Université de Strasbourg
UFR de Mathématiques & Informatique
Master 2 Calcul Scientifique & Sécurité Informatique

Mémoire de Stage

Directrice de Stage : V. Cortier, CNRS, LORIA
Lieu du Stage : LORIA, Equipe CASSIS

Analyse Formelle d'un Protocole de Vote

Cyrille Wiedling
<zell.cyrillew@cegetel.net>

Strasbourg, 1er Août 2011

Table des matières

Introduction

De nos jours, les protocoles ont une place prépondérante dans nos vies, parfois même sans que nous nous en rendions véritablement compte. Avec le développement des communications et des nouvelles technologies, les interactions humaines se font de plus en plus au travers de réseaux informatiques et non face à face. Dans la vie de tous les jours, il y a des protocoles pour quasiment tout : commander des marchandises par téléphone ou internet, effectuer des paris ou jouer au poker en ligne, réaliser des transactions bancaires ou même voter pour des élections. Personne ne pense vraiment à ces protocoles ; ils ont évolué au cours du temps, tout le monde sait comment les utiliser et, comme ils donnent satisfaction, peu de personnes s'intéressent encore vraiment à la manière dont ils fonctionnent réellement. La plupart des protocoles réalisent une série d'actions entre deux acteurs, mais une grande partie de ces protocoles nécessitent la présence de personnes pour assurer la sécurité de celles-ci. Après tout, enverrions-nous de l'argent à un étranger pour qu'il nous achète des biens ? Jouerions-nous au poker avec une personne si nous ne pouvions pas le voir mélanger et distribuer les cartes ? Accepterions-nous d'envoyer notre vote par courrier au Gouvernement sans être certain que notre anonymat soit préservé ? Il est naïf de croire que les utilisateurs d'un réseau informatique sont tous honnêtes. Il est même naïf de penser que les gestionnaires du réseau sont tous honnêtes. Il serait même naïf de penser que les concepteurs du réseau étaient tous honnêtes. Bien entendu, la grande majorité d'entre eux le sont, mais les quelques autres peuvent faire énormément de dégâts. En formalisant les protocoles, on peut examiner comment des participants malhonnêtes peuvent essayer de tricher et ainsi développer des protocoles robustes, même en présence de tricheurs.

Contexte

L'analyse formelle consiste, en partant d'une spécification d'un système, à le modéliser en langage mathématique, comme par exemple la modélisation d'une machine à café par un automate fini, avant d'en formaliser également les propriétés que l'on cherche à démontrer, puis d'en faire la vérification à l'aide de preuves mathématiques. Différentes techniques d'analyse formelle ont été développées pour des protocoles destinés au paiement, à l'authentification, etc... Dans l'étude formelle de ces protocoles, il y en a que l'on sait bien étudier, car ils utilisent des primitives simples comme le chiffrement, la signature ou le hachage. Pourtant, le problème de la sécurité de tels protocoles reste difficile car même sous de sévères contraintes, il reste indécidable [DLMS99].

Même sous un nombre restreint de sessions, ce problème est NP-complet [1] [RT03]. En particulier, il n'est envisageable de répondre à ce problème que pour un nombre très limité de sessions, le coût augmentant exponentiellement avec le nombre de celles-ci. (Les algorithmes ayant alors une fâcheuse tendance à ne plus terminer.) Plusieurs outils ont été développé néanmoins pour automatiser la vérification des protocoles comme, par exemple, AVISPA [ABB+05] ou ProVerif [Bla01]. Ces outils ont permis d'analyser de très nombreux protocoles (mails certifiés [AB03], protocole JFK (Just Fast Keying) [ABF04], ...) et de trouver des attaques.

Pourtant les protocoles de votes échappent souvent à de tels outils ou résultats généraux. En effet, leur modélisation formelle demande souvent l'utilisation de primitives plus complexes qui excluent l'utilisation des outils dont on dispose pour étudier le problème de leur sécurité. Il est donc nécessaire de réaliser des preuves directement à la main, comme dans les cas d'étude des protocoles de Fujioka, Okamoto et Ohta [DKR09], ou d'Helios [CS11], même si celles-ci sont souvent plus ardues.

Contributions

La contribution de ce mémoire concerne tout particulièrement le protocole de vote Norvégien mis en place très récemment [Gjo10], déjà implémenté et dont le code source est actuellement disponible sur Internet, apparemment encore en période de test sur de petites communes de Norvège. Nous proposons une modélisation formelle de ce protocole en pi-calcul appliqué [AF01] pour pouvoir en faire une analyse formelle et ainsi démontrer la confidentialité des votes. Comme les autres protocoles de vote, il échappe lui aussi à la vérification automatisée et c'est donc une preuve ad-mano, inspirée de celle réalisée sur Hélios [CS11], qui a été réalisée pour démontrer cette propriété dans un contexte légèrement simplifié vis-à-vis du protocole global. On fournit également une preuve de décidabilité pour la déduction associée à une théorie équationelle dérivée de celle du protocole de vote Norvégien. Ce résultat est intéressant dans la mesure où la décidabilité de la déduction est une technique de base pour mettre au point des procédures d'analyse automatique.

Le premier chapitre de ce mémoire décrira le protocole de vote dans sa globalité et présentera le schéma des protocoles qu'il met en action. Le deuxième chapitre sera consacré aux définitions importantes et nécessaires pour commencer à s'intéresser à la modélisation formelle des protocoles, aperçu de l'ensemble des notions qu'il aura été nécessaire de maîtriser pour ce stage. Le troisième chapitre introduira le pi-calcul appliqué ainsi que la modélisation formelle complète du protocole de vote Norvégien. Le quatrième chapitre portera sur le résultat de décidabilité de la déduction associée à une version dérivée de la théorie équationelle du protocole. Le cinquième chapitre exposera le travail principal de ce stage, la preuve de la confidentialité du protocole de vote dans une version simplifiée. Le sixième et dernier chapitre présentera rapidement l'outil ProVerif ainsi que son application au protocole Norvégien au travers de résultats qui en illustrent les avantages et les limites.

1. Un problème NP-complet est un problème de décision tel qu'il est possible de vérifier une solution en temps polynomial (La classe des problèmes vérifiant cette propriété est noté NP) et que ce problème est au moins aussi difficile que tous les autres problèmes de la classe NP.

Chapitre 1

Le Protocole Norvégien de Vote par Internet

Ce chapitre a pour but de présenter, au travers d'une approche assez générale, le concept de protocole dans un premier temps, ainsi que le protocole de vote Norvégien qui a été le sujet d'étude du stage.

1.1 Introduction aux Protocoles

Définition 1.1. [Sch96] Un *protocole* est une série d'étapes, impliquant deux ou plusieurs participants, conçue pour accomplir une tâche.

Cette définition est importante. Il y est question d'une « *série d'étapes* », c'est-à-dire qu'un protocole est une suite ordonnée d'étapes et qu'il a un début et une fin. Chaque étape doit être exécutée à son tour et aucune autre étape ne peut être exécutée avant que la précédente ne soit effectuée. L'expression « *deux ou plusieurs participants* » indique qu'il faut au moins deux personnes pour accomplir un protocole ; une personne isolée ne peut accomplir un protocole. Il est vrai qu'une personne seule peut réaliser une série d'étapes pour accomplir une tâche (par exemple, cuire un gâteau), mais ce n'est pas un protocole. En dernier lieu, l'expression « *conçu pour accomplir une tâche* » indique qu'un protocole doit faire quelque chose. Ce qui ressemble à un protocole mais qui n'accomplit pas une tâche n'est pas un protocole mais bien une perte de temps.

Exemple. Voici un exemple avec une variante du protocole de Needham-Schroeder [NS78]. Ce protocole est sensé fournir une authentification mutuelle entre deux parties communiquant sur un réseau.

$$\begin{aligned} A \longrightarrow B &: \{A, N_A\}_{pub(B)} \\ B \longrightarrow A &: \{N_A, N_B\}_{pub(A)} \\ A \longrightarrow B &: \{N_B\}_{pub(B)}. \end{aligned}$$

Comme on peut le voir, ce protocole réunit tous les « ingrédients » : il est constitué d'une série d'étapes, met en scène deux participants, Alice (A) et Bob (B), et accomplit une tâche puisqu'il permet aux participants de s'assurer qu'ils communiquent avec la bonne personne. La première étape consiste à l'envoi, d'Alice à Bob, d'un message

chiffré avec la clef publique de Bob contenant l'identité de l'expéditrice, Alice, et un nombre aléatoire, N_A. Après avoir reçu et déchiffré ce message grâce à sa clef privée, Bob envoie à Alice son nombre aléatoire N_A et un autre qu'il vient de générer, N_B. Le tout est chiffré avec la clef publique d'Alice. Alice déchiffre ce message avec sa clef privée, reconnait le nombre qu'elle a envoyé à Bob et lui renvoie son nombre aléatoire chiffré avec la clef publique de Bob afin que ce dernier puisse être certain qu'il communique bien avec Alice.

Malheureusement, ce protocole ne parvient pas toujours à réaliser sa tâche. En effet, il est vulnérable à une attaque de type « rencontre au milieu » où une personne peut en duper une autre sur son identité. Si un imposteur, Mallory (M), peut persuader Alice (A) d'initier une session avec elle, alors elle peut relayer le message à Bob (B) et le convaincre qu'il discute avec Alice. Regardons comment l'attaque se déroule :

$$
\begin{array}{llll}
A & \xrightarrow{\{A,N_A\}_{pub(M)}} & M & \\
 & & M & \xrightarrow{\{A,N_A\}_{pub(B)}} \quad B \\
 & & M & \xleftarrow{\{N_A,N_B\}_{pub(A)}} \quad B \\
A & \xleftarrow{\{N_A,N_B\}_{pub(A)}} & M & \\
A & \xrightarrow{\{N_B\}_{pub(M)}} & M & \\
 & & M & \xrightarrow{\{N_B\}_{pub(B)}} \quad B
\end{array}
$$

En premier lieu, Mallory a convaincu Alice d'initier une conversation avec elle, ainsi Alice envoie à Mallory un message chiffré avec la clef publique de Mallory, contenant son identité A, ainsi qu'un nombre aléatoire N_A. Mallory, dont le but est de se faire passer pour Alice auprès de Bob, va déchiffrer le message et l'envoyer à Bob après l'avoir chiffré avec la clef publique, $pub(B)$, de ce dernier. Bob, qui n'est au courant de rien, va déchiffrer le message et penser qu'Alice désire communiquer avec lui. Il va donc suivre le protocole et renvoyer à Mallory, qu'il pense être Alice, un message contenant N_A et son nombre N_B chiffré avec la clef publique d'Alice. Mallory, qui ne peut déchiffrer ce message, le renvoie tel quel à Alice. Alice reçoit finalement un message qu'elle peut déchiffrer dans lequel elle reconnaîtra N_A et un autre N_B qu'elle pensera être le nombre de Mallory. Elle va donc lui renvoyer N_B chiffré avec la clef publique $pub(M)$. Mallory va alors pouvoir découvrir N_B et l'envoyer à Bob, chiffré avec sa clef publique, $pub(B)$. A la fin de cette attaque, Bob croit, à tort, qu'Alice est entrain de communiquer avec lui et que N_A et N_B ne sont connus que d'eux seuls. Alice elle, sait qu'elle est entrain de communiquer avec Mallory mais elle ne sait pas que cette dernière a dérobé son identité auprès de Bob.

Il est intéressant de noter que cette attaque a mis 17 années à être découverte. C'est Lowe [Low95] qui a corrigé ce protocole en modifiant la seconde étape en y rajoutant l'identité de Bob :

$$
\begin{array}{lcl}
A \longrightarrow B & : & \{A, N_A\}_{pub(B)} \\
B \longrightarrow A & : & \{B, N_A, N_B\}_{pub(A)} \\
A \longrightarrow B & : & \{N_B\}_{pub(B)}.
\end{array}
$$

1.2 Présentation Générale

Le protocole de vote Norvégien est présenté [1] dans [Gjo10]. Actuellement en phase de test sur plusieurs municipalités pour des élections locales, il est possible qu'il soit déployé au cours de l'année 2012 s'il donne entière satisfaction. Il a été créé par `Scytl`, une compagnie espagnole du vote électronique. Il est sensé permettre à un électeur d'utiliser son propre ordinateur personnel pour soumettre un bulletin de vote électronique à un bureau de vote virtuel. L'utilisateur peut également revoter s'il le désire (par exemple s'il change d'avis ou s'il y a eu un problème durant la procédure), dans ce cas, seul le dernier vote est prit en compte. Il est normalement prévu par le système que l'électeur puisse également voter de manière classique au bureau de vote, son vote papier étant alors prédominant sur tous ses votes électroniques, même si ceux-ci ont eu lieu après la soumission du vote papier.

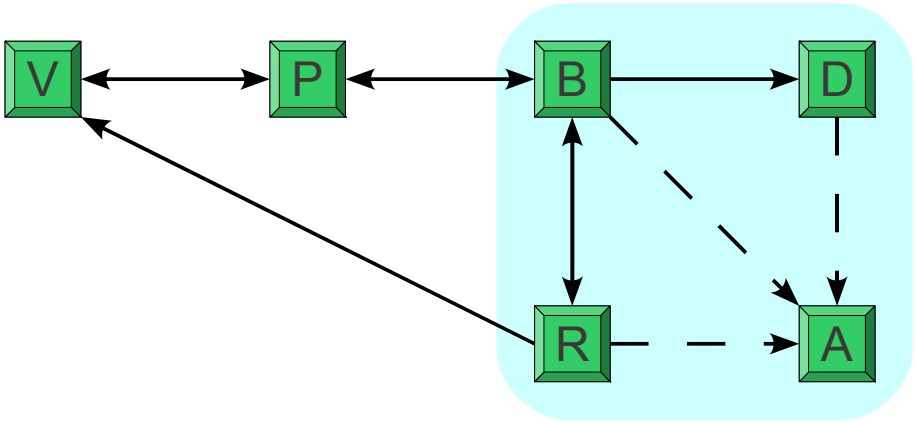


FIGURE 1.1 – Représentation des différents participants du protocole.

Plus précisément, ce protocole met en scène un votant (V) qui utilise son ordinateur (P) pour soumettre son vote à une infrastructure mise en place pour l'élection. Cette infrastructure est composée de plusieurs participants comme le montre la figure 1.1. On retrouve dans l'encart bleu clair les quatre participants : l'urne (B), qui va stocker les différents votes, comme une urne classique, le générateur de reçus (R), le déchiffreur (D) et enfin l'auditeur (A) dont le rôle est de s'assurer, a posteriori, que ni l'urne, ni le générateur de reçus, ni le déchiffreur n'ont triché.

1.3 Protocole et Descriptions

On retrouve ci-dessous (Figure 1.2), le schéma simplifié des échanges des différents messages entre les participants lors de la procédure de soumission d'un vote par un

1. Des informations complémentaires peuvent être trouvée sur le site internet **www.evalg.dep.no**.

électeur, phase qui met en jeu le votant, son ordinateur, l'urne et le générateur de reçus.

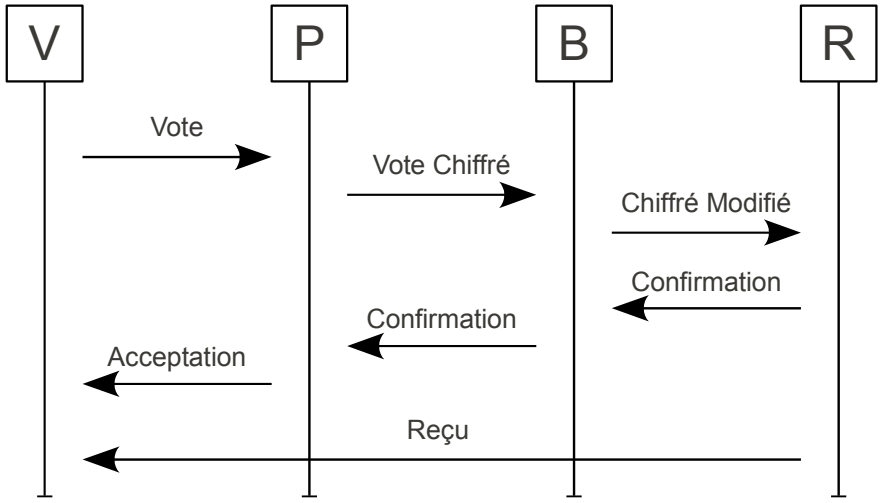


FIGURE 1.2 – Echanges de messages durant la phase de soumission d'un vote.

Le votant va donner à son ordinateur le vote qu'il désire effectuer - en l'entrant dans un champ particulier sur un site web sécurisé par exemple - et va autoriser son ordinateur à effectuer le reste du protocole pour lui. L'ordinateur va se charger de chiffrer le vote puis de l'expédier à l'urne. Cette dernière va stocker le vote, après quelques vérifications, puis va générer un autre message à partir du vote chiffré à destination du générateur de reçus. Après quelques vérifications sur ce message, le générateur de reçus produit les fameux reçus qu'il va expédier au votant, via, par exemple, SMS. En même temps, il fournit une confirmation de la prise en compte du vote à l'urne qui la transmet à son tour, après vérification, au votant. Ce dernier reçoit donc deux messages : une confirmation provenant de l'urne lui indiquant que son vote a été pris en compte et un reçu du générateur de reçus qui va lui permettre de vérifier que le vote prit en compte est le bon. (Le votant dispose d'une table de correspondance (unique par votant) entre les différents votes possibles et les reçus équivalents, lui permettant ainsi de comparer son vote au reçu réceptionné.)

Cette étape est répétée pour chaque votant désirant soumettre son vote, ou même plusieurs fois par votant si celui-ci désire revoter dans le but de changer son vote ou s'il y a eu une erreur lors du processus. (Par exemple, le reçu réceptionné n'est pas le bon.) Une fois que le temps de vote est écoulé, le protocole passe à la phase de décompte qui fait intervenir l'urne, le générateur de reçus, le déchiffreur et l'auditeur. (Voir Figure 1.3.)

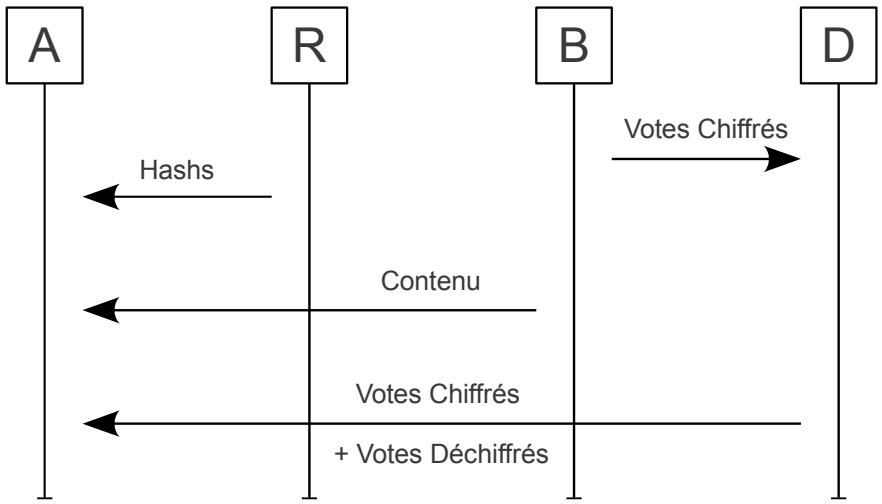


FIGURE 1.3 – Echanges de messages durant la phase de décompte final.

