Version 1.0

Kodröstning
En genomgång av det norska systemet

Författad av

Alfred Krappman

krappman@kth.se

Fredrik Öman

frdo@kth.se

Kungliga Tekniska Högskolan

Stockholm

April 2013

Innehåll

1 Introduktion **1**

1.1 Syfte 1

2 Bakgrund **3**

2.1 Allmänna kodröstningsbegrepp 3

2.2 Hur garanterar systemet säkra valresultat? 5

2.3 Hur kan man förhindra att dekrypterade röster spåras? 5

2.4 Hur kan röster skickas säkert via internet? 6

2.4.1 Ett illustrerande exempel 7

3 Det norska systemet **10**

3.1 Hur kan kvittogeneratorn svara med korrekt kvittokod? 11

3.2 Enheter i det norska systemet 13

3.3 Flödesdiagram illustrerande röstregistrering 16

3.4 Flödesdiagram illustrerande ihopräkning 22

4 Diskussion **26**

5 Slutsats **29**

1 Introduktion

"Code voting", eller kodröstning, benämner system som möjliggör säker röstning via osäkra uppkopplingar och datorer. Hur dessa specifikationer uppnås varierar mellan olika implementationer men innefattar alltid någon form av kryptering. Vanligt förekommande är också utnyttjandet av en separat säker kanal för distribution av röstningsalternativ och eventuella kvittokoder.

Kodröstning kan användas i situationer där man vill underlätta för personen som ska rösta (röstningen kan exempelvis ske genom en PC) samtidigt som man fortfarande vill garantera sekretess och dataintegritet. Systemen kan motivera sin korrekthet genom transparens där systemarkitekturen ihop med matematiska egenskaper i krypteringen försäkrar att röstningen går rätt till. Möjliga användningsområden för kodröstning innefattar opinionsundersökningar där personen skulle kunna meddela sin röst via sin dator istället för att behöva avslöja sitt val för en person via telefon. Kodröstning kan och har även använts i riktiga politiska val. Elektronisk röstning används vid politiska val i USA [10]. Denna uppsats fokuserar dock på kodröstning som ska kunna utföras via en hemma-PC. Ett sådant system testades utav Norge i vissa distrikt vid kommunal- och landstingsvalet 2011.

I samband med det norska kodröstningssystemets testomgång 2011 publicerades dokument som redovisar hur systemet bör ha implementerats,med protokoll för de viktiga enheterna i systemet. Om inte annat anges baseras information om det norska systemet på källa [4]. Hur fungerar det norska systemet, vad baserar det sin säkerhet på och hur står sig de avvägningar de gjort gentemot alternativa implementationer?

1.1 Syfte

Att bena ut det norska systemet och genom det framställa en helhetsbild med fokus på säkerhetsaspekter är av intresse då användare av systemet bör ha tillgång till så mycket information som möjligt för att kunna lita på att systemet lämpar sig för att användas i verkliga politiska val. Samtidigt bör de ha tillgång till ett diskussionsunderlag då systemet eventuellt ska göras om inför en bredare lansering. Anknytningen

till alternativa system hjälper även till förståelse om kompromisser som systemarkitekterna tvingats göra och som också bör diskuteras.

2 Bakgrund

Detta avsnitt kommer ta upp centrala begrepp inom kodröstning och en genomgång av vad som gör kodröstning ett säkert alternativ till konventionella val. Detta följs sedan av en övergripande genomgång av det norska systemet.

2.1 Allmänna kodröstningsbegrepp

För att kunna genomföra elektroniska val som tillåter röstare att rösta från godtycklig (osäker) terminal måste man kunna garantera att röstaren kan erhålla en nivå av integritet och sekretess som går att jämföra med icke-elektroniska val. I detta avsnitt tas det upp några vanliga koncept samt strategier för att hantera de problem som kan uppstå kring sekretess, integritet och användarverifiabilitet vid implementering av elektroniska val.

Ett elektroniskt system skall kunna garantera att information om användares val på röstsedeln förblir hos användaren, det får inte gå att spåra en dekrypterad röst till röstaren under valprocessen. En förekommande strategi för att erhålla detta är att hålla systemet kvittensfritt (reciept-freeness). Denna strategi går ut på att en röstare inte i efterhand skall kunna bevisa för en tredje part vilka val hon har röstat på. Detta tar i sin tur bort möjligheten att kunna ägna sig åt röstförsäljning och skyddar röstaren från externa påtryckningar [7, s. 2]. Denna egenskap är alltså vital för att kunna bibehålla en röstares integritet.

För att öka trovärdigheten för ett röstningssystem kan man ge en röstare någon form av ”feedback” som ger röstaren möjlighet att säkerställa att systemet har mottagit rösten och att rösten har tolkats av systemet på ett korrekt sätt, detta bör förstås ske utan göra några stora kompromisser kring kvittensfrihet. Denna återkoppling kan uppnås genom att använda sig av s.k. anslagstavlor (bulletin boards) [12, s. 2][7, s. 4]. Det fungerar på så sätt att användaren krypterar och signerar sin röstsedel och skickar in röstsedeln till systemet som räknar ut en svarskod m.h.a. rösten och publicerar svarskoden på anslagstavlan. Denna svarskod har räknats ut tidigare av den auktoritet som har i uppdrag att skapa alla röstsedlar finns därför på användarens röstsedel.

Anslagstavlan är publik och är därför åtkomlig för alla godkända röstdeltagare och användaren kan därför gå in på anslagstavlan och verifiera att svarskoden finns publicerad [12, s. 2-3][7, s. 4]. Det viktiga här är att svarskoden är distinkt för varje röstkort och inte för varje kandidat eftersom röstarens sekretess i sådana fall skulle bero på hur väl röstaren har gömt sina kvittokoder från tredje part. Ett system som omöjliggör för en tredje part att kontrollera vad en röstare har röstat på kallas coercion resistant [8, s. 538][7, s. 1].

En annan lösning som inte tar lika stor hänsyn till kvittofrihet är den lösning som användes under de norska lokalvalen 2011. Varje röstkort innehöll unika kvittokoder för varje enskilt val på röstkortet och kvittokoden skickades till röstaren via en systemoberoende kanal, i det här specifika fallet valde man telekomnätet och skickade kvittokoden via ett SMS. Denna strategi ger röstaren, precis som strategin med anslagstavlor, möjlighet att verifiera att systemet har tolkat rösten rätt. Men, i och med att denna implementation tillåter en röstare att verifiera att systemet registrerat de specifika röstval som gjorts så uppnås en högre grad av användarverifiabilitet jämfört med användandet av publika anslagstavlor. Risken finns dock att en tredje part kan ta reda på röstarens val om det finns kännedom om röstarens röstkoder. Att denna risk är förhållandevis liten kan motiveras med att kvittokoden skickas till användaren via en systemoberoende kanal och att systemet försöker täcka upp detta säkerhetshål genom att göra det möjligt för användaren att rösta hur många gånger hon vill. Vilket gör att en tredje part aldrig kan vara säker på vad användaren lägger sin röst på [4, s. 4, 12-14, 20].

Om ett system klassificeras som robust (system med hög grad av 'robustness') kan systemet hantera ett scenario då delar av systemet slutat fungera som förväntat. Ett exempel på detta kan vara om man delar upp valet över flera auktoriteter (t.ex. den del som ansvarar för räknandet av röster) och att det endast behövs en delmängd av dessa auktoriteter för att systemet skall fortsätta vara säkert. Flera delar kan då sluta fungera eller korrumperas utan att valet blir osäkert. [3, s. 1-3].

2.2 Hur garanterar systemet säkra valresultat?

Röstningssystem som kan motivera att en röst som registrerats korrekt alltid kommer att närvara vid ihopräkningen kallas för end-to-end [12, s. 2]. Termen används ofta i samband med "cast as intended" (rösta som tänkt) vilket beskriver system som motiverar för den röstande individen att rösten som registrerades verkligen stämmer överens med vad röstaren verkligen ville rösta på [13, s. 22]. Det norska systemet hävdar att den uppfyller de villkor som båda dessa termer kräver av ett system [13, s. 22-23].

Då användare av ett röstsystem har möjlighet att kontrollera om deras röst har registrerats korrekt anses systemet inneha egenskapen användarverifieringsmöjlighet (voter verifiability). Denna egenskap syftar till att motivera systemets trovärdighet för enskilda användare. Men en användare kan självklart inte vara säker på att systemet räknat ut ett korrekt valresultat enbart baserat på vetskapen om att deras egen röst registrerats korrekt. Om systemet gör det möjligt för alla användare att också kontrollera om valresultatet är korrekt anses systemet erbjuda universalkontroll (universal verifiability). Denna egenskap förstärker systemets trovärdighet ytterligare. [8, s. 538, 541][9, s. 81-82].

