentre anche gli altri server mantengono un file di log nel caso ci siano problemi di comunicazione o comunque per verifica.
Il Commissioner sarà a sua volta supervisionato da una commissione umana, che sia durante che dopo le elezioni prenderà decisioni su quali azioni intraprendere in risposta ai reclami. Se per esempio una percentuale elevata di elettori dovesse inoltrare reclami su firme non valide ricevute dall'amministratore, la commissione potrà sospendere immediatamente la votazione per verificare che il server di amministrazione non sia compromesso.

PROTOCOLLO

L'implementazione del canale anonimo vista sopra ci consente di far votare l'elettore in una sola sessione preservando la fairness. Infatti l'elettore può inviare un messaggio all'anonymizer che contiene sia il voto sia le chiavi per aprirlo.
Vediamo come funzionano le cose nel dettaglio.
Prima di tutto l'elettore V_i prepara il voto applicando uno schema di commitment e un blind factor $b(C_{k_i}(m_i))$ e invia il tutto a VA assieme al proprio identificativo e alla propria password. Poiché si utilizza una connessione sicura non si deve temere che la password possa essere letta da qualcuno in ascolto. VA verifica che la password sia corretta e che appartenga a un elettore che ha diritto al voto e che non ha già votato. Se è tutto a posto rispedisce il voto al mittente firmato con la propria firma digitale ($SK_{VA}(b(C_{k_i}(m_i))$. Si noti che grazie al blind factor VA non sarà in grado in futuro di riconoscere i $C_{k_i}(m_i)$ e di associarli agli elettori. L'elettore controlla l'autenticità della firma, dopodiché se è tutto in regola toglie il blind factor ottenendo quello che è il certificato elettorale vero e proprio $SK_{VA}(C_{k_i}(m_i))$. Fin qui il protocollo è stato esattamente identico a quello originario, col solo uso della password al posto della firma digitale per l'autenticazione dell'elettore. Ora vengono le novità.
A questo punto V invia all'anonymizer il messaggio che l'anonymizer dovrà spedire in un secondo momento a TF. Questo messaggio contiene il certificato elettorale sia dotato della firma di VA sia senza la firma di VA e in più le chiavi per aprire il voto e il voto in chiaro.

L'anonymizer (non TF) raccoglie i voti che gli arrivano via via e li salva ciascuno in un file separato, senza alcuna informazione sulla loro origine. Allo scadere del tempo utile alla votazione invia tutti insieme i voti a TF in ordine sparso e pubblica una lista dei messaggi mandati.
Nella fase finale di conteggio TF rimuove prima i messaggi duplicati (esattamente identici, chiavi di sessione incluse), quindi controlla tutte le firme di VA, infine pubblica una lista di tutti i voti conteggiati, conta i voti e annuncia il risultato.
Per un'eventuale verifica l'elettore può controllare che alla riga assegnata a lui sia stato associato il corretto voto in chiaro. Se non è così basta che inoltri un reclamo inviando nuovamente il certificato elettorale insieme alla chiave. In tal modo potrà mostrare di essere in possesso di una scheda elettorale regolarmente certificata e di aver votato in un certo modo.

E-Vox

Prima fase: voto

Preparazione del voto

1. $V_i \rightarrow$ VA: userID, password, $b(C_{k_i}(m_i))$

Amministrazione

2. VA controlla se Vi ha diritto al voto e se non ha già votato. Se va tutto bene procede
3. VA controlla la validità della password. Se va tutto bene procede
4. VA $\rightarrow V_i$: $SK_{VA}(b(C_{k_i}(m_i)))$
5. VA $\rightarrow$ *: userID, $SK_{VA}(b(C_{k_i}(m_i)))$, $b(C_{k_i}(m_i))$

Votazione

6. V_i toglie il blind factor, ottenendo il certificato di voto vero e proprio: $SK_{VA}(C_{k_i}(m_i))$
7. V_i verifica che la chiave di VA sia corretta. Se va tutto bene procede
8. Vi $\rightarrow$ AN: [$V_i \rightarrow$ TF: $SK_{VA}(C_{k_i}(m_i))$, $C_{k_i}(m_i)$, m_i, k_i]