L'urne, en premier lieu, va envoyer l'ensemble des votes chiffrés au déchiffreur puis va fournir à l'auditeur l'ensemble de son contenu, c'est-à-dire ces mêmes votes. Le générateur de reçus va envoyer à l'auditeur l'ensemble des hashs des votes qui lui ont été soumis par l'urne durant la phase de soumission des votes. Ainsi, l'auditeur peut comparer le contenu du générateur de reçus et de l'urne et vérifier qu'aucun vote n'a été ajouté ou supprimé dans l'urne sans que le générateur de reçus ne soit mis au courant. Le déchiffreur effectue ensuite son travail et envoie les votes chiffrés et déchiffrés à l'auditeur. Ainsi, en possession du contenu chiffré de l'urne, l'auditeur est en mesure de vérifier que le déchiffreur a bien fait son travail sans supprimer ou ajouter de votes. Une fois toutes ces confirmations effectuées, l'auditeur confirme que l'élection s'est bien déroulée et publie le résultat.

Remarque. Pour éviter qu'un lien soit possible entre l'ordre de soumission des votes et l'ordre de publication à la sortie du déchiffreur, l'urne envoie les différents votes chiffrés dans un ordre aléatoire, tout comme le déchiffreur avec les votes déchiffrés. Ainsi, il n'existe aucun rapport entre l'ordre des votants, à l'entrée, et l'ordre des résultats, à la sortie.

Chapitre 2

Préliminaires

La première étape pour la modélisation des protocoles est de modéliser les messages qu'ils manipulent. Cette modélisation est faite en utilisant des *termes*. Le but de ce chapitre est d'introduire les définitions nécessaires pour cette modélisation. Des définitions plus spécifiques pourront être incorporées dans les chapitres adéquats. Toutes ces notes proviennent de [BN98], [Cor11] et [CJ97].

2.1 Termes et *Associés*

2.1.1 Termes

La notion de terme fonde tout calcul symbolique, en particulier l'interprétation des langages de programmation évolués.

Définition 2.1. Une *signature* est un couple $(\mathcal{F}, Arite)$ où $\mathcal{F}$ est un ensemble fini et $Arite$ est une fonction de $\mathcal{F}$ dans $\mathbb{N}$. L'*arité* d'un symbole $f \in \mathcal{F}$ est $Arite(f)$. L'ensemble des symboles d'arité $p \geq 0$ de $\mathcal{F}$ est dénoté $\mathcal{F}_p$. Les éléments d'arité $0, 1, \ldots, p$ sont respectivement appelés constantes, symboles unaires, binaires, ..., symboles p-aire. On suppose que $\mathcal{F}$ contient au moins une constante.

Dans la suite, on utilisera les parenthèses et les virgules pour une représentation raccourcie des symboles avec leur arité. Par exemple, $f(,)$ est un raccourci pour le symbole binaire f.

Exemple. $\mathcal{F} = \{a, b, h, enc\}$, où a et b sont des constantes ($Arite(a) = Arite(b) = 0$), h est un symbole unaire, la fonction de hachage, ($Arite(h) = 1$) et enc est un symbole binaire, la fonction de chiffrement ($Arite(enc) = 2$).

Définition 2.2. Soit $\mathcal{X}$ un ensemble de constantes appelées *variables*. On suppose que que les ensembles $\mathcal{X}$ et $\mathcal{F}$ sont disjoints. L'ensemble $T(\mathcal{F}, \mathcal{X})$ des *termes* sur la signature $\mathcal{F}$ et l'ensemble des variables $\mathcal{X}$ est le plus petit ensemble défini par :

- $\mathcal{F}_0 \subseteq T(\mathcal{F}, \mathcal{X})$,
- $\mathcal{X} \subseteq T(\mathcal{F}, \mathcal{X})$ et
- Si $p \geq 1$, $f \in \mathcal{F}_p$ et $t_1, \ldots, t_p \in T(\mathcal{F}, \mathcal{X})$ alors $f(t_1, \ldots, t_p) \in T(\mathcal{F}, \mathcal{X})$ (i.e le résultat de l'application d'une fonction à des termes est également un terme).

La structure d'un terme peut être illustrée graphiquement, en la représentant sous la forme d'un arbre où noeuds sont des symboles de fonctions et les fils de chaque noeud sont les arguments de la fonction représentée. Par exemple, en utilisant la signature $\mathcal{F}$ donnée dans l'exemple ci-dessus, le terme $enc(enc(h(x), b), a)$ qui contient la variable x peut être visualisé comme un arbre comme le montre la figure 2.1.

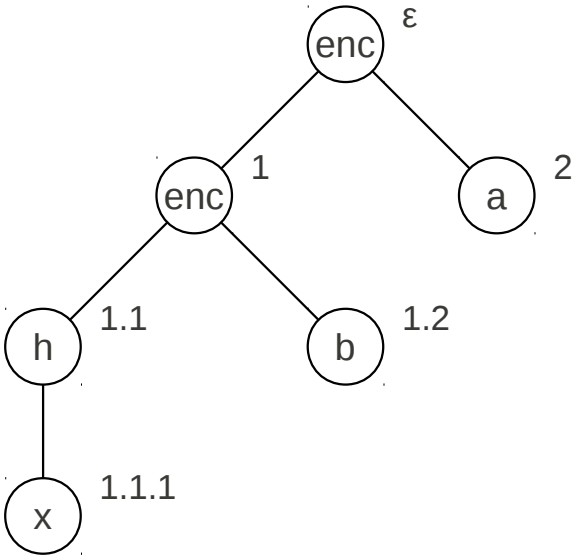


FIGURE 2.1 – Représentation sous forme d'arbre du terme $t = enc(enc(h(x), b), a)$.

En utilisant une numérotation standard des noeuds de l'arbre par des suites finies d'entiers naturels non nuls (comme illustré dans l'exemple), on peut alors utiliser la notion de position dans un terme. Dans notre exemple, la position ϵ (la suite vide, ou la racine du terme) fait référence au symbole *enc* au sommet de l'arbre et la position 1 au symbole *enc* qui apparait comme premier argument du *enc* initial. Le sous-terme de t à la position 1 est $enc(h(x), b)$ et celui à la position 1.1 est $h(x)$. On peut définir plus formellement les définitions de positions et de sous-termes par récurrence sur la structure des termes.

Définition 2.3. Soit $\mathcal{F}$ une signature, $\mathcal{X}$ un ensemble de variables disjoint de $\mathcal{F}$, et deux termes $s, t \in T(\mathcal{F}, \mathcal{X})$. L'ensemble des *positions* d'un terme s est un ensemble $Pos(s)$ de suites sur l'alphabet des entiers naturels, défini de la manière suivante :

- Si $s = x \in \mathcal{X}$, alors $Pos(s) = \{\epsilon\}$, où ϵ est la suite vide.
- Si $s = f(s_1, \ldots, s_n)$, alors

$$Pos(s) = \{\epsilon\} \cup \bigcup_{i=1}^{n} \{i \cdot p \mid p \in Pos(s_i)\}.$$

La position ϵ est appelée *position racine* du terme s et le symbole de fonction ou de variable à cette position est le *symbole racine* de s.

2.1.2 Sous-termes, taille et termes clos

Définition 2.4. Pour $p \in Pos(s)$, le *sous-terme de s à la position p*, noté $s|_p$, est défini par récurrence sur la longueur de p :

$$s|_\epsilon = s,$$
$$f(s_1, \ldots, s_n)|_{i \cdot q} = s_i|_q.$$

Notons que, pour $p = i \cdot q$, $p \in Pos(s)$ on a forcément s de la forme $s = f(s_1, \ldots, s_n)$ avec $i \leq n$.

Définition 2.5. Pour $p \in Pos(s)$, on note $s[t]_p$ le terme obtenu à partir de s en *remplaçant le sous-terme à la position p par t*, i.e.

$$s[t]_\epsilon = t,$$
$$f(s_1, \ldots, s_n)[t]_{i \cdot q} = f(s_1, \ldots, s_i[t]_q, \ldots, s_n).$$

Définition 2.6. La *taille* $|s|$ d'un terme s est définie par récurrence de la manière suivante :

$$|a| = 1, \quad \text{si } a \in \mathcal{F}_0 \cup \mathcal{X},$$
$$|f(s_1, \ldots, s_n)| = 1 + \sum_{i=1}^{n} |s_i|.$$

Définition 2.7. $Var(s)$ est l'ensemble des *variables apparaissant dans s*, i.e.

$$Var(s) = \{x \in \mathcal{X} \mid \text{il existe } p \in Pos(s) \text{ tel que } s|_p = x\}.$$

On dit que $p \in Pos(t)$ est une *position de variable* si $t|_p$ est une variable.

Exemple. Pour le terme t du précédent exemple, $t = enc(enc(h(x), b), a)$, $Pos(t) = \{\epsilon, 1, 2, 1.1, 1.2, 1.1.1\}$, $t|_{1.1} = h(x)$, $t[a]_1 = enc(a, a)$, $Var(t) = \{x\}$ et $|t| = 6$.

Définition 2.8. Soit $\mathcal{F}$ une signature et $\mathcal{X}$ un ensemble de variables disjoint de $\mathcal{F}$. Un terme $t \in T(\mathcal{F}, \mathcal{X})$ est un terme *clos* ssi $Var(t) = \emptyset$. L'ensemble de tous les termes clos de $\mathcal{F}$ est noté $T(\mathcal{F})$ (ou $T(\mathcal{F}, \emptyset)$).

2.1.3 Substitutions

La principale différence entre les symboles de constantes et ceux de variables est que les derniers peuvent être remplacés par substitutions.

Définition 2.9. Une *substitution* σ est une fonction définie sur un sous-ensemble fini (appelé *domaine*) de l'ensemble des variables $\mathcal{X}$, noté $dom(\sigma)$, à valeurs dans $T(\mathcal{F}, \mathcal{X})$. Les substitutions sont étendues à tous les termes de $T(\mathcal{F}, \mathcal{X})$ de la manière suivante :

$$\sigma(x) = \qquad x, \text{ if } x \notin dom(\sigma),$$
$$\sigma(f(t_1, \ldots, t_n)) = \qquad f(\sigma(t_1), \ldots, \sigma(t_n)).$$

On écrit souvent $t\sigma$ à la place de $\sigma(t)$.

2.2 Théories Équationelles

Les symboles définis dans les signatures ne servant qu'à représenter les fonctions considérées, il faut également détailler leurs propriétés. Pour cela, on introduit les théories équationelles qui vont modéliser ces propriétés au travers d'équations.

Définition 2.10. Une *théorie équationelle* E est un ensemble d'équations $u = v$ où u et v sont des termes avec ou sans variables.
L'égalité sur les termes, notée $=_E$, induite par une théorie équationelle, est la plus petite relation d'équivalence telle que :

- $u\theta =_E v\theta$, pour toute substitution θ,
- $(u_1 =_E v_1, \cdots, u_k =_E v_k) \Longrightarrow f(u_1, \ldots, u_k) =_E f(v_1, \ldots, v_k)$, pour tout $f \in \mathcal{F}$.

Remarque. Dans la figure suite, $\langle x, y\rangle$ correspond à $\mathsf{pair}(x, y)$ et $\{x\}_y$ à $\mathsf{enc}(x, y)$ où x est le message clair et y la clef. Il s'agit là de notations usuelles qui seront être réutilisées plus tard dans le papier.

Exemple. Voici la théorie équationelle E_{dec} correspondant à l'usage d'une fonction spécifique pour le déchiffrement, dec, et deux autres, π_1 et π_2, utilisées pour obtenir un élément d'une paire :

$$E_{\mathsf{dec}} = \left\{ \begin{array}{rcl} \mathsf{dec}(\{x\}_y, y) & = & x, \\ \pi_1(\langle x, y\rangle) & = & x, \\ \pi_2(\langle x, y\rangle) & = & y \end{array} \right\}.$$

La première équation modélise le déchiffrement et stipule que déchiffrer le message $\{x\}_y$ (chiffré avec la clef y) en utilisant la clef y permet de récupérer le message x. Les deux autres équations modélisent la projection d'une paire selon sa première ou sa seconde composante.

Exemple. Voici la théorie équationelle $E_{\oplus}$ pour la fonction du Ou eXclusif (XOR) :

$$E_{\oplus} = \left\{ \begin{array}{rcl@{\qquad}rcl} x \oplus (y \oplus z) & = & (x \oplus y) \oplus z, & x \oplus y & = & y \oplus x, \\ x \oplus x & = & 0, & x \oplus 0 & = & x \end{array} \right\}.$$

On modélise ici les propriétés associative et commutative du symbole $\oplus$ ainsi que comparaison d'égalité et la neutralité du 0.

Exemple. Il peut arriver que les fonctions de chiffrement et déchiffrement soient identiques. Dans ce cas, on considère la théorie équationelle formée de l'équation :

$$\{\{x\}_y\}_y = x \quad \text{ou} \quad \mathsf{enc}(\mathsf{enc}(x, y), y) = x.$$

On peut distinguer deux types de théorie : AC ou non-AC. Si la théorie contient un symbole AC (Associatif et Commutatif), comme $\oplus$, $+$, ..., alors la théorie est considérée comme une *théorie AC*. Par exemple, $E_{\oplus}$ est une théorie AC alors que E_{dec} ne l'est pas.

Définition 2.11. Dans le cadre de théorie AC, on définit une *égalité modulo AC*, notée $=_{AC}$ qui une égalité syntaxique classique modulo les équations associatives et commutatives de la théorie considérée.

Exemple. Si l'on reprend la théorie $E_{\oplus}$ de l'exemple précédent, on a $x \oplus y \neq y \oplus x$ (l'égalité syntaxique des deux termes est fausse) mais $x \oplus y =_{AC} y \oplus x$.

2.3 Systèmes de Réécriture

Pour faciliter la manipulation des théories équationelles, on leur associe souvent un système de réécriture.

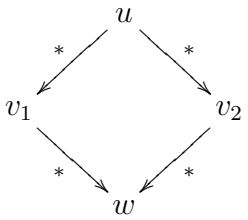
Définition 2.12. Une *règle de réécriture* est une règle de la forme $l \longrightarrow r$ où l et r sont des termes.
Un terme s se *réécrit* en t pour la règle $l \longrightarrow r$, noté $s \xrightarrow{l \longrightarrow r} t$, s'il existe une position p dans s et une substitution θ telle que $s|_p = l\theta$ et $t = s[r\theta]_p$.
Un *système de réécriture* est un ensemble $\mathcal{R}$ de règles de réécriture.

Définition 2.13. Étant donné un sytème de réécriture $\mathcal{R}$, on dit qu'un terme t est en $\mathcal{R}$-*forme normale* s'il n'existe pas de terme s tel que $t \longrightarrow_{\mathcal{R}} s$. Si $t \longrightarrow_{\mathcal{R}}^{*} s$ (i.e. t se réécrit en s en utilisant une ou plusieurs règles de réécriture de $\mathcal{R}$) et s est en $\mathcal{R}$-forme normale, alors s est une $\mathcal{R}$-forme normale de t.

Remarque. On pourra omettre de préciser quel est le système de réécriture considéré quand celui-ci se déduit aisément du contexte.

On définit plusieurs propriétés pour les systèmes de réécriture.

Définition 2.14. Un système de réécriture $\mathcal{R}$ est dit *confluent* si, pour tous termes u, v_1, v_2, si $u \longrightarrow^* v_1$ et $u \longrightarrow^* v_1$, alors il existe w tel que $v_1 \longrightarrow^* w$ et $v_2 \longrightarrow^* w$.



En particulier, une telle paire (v_1, v_2) est appelée *paire critique*. Un système de réécriture est donc confluent si toutes ses paires critiques se rejoignent.

Définition 2.15. Un système de réécriture $\mathcal{R}$ est dit *terminant* s'il n'existe pas de suite infinie de réécriture :

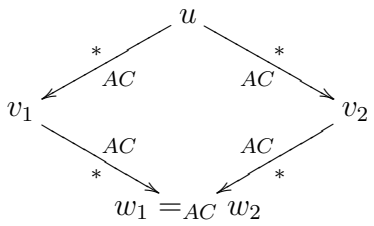
$$u_1 \longrightarrow u_2 \longrightarrow \cdots \longrightarrow u_n \longrightarrow \cdots .$$

Définition 2.16. Un système de réécriture $\mathcal{R}$ est dit *convergent* s'il est confluent et terminant. Si $\mathcal{R}$ est convergent, alors pour tout terme t il y a une et une seule $\mathcal{R}$-forme normale, notée $t{\downarrow}$.

Comme il est impossible d'associer un système de réécriture fini et convergent aux équations d'associativité et de commutativité (AC), on considère, pour ces théories AC, la réécriture de termes modulo AC.

Définition 2.17. Un terme s est *réécrit modulo AC* en t pour la règle $l \longrightarrow r$, noté $s \xrightarrow{l \longrightarrow r}_{AC} t$, s'il existe s' et t' des termes tels que $s =_{AC} s'$, $t =_{AC} t'$ et s' est réécrit en t' pour la règle $l \longrightarrow r$.

Définition 2.18. Un système de réécriture $\mathcal{R}$ est dit *AC-confluent* si, pour tous termes u, v_1, v_2, si $u \longrightarrow^*_{AC} v_1$ et $u \longrightarrow^*_{AC} v_1$, alors il existe w_1, w_2 tels que $v_1 \longrightarrow^*_{AC} w_1$, $v_2 \longrightarrow^*_{AC} w_2$ et $w_1 =_{AC} w_2$.



Définition 2.19. Un système de réécriture $\mathcal{R}$ est dit *AC-terminant*, s'il n'existe pas de suite infinie de réécriture :

$$u_1 \longrightarrow_{AC} u_2 \longrightarrow_{AC} \cdots \longrightarrow_{AC} u_n \longrightarrow_{AC} \cdots .$$

Définition 2.20. Un système de réécriture $\mathcal{R}$ est dit *AC-convergent* s'il est AC-confluent et AC-terminant.

2.4 Systèmes d'Inférence

En plus de la modélisation des messages, on veut également modéliser ce qu'un intrus est capable de calculer à partir de certains messages qui seraient en sa possession. C'est le but des systèmes d'inférence qui vont regrouper l'ensemble des possibilités d'un intrus sous forme de règles.