2.3 Hur kan man förhindra att dekrypterade röster spåras?

Det norska systemet använder sig utav mix-nets för att kasta om meddelanden för att förhindra spårning av röster vid sammanräkningen. Krypterade röster som matas in i ett mix-net kommer ut i en annan ordning. Ett förenklat mixande system skulle kunna bestå av en enda mixande enhet. Denna enhet publicerar en publik nyckel som röstare kan kryptera sina röster med. Den mixande enheten tar sedan emot de krypterade rösterna, dekrypterar dem och matar ut dem i en annan ordning än de kom in i.

Implementationen med en enda enhet är dock problematisk i och med att en enda enhet ser samtliga röster och därmed vem som har röstat på vad. Om enheten är korrupt och exempelvis publicerar resultatet så är fiaskot ett faktum. För att förhindra detta kan multipla enhet införas. Dessa enheter publicerar tillsammans en

publik nyckel som avsändaren krypterar sina röster med. Avsändaren skickar sedan in sina röster till någon enhet i systemet som sätter igång en kedjereaktion. Den första enheten modifierar krypteringen på alla inskickade röster, skyfflar runt ordningen på rösterna och skickar vidare till nästa enhet som upprepar det den första enheten gjorde. Enheterna delar på en privat nyckel i syfte att de alla måste samarbeta för att dekryptera rösterna. Efter att rösterna passerat den sista enheten samlar enheterna ihop sina delar av den privata nyckeln och dekrypterar de mixade rösterna tillsammans [14, s. 5-7].

Alternativ eller supplement till mix-nets för att förhindra spårning av röster vid sammanräkning inkluderar användandet av homomorfiska krypteringsalgoritmer. En homomorfisk krypteringsalgoritm möjliggör att den underliggande meddelandestrukturen bevaras även vid operation på motsvarande krypterande meddelanden. Låt A och B vara två krypterade meddelanden. En homomorfisk krypteringsalgoritm ger att F(A * B), där F är avkrypteringsfunktionen och * någon operation ex. multiplikation, ska kunna utföras utan att först dekryptera A och B [11]. Genom att använda sig utav denna egenskap så behöver aldrig en individuell röstares egen röst avkrypteras och riskera uppspårning. Rösterna kan istället först läggas ihop i sitt krypterade stadium och först efter ihopräkningen dekrypteras, där dekrypteringen enbart leder till valresultatet.

2.4 Hur kan röster skickas säkert via internet?

Det norska systemet använder sig utav ElGamal-kryptering för att kunna använda sig av internet då en röstares PC vidarebefodrar röstvalen. ElGamal bygger på Diffie-Hellman-protokollet och är ett exempel på ett asymmetriskt krypteringssystem [5, s. 68].

Såsom namnet antyder använder asymmetriska krypteringssystem sig av två olika nycklar. En nyckel används för att kryptera och en annan nyckel för att dekryptera. Nyckeln som används för att kryptera brukar kallas för publik nyckel. Den publika nyckeln kan distribueras fritt utan säkerhetsrisk då den enbart kan användas för att kryptera meddelanden. Den andra nyckeln brukar kallas för privat nyckel och har

möjlighet att dekryptera meddelanden som den publika nyckeln krypterat. Den privata nyckeln är alltså privat och bör endast innehas utav mottagaren [1].

ElGamals krypteringsmetod grundar sin säkerhet på svårigheten att beräkna diskreta logaritmer.

$$14^x \equiv 7 \bmod 19 \tag{1}$$

Talet i ekvation 7 i ekvation (1) tillhör gruppen Z_{19}^*. Gruppen består utav de heltal som bildas då en gruppgenerator som exempelvis talet 14 exponentieras och sedan tas modulo 19. Gruppen består alltså av talen 1, 2 ...18. En diskret-logaritmfunktion skulle vara hjälpsam för att räkna ut talet x i (1). Det finns dock inget effektivt sätt att räkna ut x på eftersom att x skulle kunna, med lika stor sannolikhet för varje möjlighet, vara vilket tal som helst i Z_{19}^*. 14 är som sagt en gruppgenerator och kan därmed vid exponentiering bilda vilket gruppelement som helst. Ett naivt angreppssätt går ut på att testa samtliga medlemmar i gruppen Z_{19}^*, vilket skulle kräva 18 körningar och därmed vara linjär i gruppens storlek [5, s. 64-65]. Om 19 byts ut mot ett gigantiskt primtal så tar det orimligt lång tid att hitta exponenten, då exponentieringen och modulosoperationen kräver mycket datorkraft. ECRYPT II rekommenderar en storlek på 1248 bitar om man vill ha ett skydd som står emot något motsvarande en statlig enhet med en budget på cirka 300 miljoner dollar [2, s. 29-30].

2.4.1 Ett illustrerande exempel

Nedan följer ett enkelt exempel på ett röstningsscenario som använder sig utav ElGamal.

Olof vill skicka in sin röst på alternativ 17. Han tänker använda sig utav ElGamal. Sessionen kommer att köra på gruppen Z_{19}^* med generatorn 2. Först måste mottagaren, rösträkningsstationen i det här fallet, beräkna en publik nyckel A.

$$A \equiv g^a \bmod p \equiv 2^6 \bmod 19 = 7 \tag{2}$$

där lilla a väljs på måfå. Stationen har nu skapat båda sina nycklar, där A är den publika nyckeln och a den privata nyckeln. Olof fortsätter med att kryptera sitt val 17, vilket han gör genom att beräkna två stycken nya tal som tillsammans utgör krypteringen av hans val. Observera att han använder sig av rösträkningsstationens publika nyckel A. Talet k (som tas modulo p) väljs på måfå och kan kastas bort efter dessa beräkningar.

$$C \equiv g^k \bmod p \equiv 2^8 \bmod 19 = 9 \tag{3}$$

$$D \equiv mA^k \bmod p \equiv 17 * 7^8 \bmod 19 = 16 \tag{4}$$

Olof skickar in C och D till rösträkningsstationen, som med hjälp av dessa två tal enkelt kan dekryptera Olofs röst och få fram talet 17 i klartext. Detta görs med hjälp av den privata nyckel a på följande sätt:

$$X \equiv C^a \bmod p \equiv 9^6 \bmod 19 = 11 \tag{5}$$

$$X^{-1} \bmod p \equiv 7 \bmod 19 \tag{6}$$

eftersom att $11z \equiv 1 \bmod 19, z = 7$

$$X^{-1}D \equiv (C^a)^{-1} * (m(g^a)^k) \bmod p \equiv (g^{ak})^{-1} * (m(g^{ak})) \bmod p \equiv m \bmod p \tag{7}$$

Med tal insatta blir det alltså:

$$7 * 16 \bmod 19 = 17 \tag{8}$$

[5, s. 68-70].

Olof ska inte behöva göra alla dessa beräkningar på egen hand, det ska hans PC göra åt honom. För att säkerställa att PCn verkligen utför en ElGamal-kryptering kan man använda sig utav något som heter proof-of-knowledge". Det norska systemet använder sig utav ett så kallat Schnorr-proof som används för att bevisa kunskap om diskreta logaritmer. Ett Schnorr-proof möjliggör för datorn att bevisa att den känner till den hemliga nyckeln a. Beviset kan skickas via en osäker kanal då beviset inte behöver innehålla själva nyckeln.

För att utföra beviset väljer datorn ett hemligt tal t < p och räknar ut talet $T = g^t$.

Talet T kan skickas över internet till en verifierare som fortsätter med att välja ett tal c tillhörande Z_{19}^*. Talet c skickas i sin tur tillbaka till datorn via internet. Datorn räknar sedan ut talet $s = ac + t \bmod p$, som den sedan slutligen skickar till verifieraren. Verifieraren kan nu vara nästan säker på att datorn verkligen känner till sin hemliga nyckel a om g^s är detsamma som A^cT.

$$A^cT = ((g^a)^c)g^t = g^{ac+t} = g^s \tag{9}$$

Denna metod är inte fullständigt säker eftersom att datorn skulle kunna gissa på ett tal s, men sannolikheten att gissningen ska fungera är 1/p (då det finns p tal att välja på)[6] och eftersom p bör väljas till ett gigantiskt primtal är denna sannolikhet mycket liten. Beviset kan också upprepas tills önskvärd säkerhet har säkerställts.