Seconda fase: raccolta e conteggio voti

9. AN $\rightarrow$ TF: invio di tutti i messaggi in ordine sparso

Conteggio

10. TF $\rightarrow$ *: $\forall i$ $SK_{VA}(C_{k_i}(m_i))$, $C_{k_i}(m_i)$, m_i, k_i

Verifica (opzionale)

11. Se V_i riscontra un errore inoltra un reclamo segnalando: $SK_{VA}(C_{k_i}(m_i))$, $C_{k_i}(m_i)$, pos, k_i

Per l'accuracy, gli autori oltre a rifarsi al protocollo originale prendono in considerazione un eventuale attacco di Denial Of Service: ogni server di E-Vox ha una lista nera di indirizzi IP da cui ha ricevuto troppe richieste. Per la democracy precisano che se un elettore invia più di un voto, TF è progettato per scartare i voti identici. La privacy può essere violata solo se anonymizer e TF colludono (mentre [FOO92] presupponeva un sistema sicuro per implementare il canale anonimo). Comunque l'autore cita a propria difesa uno studio di Rivest secondo cui le tecniche note per implementare un canale anonimo sono tutte passibili dello stesso tipo di timing attack (l'hacker legge la temporizzazione e la dimensione dei pacchetti in entrambe le estremità del canale e riesce a trarne delle conclusioni).
Per quanto riguarda i requisiti pratici: E-Vox è conveniente, in quanto si presenta all'elettore con una gradevole interfaccia grafica scritta in Java e permette di votare rapidamente e in una sola sessione; è sufficientemente flessibile, in quanto permette di specificare una lista di opzioni con anche eventuali candidati; rispetta il requisito della mobilità, poiché è progettato per funzionare da Internet e tutto quello che gli serve è un browser Web che supporti JDK 1.1

Migliorie a E-Vox

AMMINISTRATORI RIDONDANTI

L'amministratore è l'entità singola più potente. Solo la sua firma può convalidare un voto. Questo lascia a un amministratore disonesto un certo margine di manovra.
Per esempio, pochi minuti prima dello scadere del tempo utile per l'elezione, può prendere una percentuale tra gli elettori che non hanno votato e votare per loro. Chi non vota quasi certamente non controllerà se ci sono dei brogli fatti a suo nome, mentre per quanto riguarda eventuali ritardatari può sempre far credere che siano elettori disonesti che stanno cercando di votare due volte. In casi come questi, in cui si consente agli elettori di astenersi dal voto (senza mandare un messaggio di astensione), il protocollo non consente di distinguere tra elettore disonesto e amministratore disonesto.
Un possibile approccio a questo problema è creare tante unità amministrative che non hanno interesse a colludere (per esempio una per ogni candidato), ciascuna delle quali dovrà firmare il voto. Perché TF accetti un voto dovrà avere un numero minimo t di firme valide su n totali. Naturalmente occorre che sia $t > n/2$, altrimenti lo stesso elettore potrebbe preparare due voti validi facendoli firmare da due gruppi diversi di amministratori.
Questa modifica è stata effettivamente operata in [DuR99] ed ora fa parte del protocollo. L'autore fa notare che in questo modo, se gli amministratori non colludono tra loro, non è più necessario per l'accuracy che nessun elettore si astenga. Inoltre il protocollo è più scalabile, poiché il carico di lavoro può essere suddiviso tra i vari server amministratori. Questo rende E-Vox esteso ad Amministratori Multipli un possibile candidato per elezioni reali a larga scala.

ANONYMIZER RIDONDANTI

In questa sezione si propongono due migliorie al protocollo che sia [Her97] che [Dur99] propongono come possibili future estensioni.
La prima serve per ridurre il rischio che l'anonymizer scarti dei voti validi (anche in buona fede, magari per un crash temporaneo). La cosa salterà certamente fuori ma solo dopo che le elezioni si sono concluse, creando notevoli problemi. Per proteggersi da questo problema basta far sì che l'elettore mandi il proprio voto a diversi anonymizer in parallelo: se il voto arriva anche da uno solo di essi verrà correttamente conteggiato.
La seconda serve per minimizzare il rischio che l'anonymizer colluda con TF. La soluzione proposta è ortogonale alla precedente in quanto consiste nell'usare una catena di server anonymizer in modo che sia sufficiente un solo server affidabile nella catena perché sia garantito l'anonimato. Questa soluzione comunque non è fault-tolerance, poiché se solo un server ha dei problemi tutta la catena sarà impossibilitata a funzionare [Sah96]