Définition 2.21. Une *règle d'inférence* est une règle de la forme $\frac{T_1 \ \cdots \ T_n}{T}\gamma$ avec $T_1, \ldots, T_n, T \in T(\mathcal{F}, \mathcal{X})$ et γ une condition tels que si quelqu'un connait $T_1, \ldots, T_n$ et vérifie la condition γ alors il peut obtenir T.
Un *système d'inférence* est un ensemble de règles d'inférence.

Exemple. Le système d'inférence correspondant aux capacités d'un adversaire symbolique classique (appelé Dolev-Yao) est représenté en figure 2.2.

$$\mathcal{I}_{DY} = \left\{ \begin{array}{lll} & \dfrac{x \quad y}{\langle x,y \rangle} & \dfrac{x \quad y}{\{x\}_y} \\ \dfrac{\langle x,y \rangle}{x} & \dfrac{\langle x,y \rangle}{y} & \dfrac{\{x\}_y \quad y}{x} \end{array} \right.$$

FIGURE 2.2 – Système d'inférence $\mathcal{I}_{DY}$ correspondant à un adversaire Dolev-Yao.

Dans ce système d'inférence, on autorise l'intrus à réaliser une paire ou d'effectuer un chiffrement à partir de deux éléments connus, ce qui est logique puisque rien ne lui interdit d'assembler des éléments ou d'en chiffrer avec des clefs qu'il possède. Les trois autres règles stipulent qu'il peut récupérer l'un ou l'autre des éléments d'une paire et que, s'il dispose d'un message chiffré et de la clef adéquate, alors il est en mesure de déchiffrer le message.

Définition 2.22. Un terme t est *dérivable en une étape* à partir d'un ensemble de termes S pour le système d'inférence $\mathcal{I}$, noté $S \vdash^1_{\mathcal{I}} t$, s'il existe une règle d'inférence $\frac{T_1 \ \cdots \ T_n}{T}\gamma$, $t_1, \ldots, t_n \in S$ et une substitution θ telle que :

$$t_i = T_i\theta,\ t = T\theta \text{ et } \gamma\theta = \mathsf{true}.$$

Un terme t est *dérivable* à partir d'un ensemble de termes S, noté $S \vdash_{\mathcal{I}} t$, si :

- $t \in S$, ou
- Il existe $t_1, \ldots, t_n$ tels que $t_n = t$ et t_{i+1} est dérivable en une étape à partir de $S \cup \{t_1, \ldots, t_i\}$. La suite $t_1, \ldots, t_n$ est appelée *preuve* de dérivabilité de $S \vdash_{\mathcal{I}} t$.

Exemple. Considérons l'ensemble $S = \left\{ \left\{ \{k_1\}_{k_1} \right\}_{k_2}, \{s\}_{k_1}, \{k_2\}_{\langle k_3, k_4 \rangle}, \{k_4\}_{k_3}, k_3 \right\}$ représentant le point de départ de la « connaissance » de l'intrus. Voyons ce qu'il peut en déduire :

1. $k_3, \{k_4\}_{k_3} \vdash k_4$.
2. $k_3, k_4 \vdash \langle k_3, k_4 \rangle$.
3. $\langle k_3, k_4 \rangle, \{k_4\}_{k_3} \vdash k_2$.
4. $k_2, \left\{ \{k_1\}_{k_1} \right\}_{k_2} \vdash \{k_1\}_{k_1}$.

Dans cet exemple, on peut voir que l'intrus a ainsi accès aux clefs k_2, k_3 et k_4 par déduction mais ne peut avoir accès à k_1 et donc, par conséquent, ne peut avoir accès à s. (Montrer que k_1 n'est pas déductible demande un peu plus de travail, néanmoins, on peut facilement s'en convaincre dans cet exemple.)

Chapitre 3

Modélisation en Pi-Calcul Appliqué

Le but de ce chapitre est de présenter le pi-calcul appliqué, une algèbre de processus permettant de modéliser les protocoles, ainsi que, en l'utilisant, de proposer une modélisation du protocole de vote Norvégien qui servira de base pour la démonstration de la confidentialité. La description du pi-calcul appliqué ainsi que la formalisation des propriétés est tirée de [DKR09] et [CS11].

3.1 Présentation

Le pi-calcul appliqué est une algèbre de processus introduite par Martin Abadi et Cédric Fournet [AF01]. Cette algèbre est basée sur le pi-calcul [MPW92], une algèbre de processus classique comme CCS [Mil82] ou CSP [Hoa78] qui permettent de modéliser des processus communicants. La principale caractéristique du pi-calcul appliqué réside dans le fait que les messages échangés ne sont plus modélisés par des atomes mais par des termes, ce qui permet de modéliser des messages avec structure, comme ceux utilisés dans les protocoles cryptographiques. Il a d'ailleurs été utilisé pour étudier de nombreux protocoles de sécurité comme les protocoles d'authentification privée [AF03] ou les protocoles de mise en place de clefs [ABF04].

3.1.1 Syntaxe du Pi-Calcul Appliqué

On suppose l'existence d'un ensemble infini de *noms* $a, b, c, \ldots, k, \ldots, m, n, \ldots$, d'un ensemble infini de *variables* $x, y, z, \ldots$ et d'une *signature* Σ qui est un ensemble fini de *symboles de fonction*, chacun associé à une arité, qui serviront à définir les termes. On utilise les métavariables u, w pour désigner à la fois des noms et des variables. Les *termes* L, M, N, T, U, V sont construits en appliquant les symboles de fonctions à des noms, des variables et d'autres termes. On écrit $\{^M/_x\}$ pour désigner la *substitution* qui remplace la variable x par le terme M. De multiples substitutions successives peuvent être écrites de cette manière : $\{^{M_1}/_{x_1}, ^{M_2}/_{x_2}, \ldots, ^{M_k}/_{x_k}\}$ et les lettres grecques σ, τ désignent généralement des substitutions. On écrira $N\sigma$ pour le résultat de l'application de la substitution σ au terme N. On rappelle qu'un terme est *clos* s'il ne contient aucune variable. La signature Σ est dotée d'une *théorie équationelle* E. On définit une égalité modulo la théorie équationelle, dénotée $=_E$, comme la plus petite relation d'équivalence

sur les termes qui contient E et qui est close par application de symboles de fonction, de substitutions de termes par des variables et par renommage bijectif de noms.

Les *processus* et *processus étendus* sont définis dans la figure 3.1 où M, N sont des termes, n un nom, x une variable et u est une métavariable.

$P, Q, R :=$	processus
0	processus nul
$P \mid Q$	Composition parallèle
$!P$	Réplication
$\nu n.P$	Restriction d'un nom n dans P
If $M = N$ Then P Else Q	Conditionnel
$u(x).P$	Réception de message sur le channel u
$\overline{u}\langle N \rangle.P$	Emission d'un message sur le channel u

$A, B, C :=$	processus étendus
P	processus
$A \mid B$	Composition parallèle
$\nu n.A$	Restriction d'un nom n
$\nu x.A$	Restriction d'une variable x
$\{^M/_x\}$	Substitution active

FIGURE 3.1 – Syntaxe du Pi-Calcul Appliqué.

On écrira $\nu\tilde{n}$ pour désigner la série (pouvant être vide) de liens distincts deux à deux $\nu n_1.\cdots.\nu n_k$. Les *substitutions actives* $\{^M/_x\}$ sont présentes dans les processus étendus pour représenter la mémoire locale d'un processus. Les substitutions actives généralisent la construction « let » : $\nu x.(\{^M/_x\}|P)$ correspond exactement à « let $x = M$ in P ». De multiples substitutions actives successives peuvent être obtenues par composition parallèle et on pourra simplifier $\{^{M_1}/_{x_1}\}|\ldots|\{^{M_k}/_{x_k}\}$ en $\{^{M_1}/_{x_1},\ldots,^{M_k}/_{x_k}\}$ ou $\{^{\tilde{M}}/\tilde{x}\}$. Les lettres grecques σ, τ peuvent également faire référence à des substitutions actives et $N\sigma$ est le résultat de l'application de σ sur le terme N. Les processus étendus ne peuvent avoir, au maximum, qu'une substitution active par variable et il y en a exactement une lorsque la variable est soumise à une restriction (νx).

Les noms et les variables ont des *portées*, délimitées par les restrictions et les réceptions de message $(u(x))$. L'ensemble des noms restreints est noté $bn(A)$ et l'ensemble des variables restreintes est noté $bv(A)$. De manière similaires, on définit l'ensemble des noms libres, $fn(A)$ et des variables libres $fv(A)$. On pourra écrire $fn(N)$ (et respectivement $fv(N)$) pour l'ensemble des noms libres (respectivement des variables libres) qui apparaissent dans le terme N. Un processus étendu est *clos* lorsque chacune de ses variables x est soit restreinte, soit définie par une substitution active.

Un *contexte* $C[\_]$ est un processus étendu avec un « trou » (noté $\_$) à la place d'un processus étendu. Un *contexte d'évaluation* $C[\_]$ est un contexte, tel que le trou n'est pas sous une réplication, ni une conditionnelle, ni un input ou un output (émission/réception). Un contexte d'évaluation est donc de la forme « $C'|(\nu n_1, \ldots, n_k).\_$ ». Le remplacement du trou par un processus étendu P est noté $C[P]$.

Une *frame*, noté φ ou ψ, est un processus étendu constitué du processus nul (0) et de substitutions actives $\{^M/_x\}$ composées en parallèle et soumises à d'éventuelles restrictions. Le *domaine* $dom(\varphi)$ d'une frame φ est l'ensemble des variables x pour lesquelles φ contient une substitution active $\{^M/_x\}$ telle que x ne soit pas soumises à une restriction. Tout processus étendu A peut être décrit par sa frame $\varphi(A)$ en remplaçant tout processus par le processus nul dans A.

3.1.2 Sémantique du Pi-Calcul Appliqué

PAR-0	$A\|0$	$\equiv$	A
PAR-A	$(A\|B)\|C$	$\equiv$	$A\|(B\|C)$
PAR-C	$A\|B$	$\equiv$	$B\|A$
REPL	P	$\equiv$	$P\|!P$
NEW-0	$\nu n.0$	$\equiv$	0
NEW-A	$\nu u.\nu w.A$	$\equiv$	$\nu w.\nu u.A$
NEW-PAR	$\nu u.(A\|B)$	$\equiv$	$A\|\nu u.B$ Si $u \notin fn(A) \cup fv(A)$
ALIAS	$\nu x.\{^M/_x\}$	$\equiv$	0
SUBST	$\{^M/_x\}\|A$	$\equiv$	$\{^M/_x\}\|A\{^M/_x\}$
REWRITE	$\{^M/_x\}$	$\equiv$	$\{^N/_x\}$ Si $M =_E N$
COMM	$\overline{c}\langle x\rangle.P\|c(x).Q$	$\longrightarrow$	$P\|Q$
THEN	If ϕ Then P Else Q	$\longrightarrow$	P Si ϕ est vraie.
ELSE	If ϕ Then P Else Q	$\longrightarrow$	Q Sinon.

IN $\quad c(x).P \xrightarrow{c(M)} P\{^M/_x\}$

OUT-ATOM $\quad \overline{c}\langle u\rangle.P \xrightarrow{\overline{c}\langle u\rangle} P$

OPEN-ATOM $\quad \dfrac{A \xrightarrow{\overline{c}\langle u\rangle} A' \quad u \neq c}{\nu u.A \xrightarrow{\nu u.\overline{c}\langle u\rangle} A'}$

SCOPE $\quad \dfrac{A \xrightarrow{\alpha} A' \quad u \text{ n'apparait pas dans } \alpha}{\nu u.A \xrightarrow{\alpha} \nu u.A'}$

PAR $\quad \dfrac{A \xrightarrow{\alpha} A' \quad bn(\alpha) \cap fn(B) = bv(\alpha) \cap fv(B) = \emptyset}{A|B \xrightarrow{\alpha} A'|B}$

STRUCT $\quad \dfrac{A \equiv B \quad B \xrightarrow{\alpha} B' \quad B' \equiv A'}{A \xrightarrow{\alpha} A'}$

FIGURE 3.2 – Sémantique du Pi-Calcul Appliqué.

La sémantique du pi-calcul appliqué est définie par trois relations : l'*équivalence structurelle* ($\equiv$), la *réduction interne* ($\longrightarrow$) et la *réduction étiquetée* ($\xrightarrow{\alpha}$). Ces relations satisfont aux règles décrites dans la figure 3.2 et sont définies de telle sorte que : l'équivalence structurelle est la plus petite relation d'équivalence sur les processus étendus, close par α-conversion de noms restreints et de variables restreintes et par application de contexte d'évaluation ; la réduction interne est la plus petite relation sur les processus étendus close par équivalence structurelle et application de contexte d'évaluation ; et, pour la réduction étiquetée, α est une *étiquette* de la forme $c(M)$, $\overline{c}\langle u\rangle$ or $\nu u.\overline{c}\langle u\rangle$.

3.1.3 Equivalences

Deux notions d'équivalence vont être utiles pour modéliser la propriété de confidentialité des protocoles de vote.

Définition 3.1. (Equivalence Statique)
Deux frames closes φ et ψ sont *statiquement équivalentes*, noté $\varphi \approx_s \psi$, si $dom(\varphi) = dom(\psi)$ et s'il existe un ensemble de noms $\tilde{n}$, des substitutions σ et τ, tels que $\varphi \equiv \nu\tilde{n}.\sigma$ et $\psi \equiv \nu\tilde{n}.\tau$ et que, pour tous termes M, N tels que $\tilde{n} \cap (fn(M) \cup fn(N)) = \emptyset$, l'on ait $M\sigma =_E N\sigma$ si et seulement si l'on a $M\tau =_E N\tau$.

Deux processus étendus A, B sont statiquement équivalent, noté $A \approx_s B$, si leurs frames sont statiquement équivalentes, c'est-à-dire $\varphi(A) \approx_s \varphi(B)$.

Remarque. La relation $\approx_s$ est appelée équivalence *statique* car elle n'examine que l'état courant des processus et non le comportement dynamique de ceux-ci. La définition suivante capture la partie dynamique.

Définition 3.2. (Bisimilarité étiquetée)
La *bisimilarité étiquetée*, notée $\approx_l$, est la plus grande relation symétrique $\mathcal{R}$ sur les processus étendus clos telle que $A\mathcal{R}B$ implique :

1. $A \approx_s B$;
2. Si $A \longrightarrow A'$ alors $B \longrightarrow^* B'$ et $A'\mathcal{R}B'$ pour B' quelconque ;
3. Si $A \xrightarrow{\alpha} A'$ de telle sorte que $fv(\alpha) \subseteq dom(A)$ et $bn(\alpha) \cap fn(B) = \emptyset$, alors $B \longrightarrow^* \xrightarrow{\alpha} \longrightarrow^* B'$ et $A'\mathcal{R}B'$ pour B' quelconque.

3.2 Formalisation des protocoles de vote

Avant de formaliser les propriétés de sécurité, il faut définir ce qu'est un protocole de vote électronique en pi-calcul appliqué. Dans ce modèle, on modélise les parties honnêtes comme des processus tandis que les parties corrompues, considérées sous le contrôle de l'attaquant, sont, tout simplement, non modélisées.

Définition 3.3. Un *processus de vote* est un processus clos

$$VP \equiv \nu\tilde{n}.(V\sigma_1|\cdots|V\sigma_n|A_1|\cdots|A_m).$$

Les $V\sigma_i$ représentent les processus des différents votants, avec les σ_i des substitutions telle que $v \in dom(\sigma_i)$ est une variable faisant référence à la valeur d'un vote. (Le votant

$V\sigma_i$ vote $v\sigma_i$.) Les A_j représentent les autorités de l'élection qui doivent être honnêtes. On définit un contexte d'évaluation S identique à VP hormis le fait qu'il possède un trou à la place de deux des $V\sigma_i$.

Nous allons maintenant nous intéresser à la formalisation de la propriété de confidentialité du vote. Cette propriété vise à garantir que le lien entre un votant donné, V, et son vote, v, demeure secret. Elle est néanmoins assez subtile à formaliser. En effet, considérons, par exemple, le cas où tous les votants, sauf un, sont malhonnêtes. Comme les résultats du vote sont publiés à la fin de l'élection, tous les votants malhonnêtes peuvent très bien comploter ensemble et ainsi découvrir le vote de l'électeur honnête. Une astuce classique pour modéliser l'anonymat serait de demander si deux processus, l'un représentant le votant V_A, l'autre le votant V_B, sont équivalents. Toutefois, une telle équivalence ne tient pas, étant donné que les identités des votants sont révélées (et elles doivent être révélées, au moins pour que l'administrateur puisse vérifier l'éligibilité du votant). De manière similaire, l'équivalence de deux processus où seul le vote change ne peut être vérifiée, puisque les votes sont révélés à la fin du protocole. Pour garantir l'anonymat, il faut cacher le lien entre le votant et son vote et non juste le votant ou juste le vote.

Dans le but de donner une définition raisonnable pour l'anonymat, on suppose qu'au moins deux votants sont honnêtes. Soient V_A et V_B ces votants. On dit qu'un protocole de vote respecte la confidentialité si le processus où V_A vote a et V_B vote b est *observationnellement équivalent* au processus où V_A vote b et V_B vote a. Formellement, cela nous conduit à la définition suivante :

Définition 3.4. Un protocole de vote vérifie l'anonymat du vote si, pour tous votes a et b :

$$S[V_A\{^a/_v\}|V_B\{^b/_v\}] \approx_l S[V_A\{^b/_v\}|V_B\{^a/_v\}].$$

3.3 Modélisation du Protocole Norvégien

En utilisant le pi-calcul appliqué et la description faite du protocole dans [Gjo10], nous allons modéliser le protocole de vote norvégien en commençant par définir sa théorie équationelle puis les divers processus qui le compose.

3.3.1 Théorie Équationelle

Pour modéliser le protocole, on va d'abord définir les fonctions qui le composent en créant sa théorie équationelle.

Définition 3.5. Soit Σ une signature, telle que :

$$\Sigma = \{\mathsf{OK}, \mathsf{fst}, \mathsf{hash}, \mathsf{pk}, \mathsf{s}, \mathsf{snd}, \mathsf{vk}, \mathsf{blind}, \mathsf{dec}, +, *, \circ, \diamond$$
$$\mathsf{pair}, \mathsf{renc}, \mathsf{sign}, \mathsf{unblind}, \mathsf{checkpfk}_1, \mathsf{checkpfk}_2, \mathsf{checksign}, \mathsf{penc}, \mathsf{pfk}_1, \mathsf{pfk}_2\}$$

avec OK une fonction constante ; fst, hash, pk, s, snd, vk des fonctions unaires ; blind, dec, $+$, $*$, $\circ$, $\diamond$, pair, renc, sign, $\mathsf{unblind}$ des fonctions binaires ; $\mathsf{checkpfk}_1$, $\mathsf{checkpfk}_2$, $\mathsf{checksign}$, penc des fonctions ternaires et pfk_1, pfk_2 des fonctions d'arité quatre.