3 Det norska systemet

Innan valet börjar distribueras valsedlar till potentiella röstare. Dessa valsedlar innehåller röstalterantiven med tillhörande kvittokoder som är unika för varje röstare. Kvittokoderna har genererats i förväg. När en röstare vill rösta på något alternativ använder hon sin PC där hon kan logga in på en speciell röstningstjänst som kräver en identitetskontroll. Hon kommunicerar sina röstval till datorn och väntar sedan på konfirmation på att hennes röst registrerats korrekt. Denna konfirmation sker via SMS, som snart efter att hon skickat in valen till systemet ska svara med motsvarande kvittokoder. Även datorn bekräftar att röstningen skett men är inte medveten om kvittokoderna.

Då datorn mottagit röstarens val kommunicerar datorn dessa till röstsamlaren via en kanal över internet som upprättats vid identitetskontrollinloggningen. Valen skickas alltså över internet, dock i krypterad form (mer om det nedan). Röstsamlaren sparar alla mottagna (krypterade) röster tillsammans med avsändaren. Avsändaren ska kunna rösta upprepade gånger och röstsamlaren måste vara medveten om vilken röst som är senast mottagen (som därmed är kandidat för att närvara vid ihopräkningen). Detta uppnås med hjälp av sekvensnummer, som numrerar varje mottagen röst.

Röstsamlaren har en ytterligare uppgift, nämligen vidarebefordran av mottagen röst till kvittogeneratorn. Den skickar dock inte samma krypterade röstval som den mottagit utav röstaren utan utför ytterligare beräkningar på rösten med hjälp av sin egna nyckel. Denna bearbetade röst skickas till kvittogeneratorn, som med hjälp av sin egna nyckel kan utvinna kvittokoden utav röstvalet. Kvittogeneratorn kommunicerar kvittokoderna via SMS.

När valtiden gått ut stänger röstsamlaren och skickar sina ihopsamlade röster till dekrypteraren som dekrypterar rösterna och tillkännager valresultatet. Röstare kan när som helst under valomgången pappersrösta på det vanliga sättet vid en vallokal. Denna pappersröst gäller då istället för alla tidigare och framtida elektroniska röster.

3.1 Hur kan kvittogeneratorn svara med korrekt kvittokod?

Systemet använder sig utav ElGamal. Det första som sker är skapandet av nödvändiga nycklar för att ElGamal ska kunna fungera. Man bestämmer sig för en ändlig cyklisk grupp G av ordning q som genereras utav g. q måste vara ett primtal och bör vara tillräckligt stort för att det diskreta-logaritmproblemet ska vara svårt. Detta fungerar precis som vid en vanlig ElGamal-session. Kvittokoderna beror på röstvalen genom att de beräknas på följande sätt:

$$r(v) = d((f(v))^s) \tag{1}$$

v - röstvalet (exempelvis socialdemokraterna som kanske är röstval 3)

r - funktionen som omvandlar röstvalet till kvittokoden

f - en injektiv funktion som avbildar röstvalet v på gruppen G

s - ett tal unikt för varje väljare

d - en pseudoslumpmässig funktion unik för varje röstare, beter sig som en hash-funktion

Efter att dessa funktioner och tal fastställts så är det dags för att ta fram tre hemliga nycklar. De måste uppfylla förhållandet $a_1 + a_2 \equiv a_3 \bmod q$.
De distribueras till följande enheter:

a1 - Dekrypteraren

a2 - Röstsamlaren

a3 - Kvittogeneratorn

Tre publika tal y_1, y_2 och y_3 offentliggörs och beräknas genom:

$$y_1 = g^{a_1} \tag{2}$$

$$y_2 = g^{a_2} \tag{3}$$

$$y_3 = g^{a_3} \tag{4}$$

Förberedelsestadiet är nu avklarat. Röstaren har gjort sina val. Hon skickar in valen via sin PC, som fortsätter med att utföra ungefär samma beräkningar som tidigare beskrivet under *"Hur kan röster skickas säkert via internet?"*. Se tabell nedan för en jämförelse mellan ElGamal såsom beskriven i bakgrundssektionen och ElGamal såsom använd i första steget i det norska systemet.

Enkel ElGamal	**PC i norska systemet**
Publik nyckel A	Publika talet y_1
Gruppgenerator g	Gruppgenerator g
Hemligt engångstal k	Hemligt engångstal t_i
Meddelande m	Krypterat röstval $f(v_i)$
Tupel $(C, D) \equiv (g^k \bmod p, mA^k \bmod p)$	Tupel $(x_i, w_i) \equiv (g^{t_i} \bmod q, f(v_i) * (y_1^{t_i}) \bmod q)$

Notera indexsymbolen i. Röstvalen krypteras alltså var för sig. Röstsamlaren dekrypterar dock inte mottaget meddelande utan fortsätter med ytterligare bearbetning av de krypterade rösterna. Precis som tidigare producerar röstsamlaren en tupel (en tupel per röstval).

$$(\check{x}_i, \check{w}_i) = (x_i^s, w_i^s * (\check{x}_i^{a_2})) \tag{5}$$

$(\check{x}_i, \check{w}_i)$ skickas till kvittogeneratorn som med hjälp av denna tupel kan beräkna kvittokoderna, då:

$$\check{w}_i * \check{x}_i{}^{-a3} = ((w_i^s) * \check{x}_i{}^{a_2}) * \check{x}_i{}^{-a3} = w_i^s * \check{x}_i{}^{-a_1} = w_i^s * (x_i^s)^{-a_1} = (w_i x_i^{-a_1})^s = (f(v_i))^s \tag{6}$$

Kvittokoden för ett specifikt val beräknas som tidigare nämnt genom $d(f(v_i)^s)$. Eftersom att kvittogeneratorn har $(\check{x}_i, \check{w}_i), d$ samt a_3 så kan enheten också beräkna kvittokoden genom $d(\check{w}_i * \check{x}_i{}^{-a3})$.

Inget av ovanstående leder dock till att verkligen dekryptera rösten inför ihopräkningen. Det ansvaret ligger på dekrypteraren. Röstsamlaren sparar alla kodade röster som skickats in av röstare via datorn och identitetskontrollen (det vill säga tuplarna (x_i, w_i)). det är också det som dekrypteraren tar emot, orört utav röstsamlaren. Dekrypteraren innehar talet a_1, som tidigare användes vid krypteringen av rösterna

(eftersom att y_1 användes). Med hjälp av a_1 och (x_i, w_i) så kan dekrypteraren ta fram $f(v_i)$ genom följande operation:

$$w_i(x_i)^{-a_1} = (f(v_i)y_1^{t_i})(g^{t_i})^{-a_1} = (f(v_i)(g^{a_1})^{t_i})(g^{-a_1 t_i}) = f(v_i) \tag{7}$$

Dekrypteraren kan ta fram v_i med hjälp utav en funktion som backar tillbakaävbildningen på gruppen G som funktionen f tidigare gjorde.

3.2 Enheter i det norska systemet

PC

En osäker plattform som är gränssnittet mellan användaren och systemet. Hanterar kryptering av röstkoder och ser till att data paketeras på ett korrekt sätt innan det skickas iväg till identitetskontrollen. Utför en extra kontroll av de val som användaren angav vid röstningstillfället matchar den bearbetade kvittensdatan som skickas från identitetskontrollen i den avslutande delen av hela röstprocessen.

Identitetskontroll

Fungerar som en ett gränssnitt mellan datorn och röstsamlaren, där all kommunikation mellan datorn och röstsamlaren måste gå via identitetskontrolleraren. Vid en röstningsession kommunicerar röstaren vilken dator hon tänker använde till identitetskontrollen. Denna enhet följer sedan upp denna information då datorn begär anslutning till röstsamlaren, genom att kontrollera om datorn rapporterar korrekt aktiv användare. Sessionen tilldelas en sessionsid utav identitetskontrollen som sedan används genom hela röstningsprocessen. Identitetskontrollen signerar också val som användaren gjort, vilket hela tiden följs upp utav upprepade verifikationsbegärningar från övriga enheter i systemet. Dessa nästan paranoida kontroller syftar till att garantera att dataintegritet uppehålls genom hela processen. Denna enhet antas fungera perfekt och inte gå sönder eller korrumperas under röstningen. Enheten antas vara utvecklad av en extern aktör.

Röstsamlaren

Enheten tar emot och samlar ihop röstsedlar från alla användare av systemet och ansvarar för att skicka in röstdatan till dekrypteraren. Bearbetar mottagna röster och

skickar vidare till kvittogeneratorn. Kontrollerar att endast en session per användare är aktiv samtidigt och hanterar ett sekvensnummer som den använder för att vara överens med kvittogeneratorn om hur många röstningsomgångar en specifik användare startat. När röstsamlaren stänger då valet avslutas skickar den iväg alla krypterade röstsedlar till de dekrypteraren som skall anonymisera, dekryptera och i slutändan räkna röstsedlarna.