Conclusioni

Siamo partiti col chiederci se sarà mai possibile avere elezioni pubbliche su Internet. Come abbiamo visto un simile grado di mobilità renderebbe impossibile prevenire la compravendita dei voti e diversi tipi di coercizione, fenomeni quasi sempre impediti dai sistemi di voto tradizionali. Tuttavia, lo stesso tipo di problema affligge tutti i sistemi di voto a distanza, compreso il voto via posta usato dagli Americani già da diverso tempo e considerato relativamente affidabile. I fautori del voto via Internet fanno notare che, malgrado l'uso massiccio che certi Stati fanno del voto via posta (l'Oregon ha addirittura abolito i seggi elettorali), non si sono mai verificati problemi di questo genere. Sembra allora probabile che prima o poi gli Stati Uniti arriveranno a sperimentare e ad adottare sistemi di voto su Internet a larga scala.
A questo punto ci siamo chiesti se esistono protocolli di voto che, fallendo nel soddisfare il secondo requisito della privacy, possano perlomeno soddisfare tutti gli altri requisiti di un buon sistema di voto elettronico e insieme siano adatti a elezioni in remoto a larga scala e qui abbiamo subito

limitato il nostro campo d'indagine ai protocolli classici, poiché il tempo a nostra disposizione per presentare questa tesina è insufficiente per fornire un'adeguata base teorica e contemporaneamente a descrivere nel dettaglio i sofisticati protocolli di voto elaborati negli ultimi 2 – 3 anni. Così i sistemi di voto che abbiamo visto per ultimi, Sensus e E-Vox, risalgono entrambi al '97. Malgrado qualche serio difetto, questi due protocolli, nati per finalità accademiche, hanno mostrato di aderire in maniera decorosa a un buon numero di requisiti e ci hanno lasciato una buona impressione su quanto le tecniche crittografiche possano fare per assicurare l'affidabilità di una votazione elettronica.
Sebbene [FOO92] abbia avuto un largo successo e moltissime implementazioni lungo tutto l'arco degli anni '90 (per esempio anche il sistema di voto sviluppato dal Cineca si basa su questo), l'approccio dei più recenti protocolli di voto è mutato. Invece di "nascondere" il voto l'idea è quella di dividerlo in pezzi (share), dando ciascun pezzo a un'autorità di raccolta dei voti differente. Ogni autorità di raccolta (TF) conteggia i voti indipendentemente e invia i risultati parziali a un'autorità centrale (CA), che li mette insieme per ottenere il risultato finale. Mentre le TF non conoscono il voto perché ne hanno solo un pezzo, anche la CA non conosce nulla dei voti, avendo ricevuto dalle TF solo risultati aggregati. In più si fa in modo che per ricomporre il voto originario servano t di n share (in modo da supportare la verificabilità senza grossi rischi per la privacy).
Questa idea è apparsa per la prima volta in [CFSY96] e viene utilizzata in [ADGN00] (il protocollo proprietario della VoteHere.net) e in [HS00] (il protocollo utilizzato dalla SafeVote) e in altri protocolli recenti da rapidamente visionati, tutti datati 1999 o 2000.

Bibliografia

Di seguito elenco i documenti personalmente visionati per scrivere la parte centrale della tesina. Ricordo che altri riferimenti presenti in questo documento si possono trovare nell'introduzione, e nell'appendice C (Storia dei protocolli di voto crittografici)