La fonction OK est un message type, équivalent à un true. Les fonctions fst et snd (first et second) sont les représentations des projections sur, respectivement, la première et la seconde composante d'une paire dont le symbole est pair. Le symbole hash modélise une fonction de hachage. Le symbole s représente une fonction qui prend en entrée une identité secrète, id, et renvoie $\mathsf{s}(id)$ un nombre relatif à cette identité secrète. Les symboles pk et vk modélisent, respectivement, la partie publique d'une clef secrète ou d'une identité secrète. Le symbole blind fait référence à un masque, sa fonction réciproque est unblind qui permet le démasquage. Le symbole dec est celui de la fonction de déchiffrement associé au chiffrement à clef publique penc. Le symbole renc modélise une fonction de sur-chiffrement. Les symboles $+$, $*$, $\circ$ et $\diamond$ font référence à des fonctions associatives et commutatives qui agissent respectivement sur des clefs secrètes, des nombres et des messages pour les deux dernières. Le symbole sign est la modélisation d'une fonction de signature que l'on peut vérifier à l'aide de la fonction appelée par checksign. Les symboles pfk_1 et pfk_2 modélisent des preuves de connaissance à divulgation nulle, elles sont assorties de $\mathsf{checkpfk}_1$ et $\mathsf{checkpfk}_2$ qui permettent de vérifier ces preuves.

Le but de telles preuves est de démontrer la connaissance d'un secret sans le divulguer. Pour illustrer la notion de preuve de connaissance à divulgation nulle, imaginons qu'Alice dise à Bob qu'elle sait résoudre un sudoku mais que Bob ne la croit pas sans preuve. Alice voudrait lui prouver qu'elle sait effectivement résoudre le sudoku mais sans lui révéler la solution. Pour cela, elle peut utiliser 81 cartes (9 cartes pour chaque chiffre entre 1 et 9) et réaliser le sudoku en l'absence de Bob puis retourner les cartes face cachée. Elle demande alors à Bob de choisir entre une ligne, une colonne ou un carré de 3x3 cases. Elle retourne alors les cartes du choix de Bob, lui permettant ainsi de vérifier qu'effectivement le sudoku est bien fait, puisque, dans ce cas, il ne doit y avoir qu'une occurrence de chaque chiffre sur la ligne, la colonne ou dans le carré. Étant donné qu'Alice ne peut prévoir à l'avance le choix que fera Bob, elle est obligée de connaître la grille entière. Elle prouve ainsi à Bob qu'elle sait faire le sudoku sans lui révéler la solution.

Définition 3.6. Soit E la théorie équationelle qui assume que les fonctions $+$, $*$, $\circ$, $\diamond$ sont associatives et commutatives, et inclut les équations suivantes :

(1) $\mathsf{fst}(\mathsf{pair}(x, y)) = x$

(2) $\mathsf{snd}(\mathsf{pair}(x, y)) = y$

(3) $\mathsf{dec}(\mathsf{penc}(x_{plain}, x_{rand}, \mathsf{pk}(x_{sk})), x_{sk}) = x_{plain}$

(4) $\mathsf{dec}(\mathsf{blind}(\mathsf{penc}(x_{plain}, x_{rand}, \mathsf{pk}(x_{sk})), x_{blind}), x_{sk}) = \mathsf{blind}(x_{plain}, x_{blind})$

(5) $\mathsf{penc}(x_{pl}, x_{rand}, x_{pub}) \circ \mathsf{penc}(y_{pl}, y_{rand}, x_{pub}) = \mathsf{penc}(x_{pl} \diamond y_{pl}, x_{rand} * y_{rand}, x_{pub})$

(6) $\mathsf{renc}(\mathsf{penc}(x_{plain}, x_{rand}, \mathsf{pk}(x_{sk})), y_{sk}) = \mathsf{penc}(x_{plain}, x_{rand}, \mathsf{pk}(x_{sk} + y_{sk}))$

(7) $\mathsf{unblind}(\mathsf{blind}(x_{plain}, x_{blind}), x_{blind}) = x_{plain}$

(8) $\mathsf{checksign}(x_{plain}, \mathsf{vk}(x_{id}), \mathsf{sign}(x_{plain}, x_{id})) = \mathsf{OK}$

(9) $\mathsf{checkpfk}_1(\mathsf{vk}(x_{id}), \mathsf{ball}, \mathsf{pfk}_1(x_{id}, x_{rand}, x_{plain}, \mathsf{ball})) = \mathsf{OK}$
où $\mathsf{ball} = \mathsf{penc}(x_{plain}, x_{rand}, x_{pub})$.

(10) $\mathsf{checkpfk}_2(\mathsf{vk}(x_{id}), \mathsf{ball}, \mathsf{pfk}_2(x_{id}, x_{bk}, x_{plain}, \mathsf{ball})) = \mathsf{OK}$
où $\mathsf{ball} = \mathsf{renc}(x_{plain}, x_{bk})$ or $\mathsf{ball} = \mathsf{blind}(x_{plain}, x_{bk})$.

Dans la théorie équationelle E, les équations (1) et (2) modélisent les projections permettant de récupérer l'une ou l'autre des composantes d'une paire donnée. L'équation (3) modélise le déchiffrement d'un chiffrement à clef publique. Le résultat du déchiffrement est le message initial. L'équation (4) modélise le déchiffrement sous « masque ». En effet, la fonction blind, comme son nom l'indique, cache quelque chose, et, dans notre cas, il est possible de déchiffrer un message chiffré et masqué. Le résultat d'un tel déchiffrement est le message initial masqué. L'équation (5) modélise l'homomorphisme du chiffrement. L'équation (6) modélise un sur-chiffrement réalisé à partir d'un premier chiffré et d'une clef secrète qui permet d'obtenir un autre chiffré dont la clef est exprimé en fonction des deux précédentes. L'équation (7) modélise simplement le démasquage d'un message masqué. Enfin les trois dernières équations modélisent des vérifications de signatures et de preuves de connaissances.

Remarque. Par commodité, on pourra utiliser, dans la suite, les notations suivantes :

$$\begin{aligned}(x_1, \ldots, x_k) &= \mathsf{pair}(x_1, \mathsf{pair}(x_2, \mathsf{pair}(\ldots, \mathsf{pair}(x_{k-1}, x_k))\ldots)) \\ \Pi_i(x) &= \mathsf{fst}(\mathsf{snd}^{i-1}(x))\end{aligned}$$

Pour la suite, on a besoin de manipuler une théorie équationelle convergente. On prouve donc que celle définie ci-dessus l'est.

Lemme 3.7. *Soit $\mathcal{R}$ le système de réécriture issu de E en orientant toutes les équations de la gauche vers la droite. (Sauf les équations AC.) Alors $\mathcal{R}$ est AC-confluent.*

Preuve. Pour montrer la confluence, on regarde les paires critiques. Il n'y en a pas, donc $\mathcal{R}$ est AC-confluent.

■

Lemme 3.8. *$\mathcal{R}$ est AC-terminant.*

Preuve. Montrons maintenant la propriété de terminaison. Introduisons pour cela une mesure de la longueur des différents termes, $|.|$, définie par :

1. $|u| = 1$ si u est une variable ou une constante,
2. $|\mathsf{penc}(M_1, M_2, M_3)| = 2 + |M_1| + |M_2| + |M_3|$,
3. $|\mathsf{renc}(M_1, M_2)| = 2 + |M_1| + |M_2|$,
4. $|h(M_1, \ldots, M_k)| = 1 + \sum_{i=1}^{k} |M_i|$, sinon.

Selon cette définition de longueur, on peut facilement voir que toutes les équations, excepté (5) et (6), réduisent trivialement la longueur des termes. Montrons maintenant que la longueur décroit également, selon la définition, dans les équations (5) et (6) :

- Equation (5) :

$$\begin{aligned}|\mathsf{penc}(M_1, M_2, M_3) \circ \mathsf{penc}(M'_1, M'_2, M_3)| &= 1 + |\mathsf{penc}(M_1, M_2, M_3)| + |\mathsf{penc}(M'_1, M'_2, M_3)| \\ &= 1 + (2 + |M_1| + |M_2| + |M_3|) + (2 + |M'_1| + |M'_2| + |M_3|) \\ &= 5 + |M_1| + |M_2| + |M'_1| + |M'_2| + 2|M_3|.\end{aligned}$$

$$\begin{aligned}|\mathsf{penc}(M_1 \circ M_1', M_2 * M_2', M_3)| &= 2 + |M_1 \circ M_1'| + |M_2 * M_2'| + |M_3| \\ &= 2 + (1 + |M_1| + |M_1'|) + (1 + |M_2| + |M_2'|) + |M_3| \\ &= 4 + |M_1| + |M_2| + |M_1'| + |M_2'| + |M_3|.\end{aligned}$$

Alors $|\mathsf{penc}(M_1 \circ M_1', M_2 * M_2', M_3)| < |\mathsf{penc}(M_1, M_2, M_3) \circ \mathsf{penc}(M_1', M_2', M_3)|$.

- Equation (6) :

$$\begin{aligned}|\mathsf{renc}(\mathsf{penc}(M_1, M_2, \mathsf{pk}(M_3)), M_4)| &= 2 + |\mathsf{penc}(M_1, M_2, \mathsf{pk}(M_3))| + |M_4| \\ &= 2 + (2 + |M_1| + |M_2| + 1 + |M_3|) + |M_4| \\ &= 5 + |M_1| + |M_2| + |M_3| + |M_4|.\end{aligned}$$

$$\begin{aligned}|\mathsf{penc}(M_1, M_2, \mathsf{pk}(M_3 + M_4))| &= 2 + |M_1| + |M_2| + 1 + |M_3 + M_4| \\ &= 3 + |M_1| + |M_2| + (1 + |M_3| + |M_4|) \\ &= 4 + |M_1| + |M_2| + |M_3| + |M_4|.\end{aligned}$$

Alors $|\mathsf{penc}(M_1, M_2, \mathsf{pk}(M_3 + M_4))| < |\mathsf{renc}(\mathsf{penc}(M_1, M_2, \mathsf{pk}(M_3)), M_4)|$.

Toutes les équations de la théorie réduisent la longueur des termes selon cette définition. $\mathcal{R}$ est donc AC-terminant.

■

Proposition 3.9. *$\mathcal{R}$ est un système de réécriture AC-convergent.*

PREUVE. La proposition est une conséquence des deux lemmes précédents qui montrent que $\mathcal{R}$ est AC-confluent et AC-terminant.

■

3.3.2 Modélisation

On peut maintenant passer à la modélisation du protocole.

Remarque. Dans ce modèle, on ne modélise seulement que le dernier vote soumis par les votants au lieu d'explicitement permettre à ceux-ci de revoter. De plus, on ne considère qu'une option par vote. Le votant et son ordinateur ont été également fusionnés en une seule et même entité.

Les Canaux de Transmissions

En plus de modéliser les participants, il faut également modéliser les canaux de transmissions qui les lient. En particulier, il convient de décider si ces canaux sont publics (l'intrus peut écouter et émettre sur ce canal), authentifiés (l'intrus peut écouter mais ne peut pas émettre) ou privés (l'intrus ne peut ni écouter, ni émettre). La figure 3.3 schématise les différents canaux.

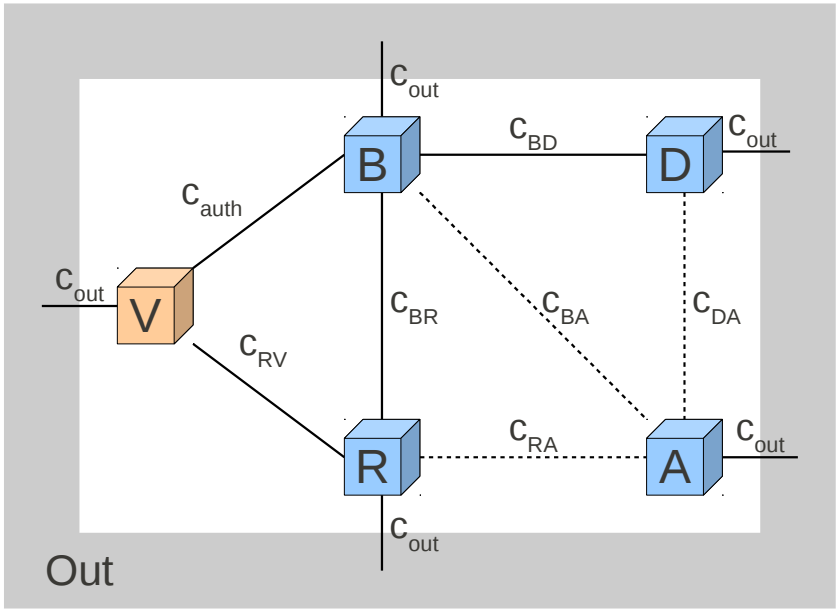


FIGURE 3.3 – Communications entre participants. Les participants en bleu correspondent à l'infrastructure du bureau de vote virtuel.

Les canaux c_{auth} et c_{RV} sont des canaux authentifiés (un par votant) et l'attaquant peut donc les écouter mais pas les utiliser pour transmettre des messages dans un sens ou dans l'autre. Dans le modèle, comme on ne peut modéliser que des canaux privés ou publics, on modélisera ces canaux comme privés mais on relaiera les communications de ces canaux sur un canal public c_{out} pour que l'attaquant puisse y avoir accès tout de même. Les canaux c_{BD}, c_{BR}, c_{BA}, c_{RA} et c_{DA}, quant à eux, sont des canaux privés, l'attaquant ne pouvant ni écouter, ni transmettre.

Afin de simplifier l'écriture des différentes modélisations, on introduit ici la définition de plusieurs opérations qui représentent les vérifications effectuées par certains participants pour s'assurer que tout est réalisé correctement et que personne n'essaie de tricher.

Définition 3.10.

$$\begin{aligned}\phi_{\mathsf{b}}(id_i, x) = [&(\Pi_1(x), \Pi_2(x), \Pi_3(x)) = x \\ &\wedge \mathsf{checksign}((\Pi_1(x), \Pi_2(x)), \mathsf{vk}(id_i), \Pi_3(x)) = \mathsf{OK} \\ &\wedge \mathsf{checkpfk}_1(\mathsf{vk}(id_i), \Pi_1(x), \Pi_2(x)) = \mathsf{OK}]\end{aligned}$$

$$(\forall k = 1..3,\ x_i^k = \Pi_k(\Pi_1(x)),\ \forall k = 4..7,\ x_i^k = \Pi_{k-2}(x))$$
$$\phi_{\mathsf{r}}(idp_i, x) = \big[(x_i^1, x_i^2, x_i^3) = \Pi_1(x) \wedge (\Pi_1(x), x_i^4, x_i^5, x_i^6, x_i^7) = x$$
$$\wedge\ \mathsf{checksign}((x_i^1, x_i^2), idp_i, x_i^3) = \mathsf{OK} \wedge \mathsf{checkpfk}_1(idp_i, x_i^1, x_i^2) = \mathsf{OK}$$
$$\wedge\ \mathsf{checkpfk}_2(idp_i, x_i^4, x_i^5) = \mathsf{OK} \wedge \mathsf{checkpfk}_2(idp_i, x_i^6, x_i^7) = \mathsf{OK}\big]$$

$$\phi_{\mathsf{s}}^{idp_R}(x, y) = [\mathsf{checksign}(x, idp_R, y) = \mathsf{OK}]$$

$$\phi_{\mathsf{v}}^{idp_R}(id_i, x, y, v, z) = [\mathsf{checksign}(x, idp_R, y) = OK\ \wedge\ v = \mathsf{unblind}(z, \mathsf{s}(id_i))]$$

Le Protocole du Votant

Le rôle du Votant (et de son ordinateur) consiste à soumettre un vote à l'urne. Dans ce but, celui-ci réalise donc un chiffré de son vote à l'aide de la clef publique g_1 ainsi qu'une preuve à divulgation nulle de connaissance, pour ainsi pouvoir prouver que le chiffré contient bien le bon vote. Il réalise également une signature et soumet le tout à l'urne au travers du canal c_{auth}. Suite à cela, il attends la réception de la confirmation de l'urne et du reçu obtenu via le générateur de reçus puis effectue une vérification pour s'assurer que la confirmation et le reçu correspondent bien à son vote initial. Si ce n'est pas le cas, le votant précise qu'il y a eu un problème.

La modélisation est la suivante (les constantes placées en paramètres sont la connaissance initiale du votant) :

$$V(c_{auth}, c_{out}, c_{RV}, g_1, id, idp_R, x_{vote}) = \nu t\ .$$
$$\mathsf{Let}\ e = \mathsf{penc}(x_{vote}, t, g_1)\ \mathsf{in}$$
$$\mathsf{Let}\ p = \mathsf{pfk}_1(id, t, x_{vote}, e)\ \mathsf{in}$$
$$\mathsf{Let}\ si = \mathsf{sign}((e, p), id)\ \mathsf{in}$$
$$\overline{c}_{auth}\langle(e, p, si)\rangle\ .$$
$$c_{auth}(x)\ .\ c_{RV}(y)\ .$$
$$\mathsf{Let}\ h = \mathsf{hash}((\mathsf{vk}(id), e, p, si))\ \mathsf{in}$$
$$\mathsf{If}\ \phi_{\mathsf{v}}^{idp_R}(id, h, x, x_{vote}, y)\ \mathsf{Then}\ \overline{c}_{out}\langle\mathsf{OK}\rangle$$
$$\mathsf{Else}\ \overline{c}_{out}\langle\mathsf{fail}\rangle$$

Le Protocole de l'Urne

Pour chaque vote reçu, l'urne va effectuer une vérification s'assurant que la signature, la preuve et le chiffré sont bien les bons. (Dans la modélisation, l'urne renvoie également le vote sur le canal public pour modéliser la lecture du canal authentifié par l'intrus.) Une fois la vérification validée, l'urne réalise un sur-chiffrement du vote chiffré puis masque le résultat en utilisant une donnée spécifique au votant qui vient de soumettre son vote. Elle génère également des preuves comme quoi ses calculs sont corrects (les preuves de connaissances) puis envoie le tout au générateur de reçu. Si

tout se passe correctement, elle reçoit une confirmation du générateur de reçu qu'elle vérifie avant de la renvoyer au votant (et sur le canal public). L'urne effectue ce travail pour les n votants puis passe à la phase de décompte. Durant cette phase, elle envoie au déchiffreur l'ensemble des votes chiffrés et son contenu (votes chiffrés, preuves et signatures) à l'auditeur.