Kvittogeneratorn

Tar emot krypterade röster bearbetade av röstsamlaren och beräknar kvittokoder som skickas ut till röstaren. Rösterna skickas via en extern kanal som är oberoende av systemet, exempelvis via SMS. Kontrollerar att antalet röster registrerade av röstsamlaren stämmer överens med antalet egna registrerade röster.

Beviskontroll

För att kunna säkerställa dataintegritet måste enheter i systemet kunna verfiera att krypteringen har genomförts på ett korrekt sätt samtidigt som användarens uppgifter skyddas. Beviskontrollen säkerställer att PCn har genomfört krypteringen av röstkoder korrekt och erbjuder andra enheter att fråga om datorn verkligen utfört beviset. Enheten antas bete sig korrekt.

Dekrypteraren

Tar emot ihopsamlade krypterade röster ifrån röstsamlaren. Utför säkerhetskontroller i samarbete med inspektören. Mixar rösterna för att anonymisera, dekrypterar och offentliggör rösterna.

Inspektören

Fungerar som en extra säkerhetsåtgärd för att se till att inga glapp finns då det är dags att räkna ihop rösterna. Ser till så att röstsamlaren och kvittogeneratorn är överens om mottaget innehåll. Verifierar signaturerna som datorerna utförde tillsammans med identitetskontrollen. Säkerställer att mixningen som dekrypteraren utför inte förstört röstkoderna.

Bevisade säkerhetsegenskaper

Följande säkerhetsegenskaper är bevisade i [4].

- Alla ärliga röstare som använder sig utav korrekt fungerande datorer kommer att få sin senast lagda röst räknad om inspektören godtar valomgången.
- Om en ärlig röstare använder en korrupt dator för att skicka in flera röstningsval kan systemet inte längre garantera att de senare rösterna räknats korrekt.
- Om röstsamlaren förblir korrekt fungerande genom valomgången så kommer inspektören kunna avgöra om valet har gått rätt till.
- Även om kvittogeneratorn är korrupt så kan den inte lista ut någonting utöver vad röstaren ska få för kvittokod.
- Röster mottagna av en korrupt dekrypterare kommer inte att kunna spåras. Även röstresultatet kommer med stor sannolikhet att vara korrekt.
- En korrupt inspektör kommer inte att kunna lista ut vem som har röstat på vad.

De ovanstående egenskaperna förutsätter att endast en intern (dvs. inte PCn eller röstaren) enhet är korrupt per valomgång. En enda korrupt enhet kan stoppa valomgången.

3.3 Flödesdiagram illustrerande röstregistrering

Följande diagram visar hur det går till då en röst korrekt registreras av systemet, samt hur kvittokoderna tas fram och skickas ut.