- [ADGN00] Adler, Dai, Green, Neff "Computational Details of VoteHere Homomorphic Election System", copyright 2000 (reperibile su http://voteHere.net)
  Documento che descrive il protocollo adottato da VoteHere (contiene materiale brevettato)
- [Cra96] Lorrie Faith Cranor "Electronic Voting - Computerized polls may save money, protect privacy" (www.acm.org/crossroads/xrds2-4/voting.html)
  Un articolo sulle caratteristiche che dovrebbe avere un buon sistema di voto elettronico, con alcuni cenni su come costruirlo
- [Cra97] Lorrie Faith Cranor and Ron K. Cytron, "Sensus: A Security-Conscious Electronic Polling System for the Internet." (http://www.research.att.com/~lorrie/pubs/hicss/)
  Proceedings of the Hawai`i International Conference on System Sciences , January 7-10, 1997, Wailea, Hawaii, USA
  Presenta il sistema di voto Sensus, implementazione dello schema di Fujioka, Okamoto, e Ohta
- [DuR99] Brandon William DuRette. "Multiple Administrators for Electronic Voting" (http://theory.lcs.mit.edu/~cis/theses/DuRette-bachelors.pdf), Bachelor's Thesis, Massachusetts Institute of Technology, May 1999.
  Espande il programma EVOX per amministratori multipli
- [FOO92] Atsushi Fujioka, Tatsuaki Okamoto, and Kazuo Ohta. "A practical secret voting scheme for large scale elections". In Jennifer Seberry and Yuliang Zheng, editors, *Advances in Cryptography*--AUSCRYPT '92, volume 718 of *Lecture Notes in Computer Science*, pages 244-251, Gold Coast, Queensland, Austrailia, 13-16 December 1992, Springer-Verlag.
  Presenta uno schema di votazione adatto per elezioni a larga scala che preserva privacy e fairness

- [Her97] Mark A. Herschberg. “Secure Electronic Voting Using the World Wide Web” (http://theory.lcs.mit.edu/~cis/theses/herschberg-masters.pdf) , Master's Thesis, Massachusetts Institute of Technology, June 1997.
  *E-Vox: p*resenta *un’implementazione del protocollo descritto nell’articolo di Fujioka, Okamoto e Ohta.*
- [HS00] Hirt M., Sako K. “Efficient Receipt-Free Voting Based on Homomorphic Encryption”, Eurocrypt 2000 (reperibile anche su www.safevote.com)
  Protocollo utilizzato da Safevote
- [NSS91] Nurmi, H., Salomaa, A., and Santean, L. “Secret ballot elections in computer networks”. Computers & Security 36, 10 (1991), 553--560.
  Primo articolo in cui gli autori parlano del two agencies protocol
- [Sah96] A. Sahuguet, “Electronic voting”, 1996 (http://www.cis.upenn.edu/~sahuguet/Voting)
- [Sal91] Salomaa, A. “Verifying and recasting secret ballots in computer networks”. In New Results and New Trends in Computer Science (Berlin, 1991), H. Maurer, Ed., vol. 555 of Lecture Notes in Computer Science, Springer-Verlag, pp. 283 --289.
  Presentati due protocolli, uno con due organismi istituzionali e uno con un solo organismo
- [Sha93] Michael Ian Shamos, “CFP'93 - Electronic Voting - Evaluating the Threat ” (http://www.cpsr.org/conferences/cfp93/shamos.html)
- [Sch94] Bruce Schneier, *Applied Cryptography*, John Wiley & Sons, New York, 1994, pp. 105-114

Appendice A

CONFRONTO DEI REQUISITI DI UN BUON SISTEMA DI VOTO IN ALCUNI AUTORI

Siccome in questa tesina sono presi come punto di riferimento i requisiti specificati dagli autori di Sensus [Cra97], tali requisiti verranno riassunti nella prima tabella, mentre le tabelle che seguiranno avranno come ultima colonna il corrispettivo in [Cra97]

REQUISITI SPECIFICATI IN [CRA97]

Proprietà	**Richiesta**	**Descrizione**
Accuracy	Si	(1) non è possibile che un voto venga alterato (2) non è possibile che un voto legittimo non sia conteggiato nello spoglio finale (3) non è possibile che un voto non valido sia conteggiato nello spoglio finale
Invulnerabilità (democracy)	Si	(1) permette di votare solo agli aventi diritto (2) garantisce che ogni elettore legittimato non possa votare più di una volta
Privacy	Si	(1) né le autorità di voto né chiunque altro può associare il voto all’elettore che lo ha espresso (2) nessun elettore può provare di aver votato in un certo modo
Verifiability	Si	Forte: chiunque può verificare che tutti i voti siano stati correttamente conteggiati Debole: chiunque può verificare che il proprio voto sia stato correttamente conteggiato
Convenience	No	Votazione rapida per l’utente e in una sola sessione
Flexibility	No	Sistema di voto adattabile a votazioni diverse
Mobility	No	Grado di libertà del luogo da cui votare