La modélisation est la suivante :

$$
\begin{aligned}
B_n(c_1,\ldots,c_n,&c_{out},c_{BR},c_{BD},c_{BA},g_1,a_2,g_3,id_1,\ldots,id_n,idp_R) = \\
&c_1(x_1)\ .\overline{c}_{out}\langle x_1\rangle \\
&\mathsf{If}\ \phi_{\mathsf{b}}(id_1,x_1)\ \mathsf{Then} \\
&\mathsf{Let}\ s_1 = \mathsf{s}(id_1)\ \mathsf{in} \\
&\mathsf{Let}\ e_1 = \mathsf{renc}(\Pi_1(x_1),a_2)\ \mathsf{in} \\
&\mathsf{Let}\ pfk_1^e = \mathsf{pfk}_2(id_1,a_2,\Pi_1(x_1),e_1)\ \mathsf{in} \\
&\mathsf{Let}\ b_1 = \mathsf{blind}(e_1,s_1)\ \mathsf{in} \\
&\mathsf{Let}\ pfk_1^b = \mathsf{pfk}_2(id_1,s_1,e_1,b_1)\ \mathsf{in} \\
&\overline{c}_{BR}\langle(x_1,e_1,pfk_1^e,b_1,pfk_1^b)\rangle\ .\ c_{BR}(y_1)\ . \\
&\mathsf{Let}\ h_1 = \mathsf{hash}((\mathsf{vk}(id_1),\Pi_1(x_1),\Pi_2(x_1),\Pi_3(x_1)))\ \mathsf{in} \\
&\mathsf{If}\ \phi_{\mathsf{s}}^{idp_R}(h_1,y_1)\ \mathsf{Then} \\
&\overline{c}_{out}\langle y_1\rangle\ .\ \overline{c}_1\langle y_1\rangle\ .\ \overline{c}_{BR}\langle\mathsf{OK}\rangle\ .\ c_{BR}(sy_1)\ . \\
&\ldots \\
&c_n(x_n)\ .\overline{c}_{out}\langle x_n\rangle \\
&\mathsf{If}\ \phi_{\mathsf{b}}(id_n,x_n)\ \mathsf{Then} \\
&\mathsf{Let}\ s_n = \mathsf{s}(id_n)\ \mathsf{in} \\
&\mathsf{Let}\ e_n = \mathsf{renc}(\Pi_1(x_n),a_2)\ \mathsf{in} \\
&\mathsf{Let}\ pfk_n^e = \mathsf{pfk}_2(id_n,a_2,\Pi_1(x_n),e_n)\ \mathsf{in} \\
&\mathsf{Let}\ b_n = \mathsf{blind}(e_n,s_n)\ \mathsf{in} \\
&\mathsf{Let}\ pfk_n^b = \mathsf{pfk}_2(id_n,s_n,e_n,b_n)\ \mathsf{in} \\
&\overline{c}_{BR}\langle(x_n,e_n,pfk_n^e,b_n,pfk_n^b)\rangle\ .\ c_{BR}(y_n)\ . \\
&\mathsf{Let}\ h_n = \mathsf{hash}((\mathsf{vk}(id_n),\Pi_1(x_n),\Pi_2(x_n),\Pi_3(x_n)))\ \mathsf{in} \\
&\mathsf{If}\ \phi_{\mathsf{s}}^{idp_R}(h_n,y_n)\ \mathsf{Then} \\
&\overline{c}_{out}\langle y_n\rangle\ .\ \overline{c}_n\langle y_n\rangle\ .\ \overline{c}_{BR}\langle\mathsf{OK}\rangle\ .\ c_{BR}(sy_n)\ . \\
&\overline{c}_{BD}\langle\Pi_1(x_1)\rangle\ .\ \ldots\ .\ \overline{c}_{BD}\langle\Pi_1(x_n)\rangle\ . \\
&\overline{c}_{BA}\langle x_1\rangle\ .\ \ldots\ .\ \overline{c}_{BA}\langle x_n\rangle
\end{aligned}
$$

Le Protocole du Générateur de Reçus

A chaque sollicitation de l'urne, le générateur de reçus vérifie d'abord l'ensemble des informations reçues : il revérifie le chiffré, la preuve et la signature initiaux puis vérifie ce qu'a créé l'urne avant d'effectuer un hachage du vote, avec ou sans signature. Il crée

le reçu en utilisant ce que lui a envoyé l'urne et l'envoie au votant à travers un canal direct et authentifié. Il signe également le hachage du vote complet (avec signature) et l'envoie à l'urne à travers un canal privé. Il stocke ensuite les hachages et recommence pour les n votants. En phase de décompte, il envoie les hachages stockés à l'auditeur.

La modélisation est la suivante :

$$
\begin{aligned}
R_n(g_1, g_2, a_3, &idp_1, \ldots, idp_n, id_R, c_{RV_1}, \ldots, c_{RV_n}, c_{BR}, c_{RA}, c_{out}) = \\
& c_{BR}(x_1) \ . \\
& \mathsf{Let}\ x_1^k = \Pi_k(\Pi_1(x_1)),\ k = 1..3\ \mathsf{in} \\
& \mathsf{Let}\ x_1^k = \Pi_{k-2}(x_1),\ k = 4...7\ \mathsf{in} \\
& \mathsf{If}\ \phi_{\mathsf{r}}(idp_1, x_1)\ \mathsf{Then} \\
& \mathsf{Let}\ hb_1 = \mathsf{hash}((idp_1, x_1^1, x_1^2, x_1^3))\ \mathsf{in} \\
& \mathsf{Let}\ hbp_1 = \mathsf{hash}((idp_1, x_1^1, x_1^2))\ \mathsf{in} \\
& \mathsf{Let}\ r_1 = \mathsf{dec}(x_1^6, a_3)\ \mathsf{in} \\
& \mathsf{Let}\ si_1 = \mathsf{sign}(hb_1, id_R)\ \mathsf{in} \\
& \overline{c}_{BR}\langle si_1 \rangle \ . \ c_{BR}(sy_1) \ . \ \overline{c}_{out}\langle r_1 \rangle \ . \ \overline{c}_{RV_1}\langle r_1 \rangle \ . \ \overline{c}_{BR}\langle \mathsf{OK} \rangle \ . \\
& \ldots \\
& c_{BR}(x_n) \ . \\
& \mathsf{Let}\ x_n^k = \Pi_k(\Pi_1(x_n)),\ k = 1..3\ \mathsf{in} \\
& \mathsf{Let}\ x_n^k = \Pi_{k-2}(x_n),\ k = 4...7\ \mathsf{in} \\
& \mathsf{If}\ \phi_{\mathsf{r}}(idp_n, x_n)\ \mathsf{Then} \\
& \mathsf{Let}\ hb_n = \mathsf{hash}((idp_n, x_n^1, x_n^2, x_n^3))\ \mathsf{in} \\
& \mathsf{Let}\ hbp_n = \mathsf{hash}((idp_n, x_n^1, x_n^2))\ \mathsf{in} \\
& \mathsf{Let}\ r_n = \mathsf{dec}(x_n^6, a_3)\ \mathsf{in} \\
& \mathsf{Let}\ si_n = \mathsf{sign}(hb_n, id_R)\ \mathsf{in} \\
& \overline{c}_{BR}\langle si_n \rangle \ . \ c_{BR}(sy_n) \ . \ \overline{c}_{out}\langle r_n \rangle \ . \ \overline{c}_{RV_n}\langle r_n \rangle \ . \ \overline{c}_{BR}\langle \mathsf{OK} \rangle \ . \\
& \overline{c}_{RA}\langle (idp_1, hbp_1, hb_1) \rangle \ . \ \ldots \ . \ \overline{c}_{RA}\langle (idp_n, hbp_n, hb_n) \rangle \ .
\end{aligned}
$$

Le Protocole du Déchiffreur

Le déchiffreur n'intervient qu'à la phase de décompte. Il récupère les votes envoyés par l'urne, les regroupe et en fait un hachage qu'il envoie à l'auditeur. S'il obtient confirmation de ce dernier, le déchiffreur poursuit sa tâche en déchiffrant l'ensemble des votes chiffrés puis les émet sur le canal public (publication des résultats) en prenant soin de les mélanger dans un premier temps. Le pi-calcul appliqué ne permettant pas de modéliser directement ce mélange, on modélise deux versions du déchiffreur, l'un ou les votes ne sont pas mélangés, noté $D_n^{12\ldots n}$, et l'un ou les deux premiers votes sont permutés, noté $D_n^{21\ldots n}$.

Les modélisations sont les suivantes :

$$
\begin{aligned}
D_n^{12\ldots n}(a_1, & c_{BD}, c_{DA}, c_{out}) = \\
& c_{BD}(x_1) \ . \ \ldots \ . \ c_{BD}(x_n) \ . \\
& \overline{c}_{DA}\langle \mathsf{hash}((x_1, \ldots, x_n))\rangle \ . \ c_{DA}(x) \ . \\
& \mathsf{If}\ [x = \mathsf{Proceed}]\ \mathsf{Then} \\
& \mathsf{Let}\ dec_k = \mathsf{dec}(x_k, a_1),\ k = 1..n\ \mathsf{in} \\
& \overline{c}_{DA}\langle dec_1\rangle \ . \ \ldots \ . \ \overline{c}_{DA}\langle dec_n\rangle \ . \\
& \overline{c}_{out}\langle dec_1\rangle \ . \ \ldots \ . \ \overline{c}_{out}\langle dec_n\rangle
\end{aligned}
$$

$$
\begin{aligned}
D_n^{213\ldots n}(a_1, & c_{BD}, c_{DA}, c_{out}) = \\
& c_{BD}(x_1) \ . \ \ldots \ . \ c_{BD}(x_n) \ . \\
& \overline{c}_{DA}\langle \mathsf{hash}((x_1, \ldots, x_n))\rangle \ . \ c_{DA}(x) \ . \\
& \mathsf{If}\ [x = \mathsf{Proceed}]\ \mathsf{Then} \\
& \mathsf{Let}\ dec_1 = \mathsf{dec}(x_2, a_1)\ \mathsf{in} \\
& \mathsf{Let}\ dec_2 = \mathsf{dec}(x_1, a_1)\ \mathsf{in} \\
& \mathsf{Let}\ dec_k = \mathsf{dec}(x_k, a_1),\ k = 3..n\ \mathsf{in} \\
& \overline{c}_{DA}\langle dec_1\rangle \ . \ \ldots \ . \ \overline{c}_{DA}\langle dec_n\rangle \ . \\
& \overline{c}_{out}\langle dec_1\rangle \ . \ \ldots \ . \ \overline{c}_{out}\langle dec_n\rangle
\end{aligned}
$$

Remarque. On ne permute que les deux premiers votes car ils sont ceux des votants honnêtes considérés dans la formalisation de la propriété de confidentialité, Permuter seulement ces deux votes doit suffire à pouvoir montrer la propriété.

Le Protocole de l'Auditeur

L'auditeur est là pour vérifier que l'élection s'est bien déroulée. Il n'intervient que dans la phase de décompte. Il reçoit toutes les informations provenant de l'urne et du générateur de reçus. Il compare les deux pour vérifier que rien ne manque et récupère ensuite les informations fournies par le déchiffreur. Il les compare aux informations déjà reçues en calculant lui-même le hachage qu'il devrait recevoir du déchiffreur et compare celui calculé et celui effectivement reçu. S'il y a concordance, il permet au déchiffreur de poursuivre sa tache, sinon l'élection échoue.

La modélisation est la suivante :

$AD_n(c_{BA}, c_{RA}, c_{DA}, c_{out}) =$

$c_{BA}(x_1)\ .\ \ldots\ .\ c_{BA}(x_n)\ .\ c_{RA}(h_1)\ .\ \ldots\ .\ c_{RA}(h_n)\ .$

Let $hb_i = \mathsf{hash}((\Pi_1(x_i), \Pi_2(x_i), \Pi_3(x_i)))$ in

Let $hbp_i = \mathsf{hash}((\Pi_1(x_i), \Pi_2(x_i)))$ in

If $[(\Pi_1(x_i), \Pi_2(x_i), \Pi_3(x_i)) \neq x_i \vee (\Pi_1(h_i), \Pi_2(h_i), \Pi_3(h_i)) \neq h_i$

$\vee\ \mathsf{checksign}((\Pi_1(x_i), \Pi_2(x_i)), \mathsf{vk}(id_i), \Pi_3(x_i)) \neq \mathsf{OK} \vee \Pi_2(h_i) \neq hbp_i \vee \Pi_3(h_i) \neq hb_i]$

Then $\overline{c}_{out}\langle\mathsf{Failed}\rangle$

Else

Let $h = \mathsf{hash}((\Pi_1(x_1), \ldots, \Pi_n(x_n)))$ in

$c_{DA}(h_d)\ .$

If $[h_d \neq h]$ Then $\overline{c}_{out}\langle\mathsf{Failed}\rangle$

Else $\overline{c}_{DA}(\mathsf{Proceed})$

Le Protocole Global

Le protocole complet peut-être finalement modélisé comme un contexte d'évaluation de la manière suivante :

$$A_n^X\,[\_] = \nu\tilde{n}\ .(\mathsf{Let}\ a_3 = a_1 + a_2\ \mathsf{in}\ ).\,\big[\_|B_n|R_n|D_n^X|AD_n|\Gamma\big]$$

avec $\tilde{n} = (a_1, a_2, id_1, id_2, id_R, c_1, c_2, c_{RV_1}, c_{RV_2}, c_{BR}, c_{BD}, c_{BA}, c_{RA}, c_{DA})$, l'ensemble des noms restreints (canaux privés, clefs privées, identités privées), $X \in \{123\ldots n, 213\ldots n\}$ et enfin $\Gamma = \{^{\mathsf{pk}(a_1)}/_{g_1},\ ^{\mathsf{pk}(a_2)}/_{g_2},\ ^{\mathsf{pk}(a_3)}/_{g_3},\ ^{\mathsf{vk}(id_1)}/_{idp_1}, \ldots, ^{\mathsf{vk}(id_n)}/_{idp_n},\ ^{\mathsf{vk}(id_R)}/_{idp_R}\}$, une substitution active qui correspond à la connaissance initiale de l'intrus. (Clefs publiques, identités publiques des votants, identité publique du générateur de reçus.)

Remarque. a_3 est une clef secrète construite à partir de a_1 et a_2, ce qui explique le petit morceau de protocole supplémentaire indiquant la création de cette clef secrète en tout début de protocole.

Chapitre 4

Attaquant Passif

Un attaquant passif se contente d'observer sans agir. Toutefois, il peut tout de même menacer la confidentialité d'un secret s'il s'avère qu'il est capable de le déduire à partir de l'ensemble des messages qu'il a pu voir. Dans le cadre des protocoles de vote, savoir que la déduction est un problème décidable est un premier pas permettant l'automatisation des vérifications de ces protocoles.

4.1 Théorie Équationelle Simplifiée

Dans cette partie, on introduit une théorie équationelle dérivée de celle du protocole de vote Norvégien, simplifiée pour pouvoir mener la démonstration de bout en bout. Les simplifications ont notamment eu lieu dans la suppression des symboles AC qui rendent les démonstrations de décidabilité réellement complexes. L'équation d'homomorphisme a également disparu.

Définition 4.1. Soit Σ une signature telle que :

$$\Sigma = \{\mathsf{OK}, \mathsf{fst}, \mathsf{hash}, \mathsf{pk}, \mathsf{s}, \mathsf{snd}, \mathsf{vk}, \mathsf{blind}, \mathsf{dec}, \mathsf{f}, \mathsf{extract}_1, \mathsf{extract}_2, \\ \mathsf{pair}, \mathsf{renc}, \mathsf{sign}, \mathsf{unblind}, \mathsf{checkpfk}_1, \mathsf{checkpfk}_2, \mathsf{checksign}, \mathsf{penc}, \mathsf{pfk}_1, \mathsf{pfk}_2\}$$

avec OK une constante ; fst, hash, pk, s, snd, vk des fonctions unaires ; blind, dec, f, $\mathsf{extract}_1$, $\mathsf{extract}_2$, pair, renc, sign, $\mathsf{unblind}$ des fonctions binaires ; $\mathsf{checkpfk}_1$, $\mathsf{checkpfk}_2$, $\mathsf{checksign}$, penc des fonctions ternaires et pfk_1, pfk_2 des fonctions d'arité quatre.

Définition 4.2. Soit E la théorie équationelle suivante :

1. $\mathsf{fst}(\mathsf{pair}(x, y)) = x$
2. $\mathsf{snd}(\mathsf{pair}(x, y)) = y$
3. $\mathsf{extract}_1(\mathsf{f}(x, y), y) = x$
4. $\mathsf{extract}_2(\mathsf{f}(x, y), x) = y$
5. $\mathsf{dec}(\mathsf{penc}(x_{plain}, x_{rand}, \mathsf{pk}(x_{sk})), x_{sk}) = x_{plain}$
6. $\mathsf{dec}(\mathsf{blind}(\mathsf{penc}(x_{plain}, x_{rand}, \mathsf{pk}(x_{sk})), x_{blind}), x_{sk}) = \mathsf{blind}(x_{plain}, x_{blind})$
7. $\mathsf{renc}(\mathsf{penc}(x_{plain}, x_{rand}, \mathsf{pk}(x_{sk})), y_{sk}) = \mathsf{penc}(x_{plain}, x_{rand}, \mathsf{pk}(\mathsf{f}(x_{sk}, y_{sk})))$

8. $\mathsf{unblind}(\mathsf{blind}(x_{plain}, x_{blind}), x_{blind}) = x_{plain}$
9. $\mathsf{checksign}(x_{plain}, \mathsf{vk}(x_{id}), \mathsf{sign}(x_{plain}, x_{id})) = \mathsf{OK}$
10. $\mathsf{checkpfk}_1(\mathsf{vk}(x_{id}), \mathsf{ball}, \mathsf{pfk}_1(x_{id}, x_{rand}, x_{plain}, \mathsf{ball})) = \mathsf{OK}$
    où $\mathsf{ball} = \mathsf{penc}(x_{plain}, x_{rand}, x_{pub})$.
11. $\mathsf{checkpfk}_2(\mathsf{vk}(x_{id}), \mathsf{ball}, \mathsf{pfk}_2(x_{id}, x_{bk}, x_{plain}, \mathsf{ball})) = \mathsf{OK}$
    où $\mathsf{ball} = \mathsf{renc}(x_{plain}, x_{bk})$ or $\mathsf{ball} = \mathsf{blind}(x_{plain}, x_{bk})$.

Dans la théorie équationelle E, une grande partie des équations sont issue de celles décrites dans la théorie équationelle du protocole de vote Norvégien. L'équation (7) modélise également un sur-chiffrement mais simplifié puisque f représente une fonction non-AC. Les équations (3) et (4) permettent de récupérer une clef secrète en connaissant l'autre à partir de la combinaison de ces deux clefs. (La connaissance de A+B et de A, donne B.)

Montrons maintenant que cette théorie est convergente.

Lemme 4.3. *Soit $\mathcal{R}$ le système de réécriture issu de E en orientant toutes les équations de la gauche vers la droite. Alors $\mathcal{R}$ est confluent.*

PREUVE. Pour montrer la confluence, il faut exhiber les paires critiques. Comme il n'y en a aucune, $\mathcal{R}$ est automatiquement confluent.