S - Auktoritet

Feid - Identitetskontroll

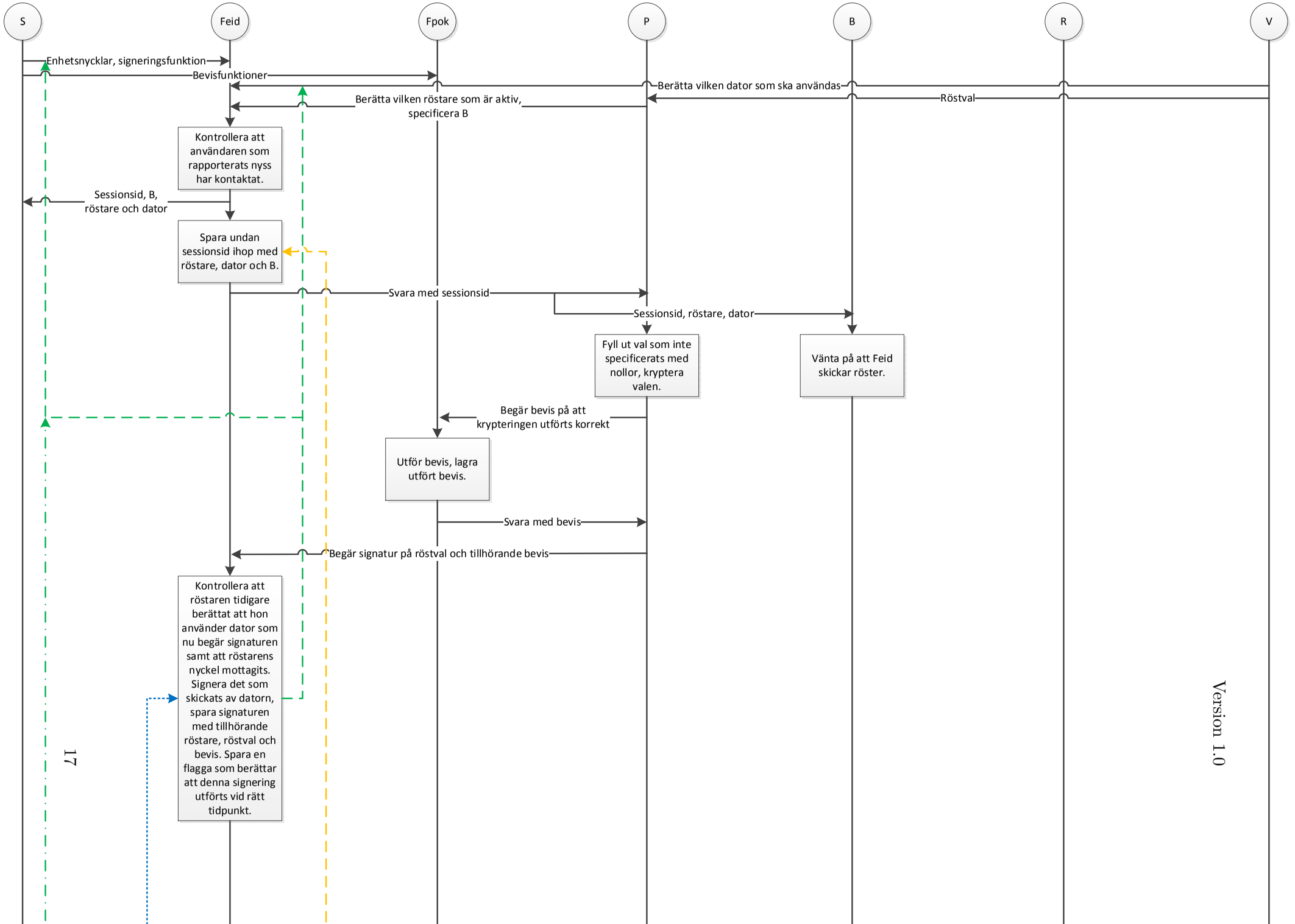
Fpok - Beviskontroll

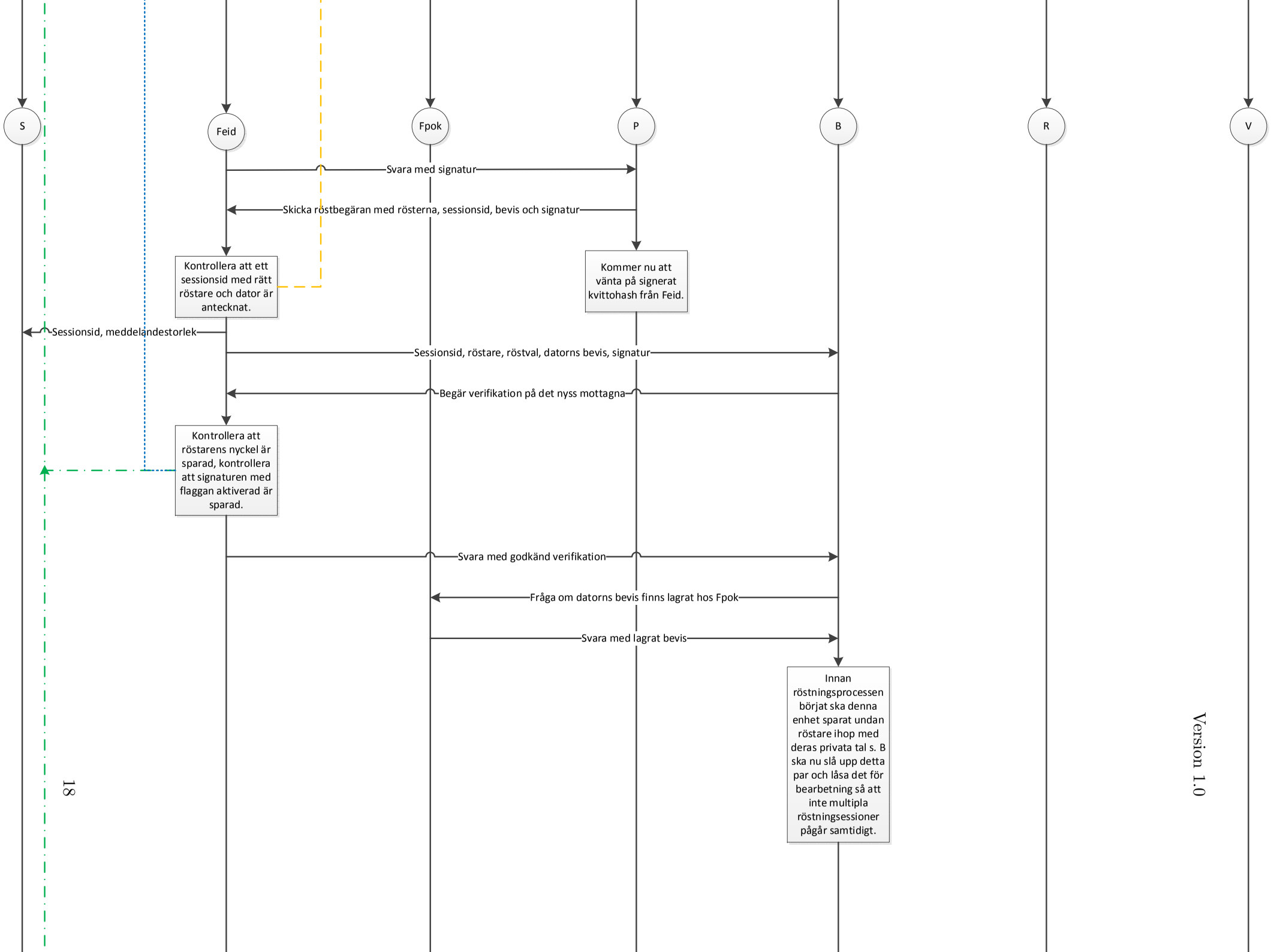
P - PC

B - Röstsamlare

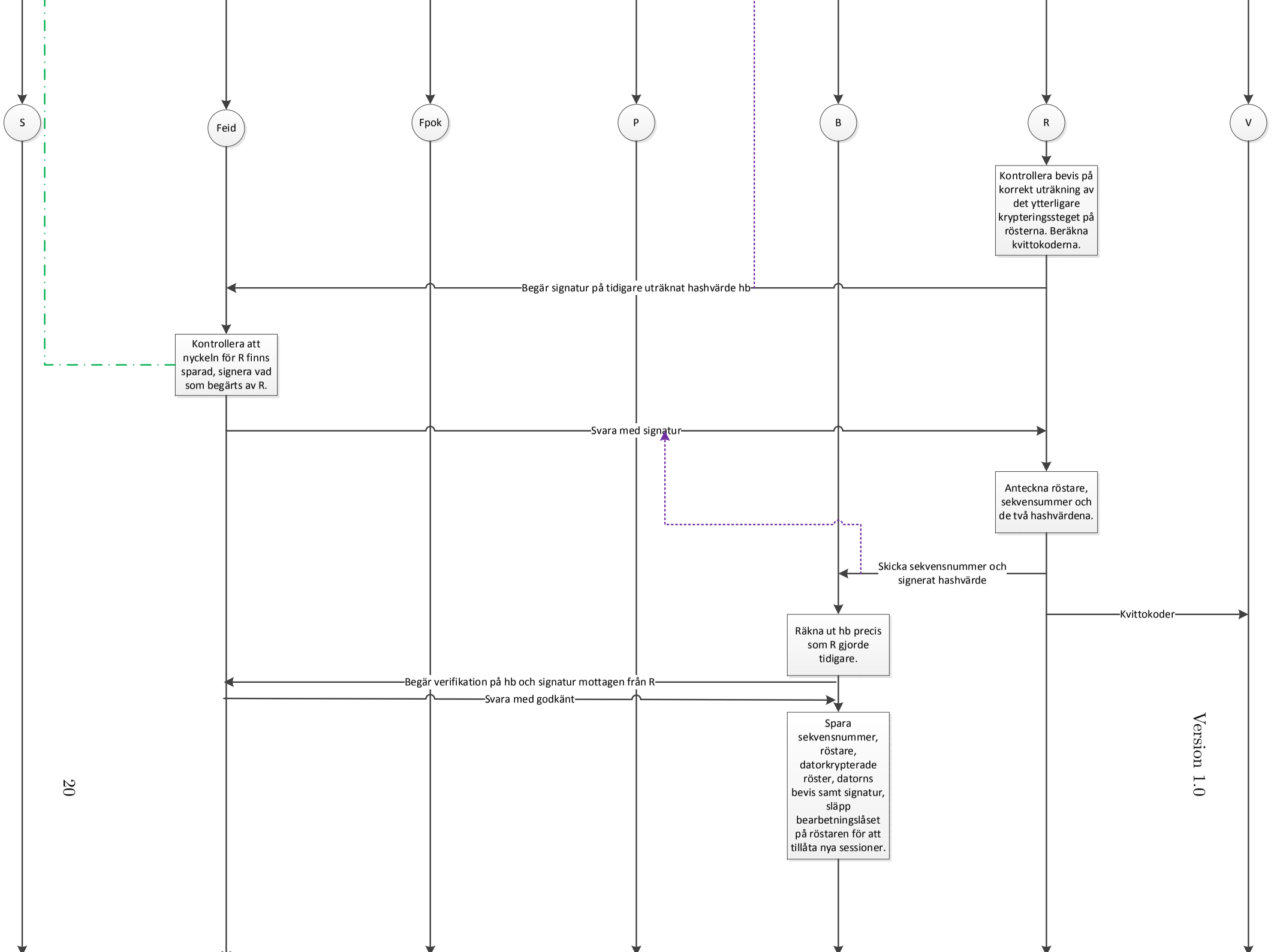
R - Kvittogenerator

V - Röstare


S
Feid
Fpok
P
B
R
V
Enhetsnycklar, signeringsfunktion
Bevisfunktioner
Berätta vilken dator som ska användas
Röstval
Berätta vilken röstare som är aktiv, specificera B
Kontrollera att användaren som rapporterats nyss har kontaktat.
Sessionsid, B, röstare och dator
Spara undan sessionsid ihop med röstare, dator och B.
Svara med sessionsid
Sessionsid, röstare, dator
Fyll ut val som inte specificerats med nollor, kryptera valen.
Vänta på att Feid skickar röster.
Begär bevis på att krypteringen utförts korrekt
Utför bevis, lagra utfört bevis.
Svara med bevis
Begär signatur på röstval och tillhörande bevis
Kontrollera att röstaren tidigare berättat att hon använder dator som nu begär signaturen samt att röstarens nyckel mottagits. Signera det som skickats av datorn, spara signaturen med tillhörande röstare, röstval och bevis. Spara en flagga som berättar att denna signering utförts vid rätt tidpunkt.


S
Feid
Fpok
P
B
R
V
Svara med signatur
Skicka röstbegäran med rösterna, sessionsid, bevis och signatur
Kontrollera att ett sessionsid med rätt röstare och dator är antecknat.
Kommer nu att vänta på signerat kvittohash från Feid.
Sessionsid, meddelandestorlek
Sessionsid, röstare, röstval, datorns bevis, signatur
Begär verifikation på det nyss mottagna
Kontrollera att röstarens nyckel är sparad, kontrollera att signaturen med flaggan aktiverad är sparad.
Svara med godkänd verifikation
Fråga om datorns bevis finns lagrat hos Fpok
Svara med lagrat bevis
Innan röstningsprocessen börjat ska denna enhet sparat undan röstare ihop med deras privata tal s. B ska nu slå upp detta par och låsa det för bearbetning så att inte multipla röstningsessioner pågår samtidigt.

V
R
B
P
Fpok
Feid
S
Fastslå nästa seq-tal. Beräkna det ytterligare krypteringssteget på rösterna. Utför bevis på att det ytterligare krypteringssteget utförts korrekt.
Skicka seq, röstare, röster krypterade av datorn, röster krypterade av B (med tillhörande bevis), datorns bevis och signaturen.
Beräkna hashvärde på röstare, röstval krypterade av dator, datorns bevis och signatur (hb). Beräkna ytterligare ett hashvärde utan signaturen (hb').
Liknande det som B nyss gjorde så låser R röstaren för bearbetning. Kontrollerar också att ingen session som tidigare utförts av samma röstare har ett högre seq-tal.
Skicka verifikationsbegäran på röstare, krypterade röstval, datorns bevis och signatur (precis som B tidigare gjorde)
Svara med godkänd verifikation
Fråga om datorns bevis finns lagrat hos Fpok
Svara med lagrat bevis


S
Feid
Fpok
P
B
R
V
Kontrollera bevis på korrekt uträkning av det ytterligare krypteringssteget på rösterna. Beräkna kvittokoderna.
Begär signatur på tidigare uträknat hashvärde hb
Kontrollera att nyckeln för R finns sparad, signera vad som begärts av R.
Svara med signatur
Anteckna röstare, sekvensummer och de två hashvärdena.
Skicka sekvensnummer och signerat hashvärde
Kvittokoder
Räkna ut hb precis som R gjorde tidigare.
Begär verifikation på hb och signatur mottagen från R
Svara med godkänt
Spara sekvensnummer, röstare, datorkrypterade röster, datorns bevis samt signatur, släpp bearbetningslåset på röstaren för att tillåta nya sessioner.