REQUISITI SPECIFICATI SULLO SCHNEIER [SCH94]

#	Richiesta	Descrizione proprietà	[Cra97]
1	Si	permette di votare solo agli aventi diritto	Democracy, 1
2	Si	garantisce che ogni elettore legittimato non possa votare più di una volta	Democracy, 2
3	Si	né le autorità di voto né chiunque altro può associare il voto all'elettore che lo ha espresso	Privacy, 1
4	Si	non è possibile che un voto venga alterato senza che la cosa venga scoperta	Accuracy, 1
5	Si	chiunque può verificare che il proprio voto sia stato correttamente conteggiato	Verificabilità debole
6	No	Chiunque sa chi ha votato e chi no	NON ESISTE

REQUISITI INDICATI IN FOO / E-VOX [FOO92,HER97]

Proprietà	Descrizione	[Cra97]
Completness	Tutti i voti validi sono conteggiati correttamente	Accuracy, 2
Soundness	L'elettore disonesto non può invalidare l'elezione	Accuracy, 1, 3
Privacy	Tutti i voti devono restare segreti	Privacy, 1
Unreusability	Nessun elettore può votare 2 volte	Democracy, 2
Eligibility	Possono votare solo gli aventi diritto al voto	Democracy, 1
Fairness / Countability	Impossibilità di conoscere il risultato dell'elezione prima del termine	NON ESISTE
Verifiability / Recoverability	Nessuno può falsificare il risultato della votazione se nessuno si astiene e tutti controllano il proprio voto (e la crittografia tiene)	Verificabilità debole

REQUISITI INDICATI IN [SAL91]

#	Descrizione proprietà	[Cra97]
1	Solo agli aventi diritto possono votare e solo una volta	Democracy, 1,2
2	Solo l'elettore sa cosa ha votato	Privacy, 1
3	Quando viene annunciato l'esito di una votazione, l'elettore può controllare che il suo voto sia stato correttamente conteggiato. Se questo non è il caso può aprire una contestazione senza rivelare il proprio voto	Verificabilità debole
4	In un certo arco di tempo l'elettore potrà cambiare voto, sempre senza rivelare il proprio voto	NON ESISTE

I COMANDAMENTI DEL VOTO ELETTRONICO IN [SHA93]

#	Import.	Descrizione proprietà	[Cra97]
1	Assoluta	Il voto di ciascuno è un inviolabile segreto	Privacy, 1
2	Assoluta	Ogni elettore voterà una volta e una volta sola e solo dove è autorizzato a farlo	Democracy, 1,2
3	Assoluta	Non sarà permesso la falsificazione, la corruzione e la compravendita dei voti	Privacy, 2
4	Media	Tutti i voti saranno riportati correttamente, con al più un piccolo scarto d'errore	Accuracy parziale
5	Minima	Il sistema di voto potrà servire per elezioni diverse	Flexibility
6	Minima	Saranno permesse verifiche campione senza violare la privacy	NON ESISTE

Appendice B

TABELLA DI RIEPILOGO DEI REQUISITI SODDISFATTI DAI PROTOCOLLI VISTI

	Democracy	**Privacy, 1**	**Verificabilità**	**Accuracy**	**Fairness**	**Note**
1	-	-	-	-	-	
2	Ok	Solo se TF, VA non colludono	-	-	-	
3	Ok	Solo se TF, VA non colludono	Debole	Problema astensione	-	
4	Ok	Ok	Debole	Problema astensione	-	Troppo pesante
5	Ok	Ok	Debole	1 problema astensione 2 incapacità di distinguere tra V e TF disonesto	Ok	
6	Ok	Ok	Debole	"	-	
7	Ok	Ok	Debole	"	Ok	
8	Ok	Ok	Debole	2 incapacità di distinguere tra V e TF disonesto	Ok	

1. protocollo ingenuo senza crittografia
2. protocollo ingenuo con crittografia
3. protocollo a due enti di [NSS91]
4. protocollo a un ente di [Sal91]
5. [FOO92]
6. Sensus
7. E-Vox
8. Estensione di E-Vox ad amministratori multipli