■

Lemme 4.4. *$\mathcal{R}$ est terminant.*

PREUVE. Soit $|.|$ une mesure pour la longueur des différents termes définie par :

1. $|u| = 1$ si u est une variable ou une constante.
2. $|\mathsf{renc}(M_1, M_2)| = 2 + |M_1| + |M_2|$,
3. $|h(M_1, \ldots, M_k)| = 1 + \sum_{i=1}^{k} |M_i|$, sinon.

Conformément à cette définition, on peut facilement voir que toutes les équations sauf (6) réduisent la longueur des termes. Montrons maintenant que la longueur décroit également avec l'équation (6). On a :

- Équation (6) :

$$\begin{aligned} |\mathsf{renc}(\mathsf{penc}(M_1, M_2, \mathsf{pk}(M_3)), M_4)| &= 2 + |\mathsf{penc}(M_1, M_2, \mathsf{pk}(M_3))| + |M_4| \\ &= 2 + (1 + |M_1| + |M_2| + 1 + |M_3|) + |M_4| \\ &= 4 + |M_1| + |M_2| + |M_3| + |M_4|. \end{aligned}$$

$$\begin{aligned} |\mathsf{penc}(M_1, M_2, \mathsf{pk}(\mathsf{f}(M_3, M_4)))| &= 1 + |M_1| + |M_2| + 1 + |\mathsf{f}(M_3, M_4)| \\ &= 2 + |M_1| + |M_2| + 1 + |M_3| + |M_4| \\ &= 3 + |M_1| + |M_2| + |M_3| + |M_4|. \end{aligned}$$

$$\text{Donc } |\mathsf{penc}(M_1, M_2, \mathsf{pk}(\mathsf{f}(M_3, M_4)))| < |\mathsf{renc}(\mathsf{penc}(M_1, M_2, \mathsf{pk}(M_3)), M_4)|.$$

On peut donc voir que toutes les équations réduisent la longueur des termes. On ne peut donc avoir une suite infinie de réductions. (Sans quoi l'on obtiendrait une suite d'entiers décroissante, non-convergente et minorée par 0.) Alors $\mathcal{R}$ est terminant.

■

Proposition 4.5. *$\mathcal{R}$ est un système de réécriture convergent.*

PREUVE. Cette proposition est la conséquence des deux précédents lemmes qui démontrent que $\mathcal{R}$ est confluent et terminant.

■

4.2 Décidabilité de la Déduction

Pour montrer la décidabilité de la déduction pour cette théorie équationelle, nous suivrons le principe de démonstration évoqué dans [BBC09] qui consiste à définir une notion de sous-termes appropriée pour la théorie afin de pouvoir prouver une propriété de déduction locale qui permet ensuite de démontrer la décidabilité de la déduction. Les démonstrations de cette partie sont placées en annexe A.

Définition 4.6. Soit $M_1, \ldots, M_k \in \mathcal{T}(\Sigma, \mathcal{X})$. La notion de sous-termes appropriée pour E, simplement notée St_E, est définie de la manière suivante :

- $St_E(u) = u$ si u est une variable ou un nom,
- $St_E(\mathsf{blind}(\mathsf{penc}(M_1, M_2, M_3), M_4)) = \{\mathsf{blind}(\mathsf{penc}(M_1, M_2, M_3), M_4)\}$
  $\cup\ \{\mathsf{blind}(M_1, M_4)\} \cup St_E(\mathsf{penc}(M_1, M_2, M_3)) \cup St_E(M_4)$
- $St_E(\mathsf{penc}(M_1, M_2, \mathsf{pk}(\mathsf{f}(M_3, M_4)))) = \{\mathsf{penc}(M_1, M_2, \mathsf{pk}(\mathsf{f}(M_3, M_4)))\} \cup St_E(M_1)$
  $\cup \{\mathsf{penc}(M_1, M_2, \mathsf{pk}(M)) \mid M \in St_E(\mathsf{f}(M_3, M_4))\} \cup St_E(M_2) \cup St_E(\mathsf{pk}(\mathsf{f}(M_3, M_4)))$
- $St_E(h(M_1, \ldots, M_k)) = \{h(M_1, \ldots, M_k)\} \cup \bigcup_{i=1}^{k} St_E(M_i)$, sinon.

On montre rapidement que cette définition de sous-termes est propre, c'est-à-dire que l'on a $St_E(St_E(M)) = St_E(M)$ pour tout terme M, grâce au Lemme 4.7.

Lemme 4.7.

$$St_E(St_E(\mathsf{penc}(M_1, M_2, \mathsf{pk}(M_3)))) = St_E(\mathsf{penc}(M_1, M_2, \mathsf{pk}(M_3)))$$
$$St_E(St_E(\mathsf{blind}(\mathsf{penc}(M_1, M_2, M_3), M_4))) = St_E(\mathsf{blind}(\mathsf{penc}(M_1, M_2, M_3), M_4))$$

Définition 4.8. S'il existe une règle $l \longrightarrow r$ du système de réécriture $\mathcal{R}$ et une substitution θ tels qu'il existe U et V des termes tels que $U = l\theta$ et $V = r\theta$, alors on dit que la réduction à lieu *en tête* et l'on note $U \xrightarrow{h} V$.

Le lemme suivant est inspiré de [BBC09].

Lemme 4.9. *(Localité) Soit $\phi = \nu\tilde{n}.\sigma$ une frame en forme normale, M un terme clos en forme normale. Si $\phi \vdash_E M$ alors il existe un terme ζ_M, appelé recette locale, tel que :*

- *$fn(\zeta_M) \cap \tilde{n} = \emptyset$ et $\zeta_M\sigma =_E M$.*
- *$\forall \zeta' \in St_E(\zeta_M),\ \forall \zeta'' \in St_E(\zeta')$ on a $\zeta''\sigma\downarrow \in St_E(\phi, \zeta'\sigma\downarrow) \cup \{\Sigma_0\}$.*
  De plus, Si $\zeta'' = F(\zeta_1, \ldots, \zeta_k)$ et $F(\zeta_1\sigma\downarrow, \ldots, \zeta_k\sigma\downarrow) \xrightarrow{h} \zeta''\sigma\downarrow$ en appliquant une règle sous-terme alors, on a $\zeta''\sigma\downarrow \in St_E(\phi) \cup \{\Sigma_0\}$.

Remarque. Une règle $l \to r$ est dite *règle sous-terme*, si $r \in St_E(l)$ ou r est une constante. Toutes les équations de E, exceptée (6), conformément à la définition de sous-terme défini dans la Définition 4.6, sont des règles sous-termes.

L'algorithme suivant, permettant de décider si $\phi \vdash_E M$ est issu de [BBC09]. L'idée est de générer, par saturation, l'ensemble de tous les sous-termes de ϕ et M qui sont déductibles de ϕ.

Algorithm 1 Algorithme de Déduction

Require: $\phi = \nu\tilde{n}.\{^{M_1}/_{x_1}, \ldots, ^{M_k}/_{x_k}\}, M$
Ensure: true/false
$S := St_E(\phi, M) \cup \{\Sigma_0\} \cup fn(\phi)$
$T := \{(M_i, x_i) | i \in \{1 \ldots k\}\} \cup \{(n, n) | n \in \{\Sigma_0\} \cup fn(\phi)\}$
$T' := \emptyset$
1: **while** $T \neq T'$ **do**
2: $T' := T$
3: **for** all $(t_1, \zeta_1), \ldots, (t_n, \zeta_n) \in T'$ and for every function symbol f **do**
4: **if** $f(t_1, \ldots, t_n) \xrightarrow{h} t$ and $t \in S$ and $t \notin \{t | (t, \zeta_t) \in T\}$ **then**
5: $(t, f(\zeta_1, \ldots, \zeta_n)) \in T$
6: **end if**
7: **if** $t = f(t_1, \ldots, t_n) \in S$ and $t \notin \{t | (t, \zeta_t) \in T\}$ **then**
8: $(t, f(\zeta_1, \ldots, \zeta_n)) \in T$
9: **end if**
10: **end for**
11: **end while**
12: **if** $(M, \zeta_M) \in T$ **then**
13: **return** true
14: **else**
15: **return** false
16: **end if**

Cet algorithme se termine forcément puisque l'on ajoute unique des sous-termes de ϕ et M.

Proposition 4.10. [BBC09] *Soit $\phi = \nu\tilde{n}.\sigma$ une frame telle que $\sigma = \{^{M_1}/_{x_1}, \ldots, ^{M_k}/_{x_k}\}$ est en forme normale, M un terme en forme normale et T l'ensemble généré par l'algorithme 1.*

1. *$\forall M' \in St_E(\phi, M)$, on a $\phi \vdash_E M'$ si et seulement si, il existe $(M', \zeta_{M'}) \in T$.*

2. *De plus, une recette $\zeta_{M'}$ générée par l'algorithme est minimale et locale.*

Corollaire 4.11. [BBC09] *Pour toute frame ϕ en forme normale et pour tout terme clos M en forme normale, $\phi \vdash_E M$ est décidable.*

Chapitre 5

Confidentialité du Vote pour le Protocole Norvégien

Ce chapitre présente le résultat principal établi durant le stage, la confidentialité du vote pour une version simplifiée du protocole de vote. Pour des soucis de visibilités, les démonstrations de ce chapitre sont placés en annexe B.

5.1 Préliminaires

5.1.1 Une Version Simplifiée

Afin d'en simplifier l'étude, l'étude du protocole Norvégien s'est d'abord effectuée sans la présence d'un auditeur. Néanmoins, une telle simplification est sensée, dans la mesure où prouver que le protocole respecte la confidentialité sans auditeur revient à montrer quelque chose de plus fort puisque cela démontre que le protocole garantit la propriété, dans la mesure où toutes les infrastructures de vote (urne, générateur de reçus et déchiffreur) sont honnêtes, sans avoir besoin d'auditer les faits et gestes de chacune des infrastructures. Qui plus est, on pourra se convaincre, mais nous reviendrons sur ce sujet, que l'ajout d'un auditeur ne modifie que peu de chose à la démonstration sans auditeur.

5.1.2 Enoncé du Résultat

On reprend les notations du chapitre 3 concernant la modélisation du protocole en supprimant l'auditeur du protocole global et des autres protocoles. (Les échanges entre l'urne/le générateur de reçus/le déchiffreur et l'auditeur sont supprimés.) En particulier, le protocole complet devient :

$$A_n^X\,[\_] = \nu\tilde{n}\ .(\mathsf{Let}\ a_3 = a_1 + a_2\ \mathsf{in}\ ).\ \big[\_|B_n|R_n|D_n^X|\Gamma\big]$$

avec $\tilde{n} = (a_1, a_2, id_1, id_2, id_R, c_1, c_2, c_{RV_1}, c_{RV_2}, c_{BR}, c_{BD})$, l'ensemble des noms restreints (canaux privés, clefs privées, identités privées), $X \in \{123\ldots n, 213\ldots n\}$ et enfin $\Gamma = \{^{\mathsf{pk}(a_1)}/_{g_1},\ ^{\mathsf{pk}(a_2)}/_{g_2},\ ^{\mathsf{pk}(a_3)}/_{g_3},\ ^{\mathsf{vk}(id_1)}/_{idp_1}, \ldots, ^{\mathsf{vk}(id_n)}/_{idp_n},\ ^{\mathsf{vk}(id_R)}/_{idp_R}\}$, une substitution active qui correspond à la connaissance initiale de l'intrus. (Clefs publiques,

identités publiques des votants, identité publique du générateur de reçus.)

Le but est de montrer la confidentialité de ce protocole de vote, c'est-à-dire :

Définition 5.1. (Confidentialité) Le protocole Norvégien vérifie la *condentialité du vote* si, pour tout v_1,v_2 des valeurs de vote, on a :

$$A_n^{12..n}\left[V\{^{c_1}/_{c_{auth}}, ^{c_{RV_1}}/_{c_{RV}}, ^{id_1}/_{id}\}\sigma | V\{^{c_2}/_{c_{auth}}, ^{c_{RV_2}}/_{c_{RV}}, ^{id_2}/_{id}\}\tau\right] \\ \approx_l A_n^{21..n}\left[V\{^{c_1}/_{c_{auth}}, ^{c_{RV_1}}/_{c_{RV}}, ^{id_1}/_{id}\}\tau | V\{^{c_2}/_{c_{auth}}, ^{c_{RV_2}}/_{c_{RV}}, ^{id_2}/_{id}\}\sigma\right]$$

avec les substitutions $\sigma = \{^{v_1}/_{x_{vote,1}}\}$ et $\tau = \{^{v_2}/_{x_{vote,1}}\}$.

Théorème 5.2. *Le protocole Norvégien satisfait la confidentialité du vote.*

PREUVE. (Idée)

Pour réaliser cette preuve basée sur celle de [CS11], on introduit des évolutions partielles (voir suite) qui permettront de définir une relation $\mathcal{R}$ entre les processus. On montrera que, pour cette relation $\mathcal{R}$ on vérifie les mêmes conditions que la relation de bisimilarité étiquetée, (voir Definition 3.2) et de plus, on a :

$$A_n^{12..n}\left[V\{^{c_1}/_{c_{auth}}, ^{c_{RV_1}}/_{c_{RV}}, ^{id_1}/_{id}\}\sigma | V\{^{c_2}/_{c_{auth}}, ^{c_{RV_2}}/_{c_{RV}}, ^{id_2}/_{id}\}\tau\right] \\ \mathcal{R}\ A_n^{21..n}\left[V\{^{c_1}/_{c_{auth}}, ^{c_{RV_1}}/_{c_{RV}}, ^{id_1}/_{id}\}\tau | V\{^{c_2}/_{c_{auth}}, ^{c_{RV_2}}/_{c_{RV}}, ^{id_2}/_{id}\}\sigma\right].$$

Comme la relation de bisimilarité étiquetée est la relation la plus large vérifiant les propriétés de la définition, si on a $A\mathcal{R}B$ alors on a $A \approx_l B$, d'où le résultat.

■

La suite de ce chapitre détaille les deux grandes étapes de la démonstration du Théorème.

5.2 Notations

Cette section introduit une série de notations utilisées dans la suite. Les premières sont principalement des raccourcis. Les suivantes, détaillées dans des sous-sections mettent en place des éléments plus importants pour la démonstration du théorème.

Définition 5.3. On considère les notations suivantes :

Votants honnêtes $(i = 1, 2)$:

$$\begin{aligned} e_i &= \mathsf{penc}(v_i, t_i, \mathsf{pk}(a_1)) \\ pfk_i &= \mathsf{pfk}_1(id_i, t_i, v_i, e_i) \\ sig_i &= \mathsf{sign}((e_i, pfk_i), id_i) \\ ballot_i &= (e_i, pfk_i, sig_i) \\ hv_i &= \mathsf{hash}((\mathsf{vk}(id_i), e_i, pfk_i, sig_i)) \end{aligned}$$

Urne $(i = 1, n)$:

$$\begin{aligned} e'_i &= \mathsf{renc}(\Pi_1(x_i), \mathsf{s}(id_i)) \\ pfk'_i &= \mathsf{pfk}_2(id_i, a_2, \Pi_1(x_i), e'_i) \\ e''_i &= \mathsf{blind}(e'_i, \mathsf{s}(id_i)) \\ pfk''_i &= \mathsf{pfk}_2(id_i, \mathsf{s}(id_i), e'_i, e''_i) \\ ballot'_i &= (x_i, e'_i, pfk'_i, e''_i, pfk''_i) \\ hbb_i &= \mathsf{hash}((\mathsf{vk}(id_i), x_i)) \end{aligned}$$

Générateur de reçus $(i = 1, n)$:

$$\begin{aligned} r_i &= \mathsf{dec}(\Pi_6(p_i), a_3) \\ hb_i &= \mathsf{hash}((\mathsf{vk}(id_i), \Pi_1(p_i), \Pi_2(p_i), \Pi_3(p_i))) \\ sig_i^R &= \mathsf{sign}(hb_i, id_R) \end{aligned}$$

Déchiffreur $(i = 1, n)$:

$$dec_j = \mathsf{dec}(d_j, a_1)$$

Autre :

$$D^{12\ldots n} = D, \quad D^{213\ldots n} = \overline{D},$$

$$\tilde{n} = a_1, a_2, id_1, id_2, id_R, c_1, c_2, c_{RV_1}, c_{RV_2}, c_{BR}, c_{BD},$$

$$\Gamma = \{{}^{\mathsf{pk}(a_1)}/_{g_1},\ {}^{\mathsf{pk}(a_2)}/_{g_2},\ {}^{\mathsf{pk}(a_1+a_2)}/_{g_3},\ {}^{\mathsf{vk}(id_1)}/_{idp_1}, \ldots,\ {}^{\mathsf{vk}(id_n)}/_{idp_n},\ {}^{\mathsf{vk}(id_R)}/_{idp_R}\}.$$

Votants :

$$\begin{aligned} V_{1,1} &= V\left\{{}^{c_1}/_{c_{auth}},\ {}^{c_{RV_1}}/_{c_{RV}},\ {}^{t_1}/_{t},\ {}^{id_1}/_{id},\ {}^{v_1}/_{v}\right\} \\ V_{1,2} &= V\left\{{}^{c_1}/_{c_{auth}},\ {}^{c_{RV_1}}/_{c_{RV}},\ {}^{t_1}/_{t},\ {}^{id_1}/_{id},\ {}^{v_2}/_{v}\right\} \\ V_{2,1} &= V\left\{{}^{c_2}/_{c_{auth}},\ {}^{c_{RV_2}}/_{c_{RV}},\ {}^{t_2}/_{t},\ {}^{id_2}/_{id},\ {}^{v_1}/_{v}\right\} \\ V_{2,2} &= V\left\{{}^{c_2}/_{c_{auth}},\ {}^{c_{RV_2}}/_{c_{RV}},\ {}^{t_2}/_{t},\ {}^{id_2}/_{id},\ {}^{v_2}/_{v}\right\} \end{aligned}$$

Substitutions :

$$\begin{aligned} \sigma &= \left\{{}^{v_1}/_{x_{vote,1}}\right\} \\ \tau &= \left\{{}^{v_2}/_{x_{vote,1}}\right\} \\ \Sigma_L &= \left\{{}^{v_1}/_{x_{vote,1}},\ {}^{v_2}/_{x_{vote,2}}\right\} \\ \Sigma_R &= \left\{{}^{v_2}/_{x_{vote,1}},\ {}^{v_1}/_{x_{vote,2}}\right\} \end{aligned}$$

Outputs :

$$\begin{aligned} \Lambda_j &= \{\{{}^{N_k}/_{x_k},\ {}^{r_k}/_{y_k},\ {}^{sig_k^R}/_{z_k}\} \mid j > 3 \wedge k \in \{3, \ldots, j-1\}\} \\ \Lambda_j^{'} &= \Lambda_j | \{{}^{N_j}/_{x_j}\} \\ \Lambda_j^{''} &= \Lambda_j^{\prime} | \{{}^{sig_j^R}/_{z_j}\} \\ \Lambda_{n+1} &= \{\{{}^{N_k}/_{x_k},\ {}^{r_k}/_{y_k},\ {}^{sig_k^R}/_{z_k}\} \mid k \in \{3, \ldots, n\}\} \end{aligned}$$

5.2.1 Evolutions partielles

Afin de définir plus facilement la relation $\mathcal{R}$, on introduit des évolutions partielles de chaque protocoles qui consistent en des descriptions étapes par étapes.