V
R
B
P
Fpok
Feid
S
Skicka bekräftelse på att allt gått rätt till
Räkna ut hb precis som både R och B har gjort tidigare.
Skicka R:s signatur
Skicka R:s signatur och sessionsid
Begär verifikation på hb och signatur mottagen indirekt från R
Svara med godkänt

3.4 Flödesdiagram illustrerande ihopräkning

Följande diagram visar hur rösterna sammanställs och hur valresultatet beräknas.

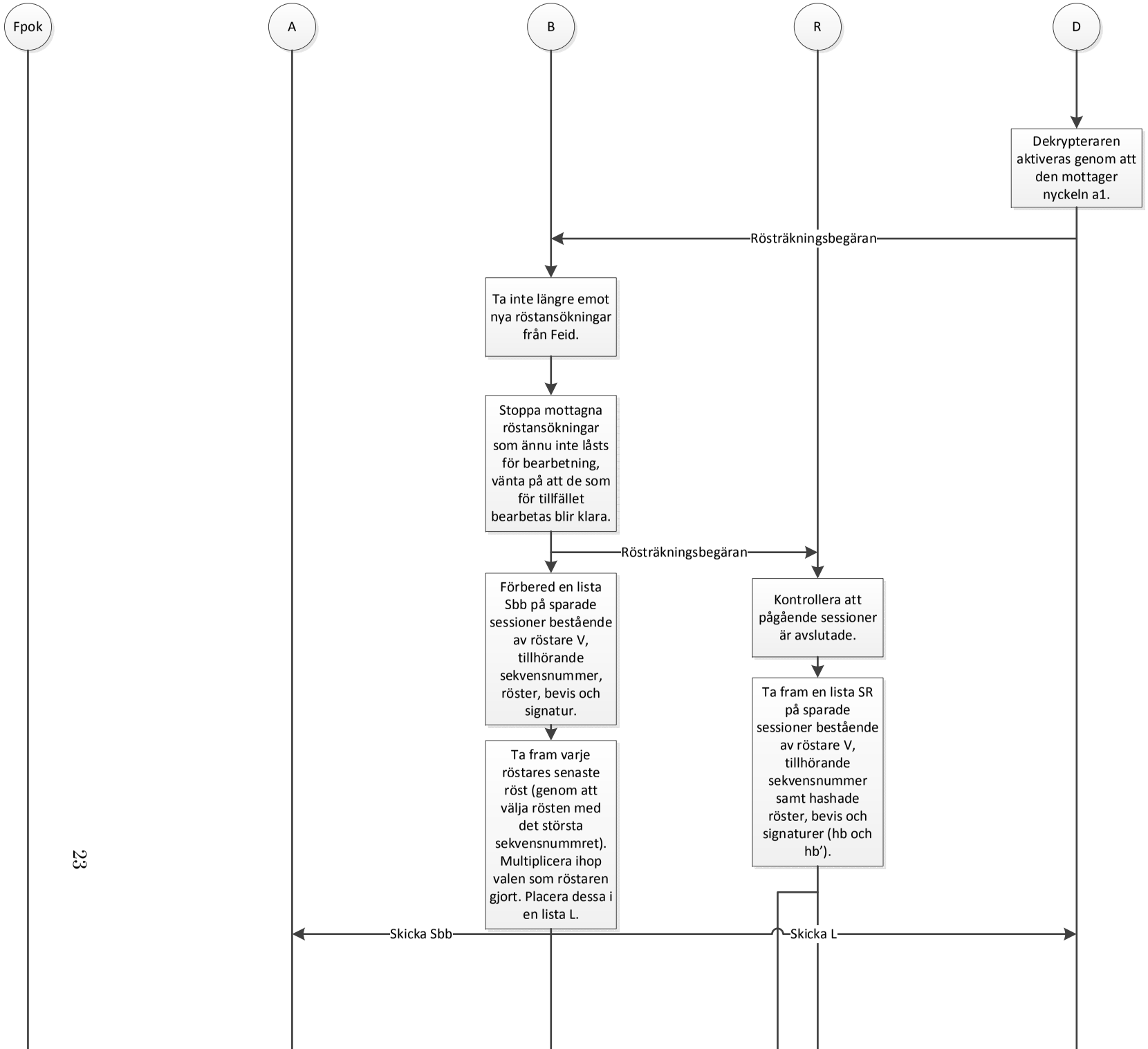
Fpok - Beviskontroll

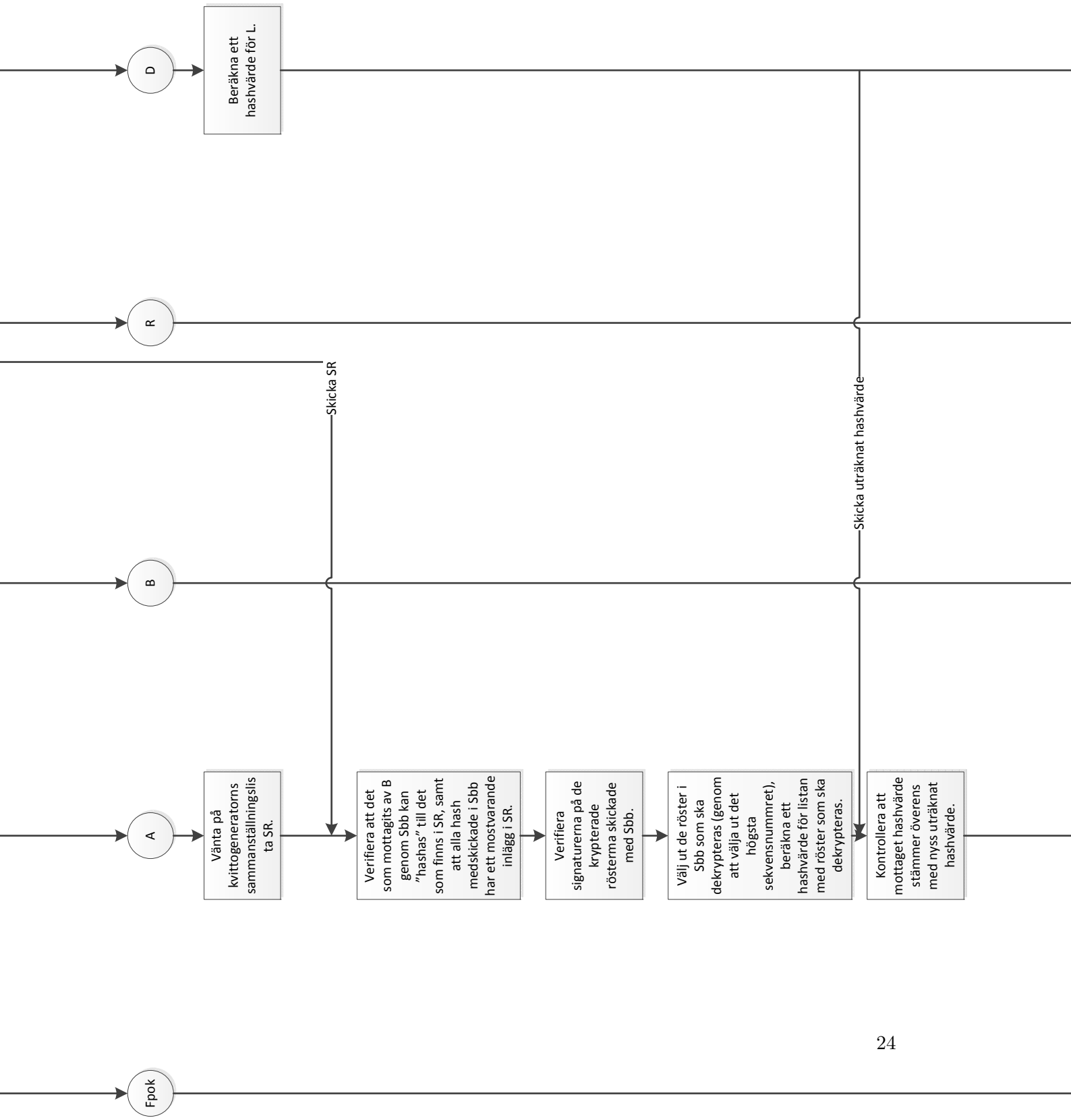
A - Inspektör

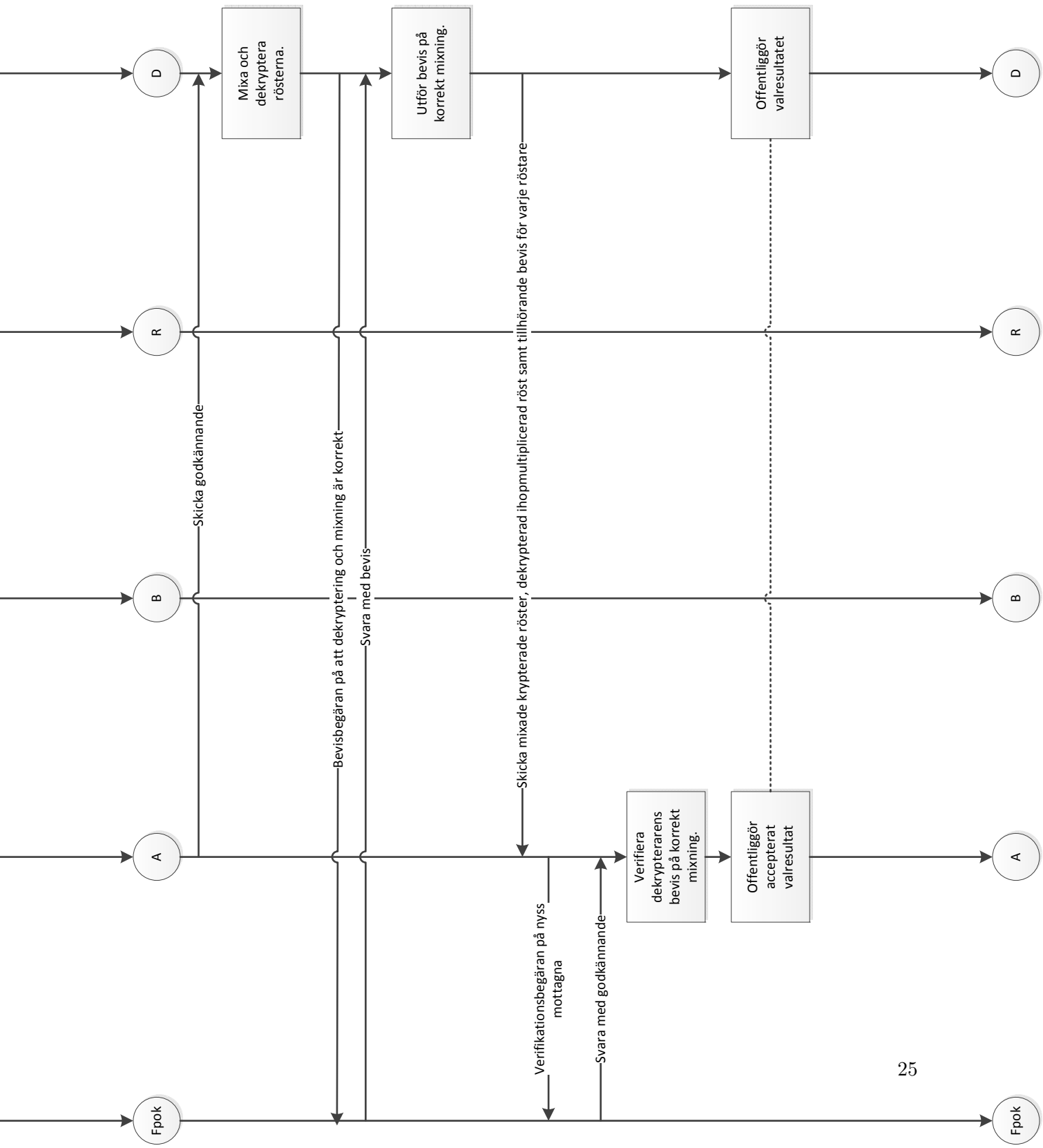
B - Röstsamlare

R - Kvittogenerator

D - Dekrypterare


Fpok
A
B
R
D
Dekrypteraren aktiveras genom att den mottager nyckeln a1.
Rösträkningsbegäran
Ta inte längre emot nya röstansökningar från Feid.
Stoppa mottagna röstansökningar som ännu inte låsts för bearbetning, vänta på att de som för tillfället bearbetas blir klara.
Rösträkningsbegäran
Förbered en lista Sbb på sparade sessioner bestående av röstare V, tillhörande sekvensnummer, röster, bevis och signatur.
Kontrollera att pågående sessioner är avslutade.
Ta fram en lista SR på sparade sessioner bestående av röstare V, tillhörande sekvensnummer samt hashade röster, bevis och signaturer (hb och hb').
Ta fram varje röstares senaste röst (genom att välja rösten med det största sekvensnummret). Multiplicera ihop valen som röstaren gjort. Placera dessa i en lista L.
Skicka Sbb
Skicka L


Fpok
A
Vänta på kvittogeneratorns sammanställningslista SR.
B
Skicka SR
R
Skicka uträknat hashvärde
D
Beräkna ett hashvärde för L.
Verifiera att det som mottagits av B genom Sbb kan ”hashas” till det som finns i SR, samt att alla hash medskickade i Sbb har ett mostvarande inlägg i SR.
Verifiera signaturerna på de krypterade rösterma skickade med Sbb.
Välj ut de röster i Sbb som ska dekrypteras (genom att välja ut det högsta sekvensnummret), beräkna ett hashvärde för listan med röster som ska dekrypteras.
Kontrollera att mottaget hashvärde stämmer överens med nyss uträknat hashvärde.


D
R
B
A
Fpok
Mixa och dekryptera rösterna.
Utför bevis på korrekt mixning.
Offentliggör valresultatet
Skicka godkännande
Bevisbegäran på att dekryptering och mixning är korrekt
Svara med bevis
Skicka mixade krypterade röster, dekrypterad ihopmultiplicerad röst samt tillhörande bevis för varje röstare
Verifiera dekrypterarens bevis på korrekt mixning.
Offentliggör accepterat valresultat
Verifikationsbegäran på nyss mottagna
Svara med godkännande

4 Diskussion