Appendice C

STORIA DEI PROTOCOLLI DI VOTO CRITTOGRAFICI

Chaum, in un articolo del 1981 [Cha81] sull'invio di posta elettronica anonima basata sull'utilizzo di pseudonimi digitali, propone il primo protocollo di voto con tecniche crittografiche. Questo protocollo fa uso di un sistema a chiave asimmetrica e di elenchi di pseudonimi digitali per rendere anonimo l'uso della posta elettronica; purtroppo non riesce a garantire che si possa risalire all'identità dei votanti. In seguito lo stesso autore proporrà una miglioria al protocollo grazie alla quale diventa impossibile risalire all'identità dei votanti [Cha88], ma in cui è anche impossibile correggere il voto di un elettore disonesto senza far ripartire l'elezione da zero [Ive92].
Nel 1985 Cohen e Fischer pubblicano una descrizione di uno schema di elezione sicura in cui è molto difficile per elettori disonesti invalidare un'elezione [CF85]. Purtroppo questo schema non protegge l'identità degli elettori dall'autorità dell'elezione. In seguito Cohen presenterà un miglioramento a questo schema distribuendo il potere dell'autorità centrale e offrendo più protezione per la privacy [Coh86]. Benaloh [Ben87] sostiene che questo schema è "ragionevolmente pratico" e che i problemi per implementarlo sono più di ordine politico che tecnico. Comunque riconosce anche che l'elettore deve possedere una conoscenza matematica di livello universitario per poter verificare i risultati dell'elezione. In più, a causa della grande complessità nelle comunicazioni previste dallo schema, votare può richiedere troppo tempo [SK94].

Nel 1994 Benaloh e Tuinstra [BT94] propongono un insieme di protocolli di voto a scrutinio segreto con possibilità di verifica che non consentono agli elettori di provare di aver votato in un certo modo (rispettando il requisito 2 della privacy). Diversamente da tutti gli altri protocolli discussi qui, questi protocolli richiedono di votare all'interno di una cabina elettorale. Gli autori sostengono che il più semplice dei loro protocolli non richiede agli elettori di eseguire calcoli complessi. Comunque, i protocolli più complessi che richiedono un minor grado di fiducia nelle autorità dell'elezione richiedono addirittura di portare un sistema elettronico portatile nella cabina elettorale. Inoltre anche il protocollo più sicuro non garantisce che l'elettore non possa essere forzato a votare in un certo modo.

RIFERIMENTI BIBLIOGRAFICI RELATIVI ALLA STORIA DEI PROTOCOLLI DI VOTO CRITTOGRAFICI

- [BT94] Benaloh, J., and Tuinstra, D. Receipt-free secret-ballot elections. In Proccedings of the Twenty-sixth Annual ACM Symposium on the Theory of Computing (May 23 --25, 1994), pp. 544--553.
- [Ben87] Benaloh, J. D. C. Verifiable Secret-Ballot Elections. PhD thesis, Yale University, December 1987. [Cha81] Chaum, D. Untraceable electronic mail, return addresses, and digital pseudonyms. Communications of the ACM 24, 2 (1981), 84--88.
- [Cha88] Chaum, D. Elections with unconditionally-secret ballots and disruption equivalent to breaking RSA. In Advances in Cryp-tology -EUROCRYPT '88 (Berlin, 1988), C. G. Gunther, Ed., vol. 330 of Lecture Notes in Computer Science, Springer-Verlag, pp. 177 -- 182.
- [Coh86] Cohen, J. D. Improving privacy in cryptographic elections. Tech. Rep. YALEU/DCS/TR-454, Yale University, February 1986.
- [CF85] Cohen, J. D., and Fischer, M. J. A robust and verifiable cryptographically secure election scheme (extended abstract). Tech. Rep. YALEU/DCS/TR-454, Yale University, July 1985. Also appeared in 1985 Foundations of Computer Science conference proceedings.
- [Ive92] Iversen, K. R. A cryptographic scheme for computerized general elections. In Advances in Cyptology -CRYPTO '91 (Berlin, 19 92), vol. 576 of Lecture Notes in Computer Science, Springer-Verlag, pp. 405 --419.
- [SK94] Sako, K., and Kilian, J. Secure voting using partially compatible homomorphisms. In Advances in Cryptology, Crypto'94 (1994), Lecture Notes in Computer Science, Springer-Verlag.