Définition 5.4. Evolutions partielles du protocole global, qui représentent l'enrichissement de la frame avec les données connues par l'attaquant au fur et à mesure de l'éxécution du protocole.

$$
\begin{aligned}
A_0 &= \nu\tilde{n}.\,[\_|\Gamma] \\
A_1 &= \nu\tilde{n}, t_1\ .\ \left[\_|\Gamma|\{^{ballot_1}/_{x_1}\}\right] \\
A_2 &= A_1\left[\_|\{^{ballot_1}/_{b_1}\}\right] \\
A_3 &= A_2\left[\_|\{^{sig_1^R}/_{z_1}\}\right] \\
A_4 &= A_3\left[\_|\{^{r_1}/_{y_1}\}\right] \\
A_5 &= \nu\tilde{n}, t_1, t_2\ .\ \left[\_|\Gamma|\{^{ballot_1}/_{x_1}\}|\{^{r_1}/_{y_1}\}|\{^{sig_1^R}/_{z_1}\}|\{^{ballot_1}/_{x_2}\}\right] \\
A_6 &= A_5\left[\_|\{^{ballot_2}/_{b_2}\}\right] \\
A_7 &= A_6\left[\_|\{^{sig_2^R}/_{z_2}\}\right] \\
A_8 &= A_7\left[\_|\{^{r_2}/_{y_2}\}\right] \\
A_{9,j} &= A_8\left[\_|\{^{dec_i}/_{result_i}\ |\ i = 1\ldots j\}\right]
\end{aligned}
$$

$$
\begin{aligned}
\overline{A}_{9,1} &= \overline{A}_8\left[\_|\{^{dec_2}/_{result_1}\}\right] \\
\overline{A}_{9,2} &= \overline{A}_8\left[\_|\{^{dec_2}/_{result_1}\}|\{^{dec_1}/_{result_2}\}\right] \\
\overline{A}_{9,j} &= \overline{A}_8\left[\_|\{^{dec_2}/_{result_1}\}|\{^{dec_1}/_{result_2}\}|\{^{dec_i}/_{result_i}\ |\ i = 3\ldots j\}\right]
\end{aligned}
$$

Définition 5.5. Evolutions partielles des protocoles représentants les votants honnêtes.

$$
\begin{aligned}
V_{i,j}^1 &= \overline{c}_j\langle ballot_j\rangle.V_{i,j}^2 \\
V_{i,j}^2 &= c_i(z_i).V_{i,j}^3 \\
V_{i,j}^3 &= c_{RV_i}(y_i)
\end{aligned}
$$

Définition 5.6. Evolutions partielles de l'urne pour les votants honnêtes ($j = 1, 2$) et pour les votants corrompus ($j = 3, n$). La distinction est effectuée pour supprimer quelques étapes inutiles dans le second cas.

Pour $j = 1, 2$:

$$
\begin{aligned}
B^1_{j,n} &= c_j(x_j).B^{1.1}_{j,n} \\
B^{1.1}_{j,n} &= \overline{c}_{out}\langle x_j\rangle.B^{1.2}_{j,n} \\
B^{1.2}_{j,n} &= \mathsf{If}\ \phi_{\mathsf{b}}(id_j, x_j)\ \mathsf{Then}\ B^{1.3}_{j,n}\ \mathsf{Else}\ 0 \\
B^{1.3}_{j,n} &= \overline{c}_{BR}\langle ballot'_j\rangle.B^2_{j,n} \\
B^2_{j,n} &= c_{BR}(q_j).B^3_{j,n} \\
B^3_{j,n} &= \mathsf{If}\ \phi^{idp_R}_{\mathsf{s}}(hbb_j, q_j)\ \mathsf{Then}\ B^{3.1}_{j,n}\ \mathsf{Else}\ 0 \\
B^{3.1}_{j,n} &= \overline{c}_{out}\langle q_j\rangle.B^{3.2}_{j,n} \\
B^{3.2}_{j,n} &= \overline{c}_j\langle q_j\rangle.B^{3.3}_{j,n} \\
B^{3.3}_{j,n} &= \overline{c}_{BR}\langle \mathsf{OK}\rangle.B^4_{j,n} \\
B^4_{j,n} &= c_{BR}(sy_j).B^1_{j+1,n} \\
\\
B^5_{j,n} &= \overline{c}_{BD}\langle \Pi_1(x_j)\rangle.B^5_{j+1,n}
\end{aligned}
$$

Pour $j = 3, n$:

$$
\begin{aligned}
B^1_{j,n} &= c_j(x_j).B^{1.2}_{j,n} \\
\\
B^{1.2}_{j,n} &= \mathsf{If}\ \phi_{\mathsf{b}}(id_j, x_j)\ \mathsf{Then}\ B^{1.3}_{j,n}\ \mathsf{Else}\ 0 \\
B^{1.3}_{j,n} &= \overline{c}_{BR}\langle ballot'_j\rangle.B^2_{j,n} \\
B^2_{j,n} &= c_{BR}(q_j).B^3_{j,n} \\
B^3_{j,n} &= \mathsf{If}\ \phi^{idp_R}_{\mathsf{s}}(hbbj, q_j)\ \mathsf{Then}\ B^{3.2}_{j,n}\ \mathsf{Else}\ 0 \\
\\
B^{3.2}_{j,n} &= \overline{c}_j\langle q_j\rangle.B^{3.3}_{j,n} \\
B^{3.3}_{j,n} &= \overline{c}_{BR}\langle \mathsf{OK}\rangle.B^4_{j,n} \\
B^4_{j,n} &= c_{BR}(sy_j).B^1_{j+1,n} \\
B^4_{n,n} &= c_{BR}(sy_j).B^5_{1,n} \\
B^5_{j,n} &= \overline{c}_{BD}\langle \Pi_1(x_j)\rangle.B^5_{j+1,n} \\
B^5_{n,n} &= \overline{c}_{BD}\langle \Pi_1(x_j)\rangle
\end{aligned}
$$

Définition 5.7. Evolutions partielles pour le déchiffreur, on distingue le premier cas sans mélange (D) et le second avec mélange ($\overline{D}$).

$$
\begin{aligned}
D^1_{j,n} &= c_{BD}(d_j).D^1_{j+1,n} \\
D^1_{n,n} &= c_{BD}(d_n).D^2_{1,n} \\
D^2_{j,n} &= \overline{c}_{out}\langle dec_j\rangle.D^2_{j+1,n} \\
D^2_{n,n} &= \overline{c}_{out}\langle dec_n\rangle
\end{aligned}
\qquad
\begin{aligned}
\overline{D}^1_{j,n} &= c_{BD}(d_j).\overline{D}^1_{j+1,n} \\
\overline{D}^1_{n,n} &= c_{BD}(d_n).\overline{D}^2_{1,n} \\
\overline{D}^2_{1,n} &= \overline{c}_{out}\langle dec_2\rangle.\overline{D}^2_{2,n} \\
\overline{D}^2_{2,n} &= \overline{c}_{out}\langle dec_1\rangle.\overline{D}^2_{3,n} \\
\overline{D}^2_{j,n} &= \overline{c}_{out}\langle dec_j\rangle.\overline{D}^2_{j+1,n} \\
\overline{D}^2_{n,n} &= \overline{c}_{out}\langle dec_n\rangle
\end{aligned}
$$

Définition 5.8. Evolutions partielles du générateurs de reçus. Comme pour l'urne on distingue le cas des votants honnêtes de celui pour les votants corrompus :

Pour $j = 1, 2$:

$$
\begin{aligned}
R^1_{j,n} &= c_{BR}(p_j).R^2_{j,n} \\
R^2_{j,n} &= \mathsf{If}\ \phi_{\mathsf{r}}(idp_j, p_j)\ \mathsf{Then}\ R^{2.1}_{j,n}\ \mathsf{Else}\ 0 \\
R^{2.1}_{j,n} &= \overline{c}_{BR}\langle sig^R_j\rangle.R^3_{j,n} \\
R^3_{j,n} &= c_{BR}(sy_j).R^4_{j,n} \\
R^4_{j,n} &= \overline{c}_{out}\langle r_j\rangle.R^{4.1}_{j,n} \\
R^{4.1}_{j,n} &= \overline{c}_{RV_j}\langle r_j\rangle.R^{4.2}_{j,n} \\
R^{4.2}_{j,n} &= \overline{c}_{BR}\langle \mathsf{OK}\rangle.R^1_{j+1,n}
\end{aligned}
$$

pour $j = 3, n$:

$$
\begin{aligned}
R^1_{j,n} &= c_{BR}(p_j).R^2_{j,n} \\
R^2_{j,n} &= \mathsf{If}\ \phi_{\mathsf{r}}(idp_j, p_j)\ \mathsf{Then}\ R^{2.1}_{j,n}\ \mathsf{Else}\ 0 \\
R^{2.1}_{j,n} &= \overline{c}_{BR}\langle sig^R_j\rangle.R^3_{j,n} \\
R^3_{j,n} &= c_{BR}(sy_j).R^{4.1}_{j,n} \\
\\
R^{4.1}_{j,n} &= \overline{c}_{RV_j}\langle r_j\rangle.R^{4.2}_{j,n} \\
R^{4.2}_{j,n} &= \overline{c}_{BR}\langle \mathsf{OK}\rangle.R^1_{j+1,n} \\
R^{4.2}_{n,n} &= \overline{c}_{BR}\langle \mathsf{OK}\rangle
\end{aligned}
$$

5.3 Equivalence Statique

Afin de montrer l'équivalence statique nécessaire à la démonstration du théorème, on introduit encore plusieurs définitions.

Définition 5.9. Soit $id \in \{id_1, \ldots, id_n\}$. Un terme N est un *vote id-valide* si $\phi_{\mathsf{b}}^{id}(N) = \mathsf{true}$, c'est-à-dire :

$$\begin{cases} N & = & (N_1, N_2, N_3) \\ \mathsf{checksign}((N_1, N_2), \mathsf{vk}(id), N_3) & =_E & \mathsf{OK} \\ \mathsf{checkpfk}_1(\mathsf{vk}(id), N_1, N_2) & =_E & \mathsf{OK} \end{cases}.$$

Définition 5.10. Soit $N_3, \ldots, N_n$ des termes libres. On définit les substitutions suivantes :

$$\begin{aligned} \tilde{n} &= a_1, a_2, t_1, t_2, id_1, id_2, id_R \\ \sigma_N &= \{^{ballot_1}/_{x_1},\ ^{ballot_2}/_{x_2},\ ^{N_j}/_{x_j} |\ j = 3..n\} \\ \sigma_N^k &= \{^{ballot_1}/_{x_1},\ ^{ballot_2}/_{x_2},\ ^{N_j}/_{x_j} |\ j = 3..k\} \\ \Sigma_L &= \left\{^{v_1}/_{x_{vote,1}},\ ^{v_2}/_{x_{vote,2}}\right\}, \\ \Sigma_R &= \left\{^{v_2}/_{x_{vote,1}},\ ^{v_1}/_{x_{vote,2}}\right\}, \\ R &= \{^{\mathsf{dec}(\Pi_1(x_1),a_1)}/_{result_1}\}|\{^{\mathsf{dec}(\Pi_1(x_2),a_1)}/_{result_2}\} \\ \overline{R} &= \{^{\mathsf{dec}(\Pi_1(x_2),a_1)}/_{result_1}\}|\{^{\mathsf{dec}(\Pi_1(x_1),a_1)}/_{result_2}\} \end{aligned}$$

$$\begin{aligned} \theta = &\{^{\mathsf{pk}(a_1)}/_{g_1}\}|\{^{\mathsf{pk}(a_2)}/_{g_2}\}|\{^{\mathsf{pk}(a_3)}/_{g_3}\}|\{^{\mathsf{vk}(id_R)}/_{idp_R}\}|\{^{ballot_1}/_{b_1}\}|\{^{ballot_2}/_{b_2}\}| \\ &\{\{^{\mathsf{vk}(id_i)}/_{idp_i}\}|\ i = 1..n\}|\{\{^{\mathsf{dec}(\Pi_1(x_i),a_1)}/_{result_i}\}|\ i = 3..n\}| \\ &\{\{^{\mathsf{dec}(\mathsf{blind}(\mathsf{renc}(\Pi_1(x_i),a_2),\mathsf{s}(id_i)),a_3)}/_{y_i}\}|\ \{^{\mathsf{sign}(\mathsf{hash}((\mathsf{vk}(id_i),x_i)),id_R)}/_{z_i}\}|\ i = 1..n\}, \end{aligned}$$

$$\begin{aligned} \theta_0 = &\{^{\mathsf{pk}(a_1)}/_{g_1}\}|\{^{\mathsf{pk}(a_2)}/_{g_2}\}|\{^{\mathsf{pk}(a_3)}/_{g_3}\}|\{^{\mathsf{vk}(id_R)}/_{idp_R}\}|\{^{ballot_1}/_{b_1}\}|\{^{ballot_2}/_{b_2}\}| \\ &\{\{^{\mathsf{vk}(id_i)}/_{idp_i}\}|\ i = 1..n\}, \end{aligned}$$

On décompose θ_0 en éclantant les ballots b_1 et b_2 :

$$\begin{aligned} \theta_0^d = &\{^{\mathsf{pk}(a_1)}/_{g_1}\}|\{^{\mathsf{pk}(a_2)}/_{g_2}\}|\{^{\mathsf{pk}(a_3)}/_{g_3}\}|\{^{\mathsf{vk}(id_R)}/_{idp_R}\}|\{\{^{\mathsf{penc}(x_{vote,i},r_i,\mathsf{pk}(a_1))}/_{e_i}\}| \\ &\{^{\mathsf{pfk}_1(id_i,r_i,x_{vote,i},e_i)}/_{pfk_i}\}|\{^{\mathsf{sign}((e_i,pfk_i),id_i)}/_{sig_i}\}|\ i = 1,2\}|\{\{^{\mathsf{vk}(id_i)}/_{idp_i}\}|\ i = 1..n\}, \end{aligned}$$

$$\begin{aligned} \theta_1 &= \theta_0^d \cup \{^{\mathsf{dec}(\mathsf{blind}(\mathsf{renc}(\Pi_1(x_1),a_2),\mathsf{s}(id_1),a_3)}/_{y_1}\} \cup \{^{\mathsf{sign}(\mathsf{hash}((\mathsf{vk}(id_1),x_1)),id_R)}/_{z_1}\}, \\ \theta_i &= \theta_{i-1} \cup \{^{\mathsf{dec}(\mathsf{blind}(\mathsf{renc}(\Pi_1(x_i),a_2),\mathsf{s}(id_i)),a_3)}/_{y_i}\} \cup \{^{\mathsf{sign}(\mathsf{hash}((\mathsf{vk}(id_i),x_i)),id_R)}/_{z_i}\}, \\ \theta_3' &= \theta_n \cup \{^{\mathsf{dec}(\Pi_1(x_3),a_1)}/_{result_3}\} \text{ et } \theta_i' = \theta_{i-1} \cup \{^{\mathsf{dec}(\Pi_1(x_i),a_1)}/_{result_i}\} \text{ pour } i = 4..n. \end{aligned}$$

Proposition 5.11. *Soit $N_i\theta_i\Sigma$ des votes id_i-valides pour $\Sigma \in \{\Sigma_L, \Sigma_R\}$ et $i = 3..n$, on a :*

$$\phi_1 = \nu\tilde{n}.(\theta|R)\sigma_N\Sigma_L \approx_s \phi_2 = \nu\tilde{n}.(\theta|\overline{R})\sigma_N\Sigma_R.$$

On va montrer ce résultat par récurrence. On commence par un cas de base, non trivial, à démontrer :

Lemme 5.12. *Soit $\phi_2 = \nu\tilde{n}.\theta_2$. Alors, on a :*

$$\phi_2\sigma_N^2\Sigma_L \approx_s \phi_2\sigma_N^2\Sigma_R.$$

Remarque. Montrer le résultat pour $n = 2$ prouve également le résultat pour $n = 1$ et $n = 0$.

Lemme 5.13. *Soit $N_i\theta_i\sigma_N\Sigma$ des votes id_i-valides pour $\Sigma \in \{\Sigma_L, \Sigma_R\}$ et $i = 3..n$.*

$$result_1\sigma_N\Sigma_L =_E v_1 =_E result_1\sigma_N\Sigma_R$$
$$result_2\sigma_N\Sigma_L =_E v_2 =_E result_2\sigma_N\Sigma_R.$$

Lemme 5.14. *Soit $\phi = \nu\tilde{n}.\theta$ et $\phi' = \nu\tilde{n}.\theta'$ deux frames telles que $\theta = \theta' \cup \{^{M\theta'}/_y\}$ avec M un terme libre. Alors :*

$$\phi\Sigma_L \approx_s \phi\Sigma_R \iff \phi'\Sigma_L \approx_s \phi'\Sigma_R.$$

On peut maintenant réaliser la preuve de la proposition 5.11 :

PREUVE. (Idée)

Initialisation : On utilise le Lemme 5.12, prouvant que $\phi_2\sigma_N^2\Sigma_L \approx_s \phi_2\sigma_N^2\Sigma_R$.

Hypothèse de récurrence n°1 : Soit $i \geq 3$, $\phi_i = \nu\tilde{n}.\theta_i$ telle que $\phi_i\sigma_N^i\Sigma_L \approx_s \phi_i\sigma_N^i\Sigma_R$. Deplus, $\forall k \leq i$ $y_k\theta_k\sigma_N^k\Sigma_L = U_k\theta_{k-1}\sigma_N^{k-1}\Sigma_L$ pour U_k un terme libre ou $y_k\theta_k\sigma_N^k\Sigma_L =_E \mathsf{dec}(\mathsf{blind}(M_k, \mathsf{s}(id_k))$ avec M_k en forme normale.

Une fois ceci démontré, on obtient $\phi_n\sigma_N\Sigma_L \approx_s \phi_n\sigma_N\Sigma_R$ qui est un second cas de base, si l'on note $\phi_n = \phi'_2$.

Hypothèse de récurrence n°2 : Soit $i \geq 3$, $\phi'_i = \nu\tilde{n}.\theta'_i$ telle que $\phi'_i\sigma_N\Sigma_L \approx_s \phi'_i\sigma_N\Sigma_R$.

On a finalement : $\phi\sigma_N\Sigma_L \approx_s \phi\sigma_N\Sigma_R$ avec $\phi = \nu\tilde{n}.\theta$. Si l'on rajoute les résultats des Lemmes 5.13 et 5.14, on obtient le résultat :

$$\phi_1 = \nu\tilde{n}.(\theta|R)\sigma_{N_n}\Sigma_L \approx_s \phi_2 = \nu\tilde{n}.(\theta|\overline{R})\sigma_{N_n}\Sigma_R.$$

■

Remarque. La démonstration des deux récurrences est particulièrement complexe est n'est pas encore entièrement réalisée. Des problèmes sont apparus sur le tard mais seront corrigés lors du début de la thèse. C'est notamment pour cette raison que ce mémoire ne contient que l'idée de la démonstration de cette proposition, ayant jugé qu'il était inutile de joindre des preuves fausses à ce mémoire.