Föregående avsnitt presenterade information om teknisk bakgrund, centrala begrepp och själva protokollgenomgången. Nedan följer kommentarer kring styrkor och svagheter med det norska systemet och vilka aspekter det norska systemet valde att fokusera på.

Det norska systemet skickar kvittokoder via SMS till användaren. Kvittokoderna är knutna till de specifika val hon matat in i PCn. Detta medför att det norska systemet uppfyller voter verifiability och cast-as-inteded då användaren kan kontrollera om systemet registrerat rösten korrekt. Vidare så fungerar den skickade kvittokoden som en försäkran om att rösten kommer att vara med i ihopräkningen. Eftersom att kvittokoden skickats har ett antal enheter registrerat rösten och kan därmed anses vara tillräckligt sparat för att korrupta enheter ska få det väldigt svårt att famla bort rösten. När rösten väl har registrerats motiverar matematiska egenskaper tillsammans med flödet i systemet att rösten kommer att räknas, vilket gör det norska systemet till ett end-to-end-system.

Kvittokoderna medför en risk för att en röstares integritet kan kränkas. Om en tredje part känner till användarens valsedel med kvittokoderna och får se SMS:et så är också den tredje parten medveten om vad som röstats på. Systemet förutsätter att mobiltelefoner och brev är något privat. Vid exempelvis familjerelationer måste inte så vara fallet. Mer politiskt intresserade familjemedlemmar skulle kunna utöva press mot andra familjemedlemmar som i sin tur kanske röstar på grund av grupptrycksanledningar. På så sätt riskerar grupptryck att spela en större roll vid ett elektroniskt val som denna jämfört med den traditionella vallokalsröstningen.

Det norska systemet tillåter dock att en röstare röstar på traditionellt vis i en vallokal. Om en röstare väljer att pappersrösta ignoreras de elektroniska rösterna. Om röstaren då utsätts för press hemifrån kan personen låtsasröstapå utpressat alternativ och vid ett senare tillfälle pappersrösta på det röstaren egentligen vill. På så sätt kan det norska systemet argumentera för att den innehar egenskapen coercion-resistant. I och med

att den tillåter upprepad elektronisk röstning och dessutom erbjuder möjligheten att pappersrösta är det mycket svårt för en tredje part att förhindra någon att rösta på det dem vill. Detta förutsätter dock att den utpressade är tillräckligt politiskt intresserad för att anstränga sig och det mer eller mindre passiva grupptrycket kan fortfarande påverka.

Det norska systemet uppfyller heller inte universal-verifiability då enskilda röstare inte kan kontrollera valresultatet. Röstare måste lita på att den norska regeringen konstruerat systemet som beskrivet i de publicerade protokollen. Det kräver att norrmännen har högt förtroende för sin regering. Systemet skulle alltså vara mindre lämpligt i situationer där trovärdigheten hos auktoriteten kan ifrågasättas. Att norrmännen faktiskt har tillräckligt högt förtroende demonstreras av en förstudie som utfördes innan valet där en testomgång utan kvittokoder genomfördes. Majoriteten av deltagarna uppgav trots detta stort förtroende för systemet, även fast det då tappat sin end-to-end-egenskap [13, s. 24]. En annan laborationsstudie testade röstarnas förståelse för säkerheten i systemet ytterligare. De ombad bl. a. deltagarna i studien att rösta via en hemsida som uppmanade röstarna att mata in röstkoderna i datorn (!). Röstarna fick sedan två stycken sms skickade till sig, varav en innehållandes fel kvittokoder. Ingen av deltagarna i experimentet märkte att någonting inte var som det ska. Detta demonstrerar ytterligare att röstarna har ett närmast naivt förtroende i systemet [13, s. 31].

Cramer, Franklin, Schoenmakers och Young har designat [3] ett system som till skillnad från det norska innehar egenskapen universal verifiability. Den använder sig av anslagstavlor där röstare publicerar sina kodade röster. Krypteringen är homomorfisk. Röstare måste också skicka in nyckel nödvändig för dekryptering till en centralenhet.

Detta system uppfyller universal verifiability då samtliga röstare efter valet själva kan lägga ihop krypterade röster redovisade på de offentliga anslagstavlorna och dekryptera summeringen med hjälp av en specialnyckel som centralenheten publicerar vid valets slut. Denna summering kan sedan jämföras med det officiella valresultatet. På så sätt kan inte staten modifiera/borträkna röster bakom kulisserna vilket gör detta

system mer lämpligt om det finns anledning till att tvivla på ihopräkningsprocessen. Samtidigt har centralenheten som nämnt möjlighet att dekryptera enskilda röster. Detta kan lindras genom att stycka upp centralenheten i mindre enheter som individuellt ansvarar för en mindre del av röstinsamlingen.

Detta alternativa system skyddar dock inte mot utpressning på så sätt att det går att kontrollera om en person har röstat eller inte, eftersom att anslagstavlan innehåller en bekräftelse på att en röst lagts genom att den redovisar krypterade röster. Detta öppnar upp för möjligheten att sälja sin röst genom att lova att inte rösta.

Det norska systemet litar fullständigt på identitetskontrollen som tillförs av en tredje part. Om inte den kontrollen fungerar korrekt kommer systemet heller inte att fungera korrekt då röstare kan utge sig för att vara någon annan. Samma enhet tar också hand om dataintegritet vid överföring och är därför mycket kritisk. Det norska systemet klarar inte heller av att genomföra en valomgång om en intern enhet korrumperas. Detta kan ses som att det norska systemet är väldigt bräckligt och på så sätt inte uppfyller egenskapen robustness. Å andra sidan så bör identitetskontrollen kunna implementeras av ett välkänt och testat alternativ som redan används till identitetsverifiering vid exempelvis banktjänster. De interna enheterna kommer också att vara välskyddade och yttre påverkan svårgenomförd då de inte måste vara direktkopplade mot internet (enbart PCn som skickar in de krypterade rösterna via identitetskontrollen måste ske via internet).

5 Slutsats

Genom att läsa denna uppsats har läsaren förhoppningsvis kunnat förstå hur ett elektroniskt röstsystem, och framförallt det norska systemet, fungerar arkitekturmässigt, samt fått en god uppfattning om den matematik som används för att säkerställa konfidentialitet och sekretess. Alternativa implementationer har diskuterats för att ge en bättre förståelse om varför vissa implementationer valdes, t.ex. varför det norska systemet valde en mindre robust implementation gällande kvittensfrihet.

Den matematiska bakgrunden har även den genomförts med fokus på att göra uppsatsen tillgänglig. De förenklingar som har gjorts motiveras med att det som utlämnats eller förenklats inte har varit direkt relevant för att förklara hur rösterna går från röstare till ihopräkning. Samtidigt tappar vi möjligheten att diskutera de kryptografiska val som gjorts. Vidarearbete på uppsatsen skulle kunna tillföra ett diskussionsavsnitt om kryptering och exempelvis diskutera eventuella prestandaproblem med systemet till följd av valet av t.ex. primtalsstorlek.

Den överskådliga bild av det norska systemet (Sektion 3.3 och 3.4) har gjorts i syfte att förklara hur det norska systemet fungerar och den detaljnivå som dessa flödesdiagram innehåller är medvetet lagd på en nivå som inte tar upp några specifika detaljer om de enskilda enheterna. Detta beror dels på att det inte krävdes för att förklara hur systemet är uppbyggt och dels på grund av tidsbrist. En mer detaljerad vy över systemet kompletterat med fokus på enheter som ansvarar för dekryptering och verifikation skulle bidra till ytterligare förståelse för hur systemet fungerar, samt belysa eventuella matematiska eller arkitektoniska säkerhetsbrister. En utökning tordes då kunna vara att ifrågasätta systemets egenskaper och undersöka hur systemet reagerar på avsiktliga sabotageförsök.

Referenser

[1] Description of symmetric and asymmetric encryption. http://support.microsoft.com/kb/246071, Oct. 2007. Accessed: 2013-04-12.

[2] Ecrypt ii yearly report on algorithms and keysizes. http://www.ecrypt.eu.org/documents/D.SPA.17.pdf, 2010. Accessed: 2013-04-12.

[3] R. J. Cramer, M. Franklin, L. A. Schoenmakers, and M. Yung. Multi-authority secret-ballot elections with linear work. Technical report, Amsterdam, The Netherlands, The Netherlands, 1995.

[4] K. Gjøsteen. Analysis of an internet voting protocol. Cryptology ePrint Archive, Report 2010/380, 2010. http://eprint.iacr.org/.

[5] J. Hoffstein, J. Pipher, and J. Silverman. *An Introduction to Mathematical Cryptography*. Springer Publishing Company, Incorporated, 1 edition, 2008.

[6] S. Hohenberger. Lecture 10: More on proofs of knowledge. http://cs.jhu.edu/~susan/600.641/scribes/lecture10.pdf, Feb. 2007. [Unpublished lecture notes], Accessed: 2013-04-12.

[7] A. Juels, D. Catalano, and M. Jakobsson. Coercion-resistant electronic elections. Cryptology ePrint Archive, Report 2002/165, 2002. http://eprint.iacr.org/.

[8] R. Küsters, T. Truderung, and A. Vogt. Verifiability, Privacy, and Coercion-Resistance: New Insights from a Case Study. Technical report, University of Trier, 2011. Online available at http://infsec.uni-trier.de/publications/paper/KuestersTruderungVogt-TR-2011.pdf. Also appeared as Technical Report 2011/517, Cryptology ePrint Archive, 2011. http://eprint.iacr.org/2011/517/.

[9] M. Nandi, S. Popoveniuc, and P. L. Vora. Stamp-it: A method for enhancing the universal verifiability of e2e voting systems. In S. Jha and A. Mathuria, editors, *ICISS*, volume 6503 of *Lecture Notes in Computer Science*, pages 81–95. Springer, 2010.

[10] M. Orcutt. The states with the riskiest voting technology. http://www.technologyreview.com/news/506676/the-states-with-the-riskiest-voting-technology/?texas, Nov. 2012. Accessed: 2013-04-12.

[11] R. Rivest. Lecture notes 15 : Voting, homomorphic encryption. http://web.mit.edu/6.857/OldStuff/Fall02/handouts/L15-voting.pdf, Oct. 2002. [Unpublished lecture notes], Accessed: 2013-04-12.

[12] P. Y. A. Ryan and V. Teague. Pretty good democracy. In B. Christianson, J. A. Malcolm, V. Matyas, and M. Roe, editors, *Security Protocols Workshop*, volume 7028 of *Lecture Notes in Computer Science*, pages 111–130. Springer, 2009.

[13] I. S. G. Stenerud and C. Bull. When reality comes knocking norwegian experiences with verifiable electronic voting. In M. J. Kripp, M. Volkamer, and R. Grimm, editors, *Electronic Voting*, volume 205 of *LNI*, pages 21–33. GI, 2012.

[14] D. Wikström. *On the Security of Mix-nets and Related Problems*. Trita-NA. Institutionen för numerisk analys och datalogi (Stockholm), 2004.