On peut énoncer un corollaire à partir de la Proposition 5.11, qui découle de la démonstration par récurrence :

Corollaire 5.15. *Pour $i = 0 \dots n$, $j = 3 \dots n$, et $N_k\theta_k\Sigma$ des votes id_k-valides pour $\Sigma \in \{\Sigma_L, \Sigma_R\}$ et $k = 3 \dots n$, on a :*

$$\nu\tilde{n}.\theta_i\sigma_N^i\Sigma_L \approx_s \nu\tilde{n}.\theta_i\sigma_N^i\Sigma_R$$
$$\nu\tilde{n}.\theta'_j\sigma_N\Sigma_L \approx_s \nu\tilde{n}.\theta'_j\sigma_N\Sigma_R$$

5.4 Relation

On définit la relation $\mathcal{R}$ de la manière suivante :

Définition 5.16. Etant donné un entier $n \geq 2$, $\forall\ 3 \leq j \leq n$, soit M_j des termes et N_j des termes tels que $N_j\theta_{j-1}\sigma_N^{j-1}\Sigma_L$ est un vote id_j-valide, et tels que $fv(M_j) \cup fv(N_j) \subseteq dom(A^8)$ et $(fn(M_j) \cup fn(N_j)) \cap bn(A^8) = \emptyset$. On considère la plus petite relation $\mathcal{R}$ close par équivalence structurelle et qui inclut les relations entre processus étendus suivantes :

Votant 1 :

$$A_0\left[V^1_{1,1}|V^1_{2,2}|B^1_{1,n}|R^1_{1,n}|D^1_{1,n}\right] \sim_{\mathcal{R}} A_0\left[V^1_{1,2}|V^1_{2,1}|B^1_{1,n}|R^1_{1,n}|\overline{D}^1_{1,n}\right] \tag{5.1}$$

$$A_1\left[V^2_{1,1}|V^1_{2,2}|B^{1.1}_{1,n}|R^1_{1,n}|D^1_{1,n}\right]\sigma \sim_{\mathcal{R}} A_1\left[V^2_{1,2}|V^1_{2,1}|B^{1.1}_{1,n}|R^1_{1,n}|\overline{D}^1_{1,n}\right]\tau \tag{5.2}$$

$$A_2\left[V^2_{1,1}|V^1_{2,2}|B^{1.2}_{1,n}|R^1_{1,n}|D^1_{1,n}\right]\sigma \sim_{\mathcal{R}} A_2\left[V^2_{1,2}|V^1_{2,1}|B^{1.2}_{1,n}|R^1_{1,n}|\overline{D}^1_{1,n}\right]\tau \tag{5.3}$$

$$A_2\left[V^2_{1,1}|V^1_{2,2}|B^{1.3}_{1,n}|R^1_{1,n}|D^1_{1,n}\right]\sigma \sim_{\mathcal{R}} A_2\left[V^2_{1,2}|V^1_{2,1}|B^{1.3}_{1,n}|R^1_{1,n}|\overline{D}^1_{1,n}\right]\tau \tag{5.4}$$

$$A_2\left[V^2_{1,1}|V^1_{2,2}|B^2_{1,n}|\ R^2_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right]\sigma \sim_{\mathcal{R}} A_2\left[V^2_{1,2}|V^1_{2,1}|B^2_{1,n}|R^2_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right]\tau \tag{5.5}$$

$$A_2\left[V^2_{1,1}|V^1_{2,2}|B^2_{1,n}|\ R^{2.1}_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right]\sigma \sim_{\mathcal{R}} A_2\left[V^2_{1,2}|V^1_{2,1}|B^2_{1,n}|R^{2.1}_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right]\tau \tag{5.6}$$

$$A_2\left[V^2_{1,1}|V^1_{2,2}|B^3_{1,n}\{^{sig^R_1}/_{q_1}\}|\ R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right]\sigma \sim_{\mathcal{R}} A_2\left[V^2_{1,2}|V^1_{2,1}|B^3_{1,n}\{^{sig^R_1}/_{q_1}\}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right]\tau \tag{5.7}$$

$$A_2\left[V^2_{1,1}|V^1_{2,2}|B^{3.1}_{1,n}\{^{sig^R_1}/_{q_1}\}|\ R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right]\sigma \sim_{\mathcal{R}} A_2\left[V^2_{1,2}|V^1_{2,1}|B^{3.1}_{1,n}\{^{sig^R_1}/_{q_1}\}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right]\tau \tag{5.8}$$

$$A_3\left[V^2_{1,1}|V^1_{2,2}|B^{3.2}_{1,n}\{{}^{sig^R_1}/_{q_1}\}|\ R^3_{1,n}\{{}^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right]\sigma \\ \sim_{\mathcal{R}}\ A_3\left[V^2_{1,2}|V^1_{2,1}|B^{3.2}_{1,n}\{{}^{sig^R_1}/_{q_1}\}|R^3_{1,n}\{{}^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right]\tau \tag{5.9}$$

$$A_3\left[V^3_{1,1}|V^1_{2,2}|B^{3.3}_{1,n}|\ R^3_{1,n}\{{}^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right]\sigma \\ \sim_{\mathcal{R}}\ A_3\left[V^3_{1,2}|V^1_{2,1}|B^{3.3}_{1,n}|R^3_{1,n}\{{}^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right]\tau \tag{5.10}$$

$$A_3\left[V^3_{1,1}|V^1_{2,2}|B^4_{1,n}|\ R^4_{1,n}\{{}^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right]\sigma \\ \sim_{\mathcal{R}}\ A_3\left[V^3_{1,2}|V^1_{2,1}|B^4_{1,n}|R^4_{1,n}\{{}^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right]\tau \tag{5.11}$$

$$A_4\left[V^3_{1,1}|V^1_{2,2}|B^4_{1,n}|\ R^{4.1}_{1,n}\{{}^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right]\sigma \\ \sim_{\mathcal{R}}\ A_4\left[V^3_{1,2}|V^1_{2,1}|B^4_{1,n}|R^{4.1}_{1,n}\{{}^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right]\tau \tag{5.12}$$

$$A_4\left[V^1_{2,2}|B^4_{1,n}|R^{4.2}_{1,n}|D^1_{1,n}\right]\sigma\ \sim_{\mathcal{R}}\ A_4\left[V^1_{2,1}|B^4_{1,n}|R^{4.2}_{1,n}|\overline{D}^1_{1,n}\right]\tau \tag{5.13}$$

Votant 2 :

$$A_4\left[V^1_{2,2}|B^1_{2,n}|R^1_{2,n}|D^1_{1,n}\right]\sigma\ \sim_{\mathcal{R}}\ A_4\left[V^1_{2,1}|B^1_{2,n}|R^1_{2,n}|\overline{D}^1_{1,n}\right]\tau \tag{5.14}$$

$$A_5\left[V^2_{2,2}|B^{1.1}_{2,n}|R^1_{2,n}|D^1_{1,n}\right]\Sigma_L\ \sim_{\mathcal{R}}\ A_5\left[V^2_{2,1}|B^{1.1}_{2,n}|R^1_{2,n}|\overline{D}^1_{1,n}\right]\Sigma_R \tag{5.15}$$

$$A_6\left[V^2_{2,2}|B^{1.2}_{2,n}|R^1_{2,n}|D^1_{1,n}\right]\Sigma_L\ \sim_{\mathcal{R}}\ A_6\left[V^2_{2,1}|B^{1.2}_{2,n}|R^1_{2,n}|\overline{D}^1_{1,n}\right]\Sigma_R \tag{5.16}$$

$$A_6\left[V^2_{2,2}|B^{1.3}_{2,n}|R^1_{2,n}|D^1_{1,n}\right]\Sigma_L\ \sim_{\mathcal{R}}\ A_6\left[V^2_{2,1}|B^{1.3}_{2,n}|R^1_{2,n}|\overline{D}^1_{1,n}\right]\Sigma_R \tag{5.17}$$

$$A_6\left[V^2_{2,2}|B^2_{2,n}|\ R^2_{2,n}\{{}^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right]\Sigma_L \\ \sim_{\mathcal{R}}\ A_6\left[V^2_{2,1}|B^2_{2,n}|R^2_{2,n}\{{}^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right]\Sigma_R \tag{5.18}$$

$$A_6\left[V^2_{2,2}|B^2_{2,n}|\ R^{2.1}_{2,n}\{{}^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right]\Sigma_L \\ \sim_{\mathcal{R}}\ A_6\left[V^2_{2,1}|B^2_{2,n}|R^{2.1}_{2,n}\{{}^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right]\Sigma_R \tag{5.19}$$

$$A_6\left[V^2_{2,2}|B^3_{2,n}\{{}^{sig^R_2}/_{q_2}\}|\ R^3_{2,n}\{{}^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right]\Sigma_L \\ \sim_{\mathcal{R}}\ A_6\left[V^2_{2,1}|B^3_{2,n}\{{}^{sig^R_2}/_{q_2}\}|R^3_{2,n}\{{}^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right]\Sigma_R \tag{5.20}$$

$$A_6\left[V^2_{2,2}|B^{3.1}_{2,n}\{{}^{sig^R_2}/_{q_2}\}|\ R^3_{2,n}\{{}^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right]\Sigma_L$$
$$\sim_{\mathcal{R}}\ A_6\left[V^2_{2,1}|B^{3.1}_{2,n}\{{}^{sig^R_2}/_{q_2}\}|R^3_{2,n}\{{}^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right]\Sigma_R \quad (5.21)$$

$$A_7\left[V^2_{2,2}|B^{3.2}_{2,n}\{{}^{sig^R_2}/_{q_2}\}|\ R^3_{2,n}\{{}^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right]\Sigma_L$$
$$\sim_{\mathcal{R}}\ A_7\left[V^2_{2,1}|B^{3.2}_{2,n}\{{}^{sig^R_2}/_{q_2}\}|R^3_{2,n}\{{}^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right]\Sigma_R \quad (5.22)$$

$$A_7\left[V^3_{2,2}|B^{3.3}_{2,n}|\ R^3_{2,n}\{{}^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right]\Sigma_L$$
$$\sim_{\mathcal{R}}\ A_7\left[V^3_{2,1}|B^{3.3}_{2,n}|R^3_{2,n}\{{}^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right]\Sigma_R \quad (5.23)$$

$$A_7\left[V^3_{2,2}|B^{4}_{2,n}|\ R^4_{2,n}\{{}^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right]\Sigma_L$$
$$\sim_{\mathcal{R}}\ A_7\left[V^3_{2,1}|B^{4}_{2,n}|R^4_{2,n}\{{}^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right]\Sigma_R \quad (5.24)$$

$$A_8\left[V^3_{2,2}|B^{4}_{2,n}|\ R^{4.1}_{2,n}\{{}^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right]\Sigma_L$$
$$\sim_{\mathcal{R}}\ A_7\left[V^3_{2,1}|B^{4}_{2,n}|R^{4.1}_{2,n}\{{}^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right]\Sigma_R \quad (5.25)$$

$$A_8\left[B^4_{2,n}|R^{4.2}_{2,n}|D^1_{1,n}\right]\Sigma_L\ \sim_{\mathcal{R}}\ A_7\left[B^4_{2,n}|R^{4.2}_{2,n}|\overline{D}^1_{1,n}\right]\Sigma_R \quad (5.26)$$

Soumissions : $(j = 3 \dots n)$

$$A_8\left[B^1_{j,n}|R^1_{j,n}|D^1_{1,n}|\Lambda_j\right]\Sigma_L\ \sim_{\mathcal{R}}\ A_8\left[B^1_{j,n}|R^1_{j,n}|\overline{D}^1_{1,n}|\Lambda_j\right]\Sigma_R \quad (5.27)$$
$$A_8\left[B^{1.2}_{j,n}\{{}^{M_j}/_{x_j}\}|R^1_{j,n}|D^1_{1,n}|\Lambda_j\right]\Sigma_L\ \sim_{\mathcal{R}}\ A_8\left[B^{1.2}_{j,n}\{{}^{M_j}/_{x_j}\}|R^1_{j,n}|\overline{D}^1_{1,n}|\Lambda_j\right]\Sigma_R \quad (5.28)$$
$$A_8\left[B^{1.3}_{j,n}|R^1_{j,n}|D^1_{1,n}|\Lambda'_j\right]\Sigma_L\ \sim_{\mathcal{R}}\ A_8\left[B^{1.3}_{j,n}|R^1_{j,n}|\overline{D}^1_{1,n}|\Lambda'_j\right]\Sigma_R \quad (5.29)$$
$$A_8\left[B^{2}_{j,n}|R^2_{j,n}\{{}^{ballot'_j}/_{p_j}\}|D^1_{1,n}|\Lambda'_j\right]\Sigma_L\ \sim_{\mathcal{R}}\ A_8\left[B^{2}_{j,n}|R^2_{j,n}\{{}^{ballot'_j}/_{p_j}\}|\overline{D}^1_{1,n}|\Lambda'_j\right]\Sigma_R \quad (5.30)$$
$$A_8\left[B^{2}_{j,n}|R^{2.1}_{j,n}\{{}^{ballot'_j}/_{p_j}\}|D^1_{1,n}|\Lambda'_j\right]\Sigma_L\ \sim_{\mathcal{R}}\ A_8\left[B^{2}_{j,n}|R^{2.1}_{j,n}\{{}^{ballot'_j}/_{p_j}\}|\overline{D}^1_{1,n}|\Lambda'_j\right]\Sigma_R \quad (5.31)$$

$$A_8\left[B^{3}_{j,n}\{{}^{sig^R_j}/_{q_j}\}|\ R^3_{j,n}\{{}^{ballot'_j}/_{p_j}\}|D^1_{1,n}|\Lambda'_j\right]\Sigma_L$$
$$\sim_{\mathcal{R}}\ A_8\left[B^{3}_{j,n}\{{}^{sig^R_j}/_{q_j}\}|R^3_{j,n}\{{}^{ballot'_j}/_{p_j}\}|\overline{D}^1_{1,n}|\Lambda'_j\right]\Sigma_R \quad (5.32)$$

$$A_8\left[B^{3.2}_{j,n}\{{}^{sig^R_j}/_{q_j}\}|\ R^3_{j,n}\{{}^{ballot'_j}/_{p_j}\}|D^1_{1,n}|\Lambda'_j\right]\Sigma_L$$
$$\sim_{\mathcal{R}}\ A_8\left[B^{3.2}_{j,n}\{{}^{sig^R_j}/_{q_j}\}|R^3_{j,n}\{{}^{ballot'_j}/_{p_j}\}|\overline{D}^1_{1,n}|\Lambda'_j\right]\Sigma_R \quad (5.33)$$

$$A_8\left[B^{3.3}_{j,n}|R^3_{j,n}\{^{ballot'_j}/_{p_j}\}|D^1_{1,n}|\Lambda''_j\right]\Sigma_L \sim_{\mathcal{R}} A_8\left[B^{3.3}_{j,n}|R^3_{j,n}\{^{ballot'_j}/_{p_j}\}|\overline{D}^1_{1,n}|\Lambda''_j\right]\Sigma_R \quad (5.34)$$

$$A_8\left[B^{4}_{j,n}|R^{4.1}_{j,n}\{^{ballot'_j}/_{p_j}\}|D^1_{1,n}|\Lambda''_j\right]\Sigma_L \sim_{\mathcal{R}} A_8\left[B^{4}_{j,n}|R^{4.1}_{j,n}\{^{ballot'_j}/_{p_j}\}|\overline{D}^1_{1,n}|\Lambda''_j\right]\Sigma_R \quad (5.35)$$

$$A_8\left[B^{4}_{j,n}|R^{4.2}_{j,n}|D^1_{1,n}|\Lambda_{j+1}\right]\Sigma_L \sim_{\mathcal{R}} A_8\left[B^{4}_{j,n}|R^{4.2}_{j,n}|\overline{D}^1_{1,n}|\Lambda_{j+1}\right]\Sigma_R \quad (5.36)$$

Décompte : $(k = 1 \dots n,\ j = 3 \dots n)$

$$A_8\left[B^5_{k,n}|D^1_{k,n}|\Lambda_{n+1}\right]\Sigma_L \sim_{\mathcal{R}} A_8\left[B^5_{k,n}|\overline{D}^1_{k,n}|\Lambda_{n+1}\right]\Sigma_R \quad (5.37)$$

$$A_8\left[D^2_{1,n}\{^{\Pi_1(x_1)}/_{d_1}\}|\Lambda_{n+1}\right]\Sigma_L \sim_{\mathcal{R}} A_8\left[\overline{D}^2_{1,n}\{^{\Pi_1(x_2)}/_{d_2}\}|\Lambda_{n+1}\right]\Sigma_R \quad (5.38)$$

$$A_{9,1}\left[D^2_{2,n}\{^{\Pi_1(x_2)}/_{d_2}\}|\Lambda_{n+1}\right]\Sigma_L \sim_{\mathcal{R}} \overline{A}_{9,1}\left[\overline{D}^2_{2,n}\{^{\Pi_1(x_1)}/_{d_1}\}|\Lambda_{n+1}\right]\Sigma_R \quad (5.39)$$

$$A_{9,j-1}\left[D^2_{j,n}\{^{\Pi_1(x_j)}/_{d_j}\}|\Lambda_{n+1}\right]\Sigma_L \sim_{\mathcal{R}} \overline{A}_{9,j-1}\left[\overline{D}^2_{j,n}\{^{\Pi_1(x_j)}/_{d_j}\}|\Lambda_{n+1}\right]\Sigma_R \quad (5.40)$$

$$A_{9,n}\left[\Lambda_{n+1}\right]\Sigma_L \sim_{\mathcal{R}} \overline{A}_{9,n}\left[\Lambda_{n+1}\right]\Sigma_R \quad (5.41)$$

Erreurs : (Un vote est rejeté) $(j = 3 \dots n)$

$$A_8\left[R^1_{j,n}|D^1_{1,n}|\Lambda_j|\{^{M_j}/_{x_j}\}\right]\Sigma_L \sim_{\mathcal{R}} A_8\left[R^1_{j,n}|\overline{D}^1_{1,n}|\Lambda_j|\{^{M_j}/_{x_j}\}\right]\Sigma_R \quad (5.42)$$

Il convient maintenant de montrer que cette relation $\mathcal{R}$ vérifie bien les propriétés que l'on souhaite, c'est-à-dire :

1. Si $A \longrightarrow A'$ alors $B \longrightarrow^* B'$ et $A'\mathcal{R}B'$ pour B' quelconque ;
2. Si $A \xrightarrow{\alpha} A'$ de telle sorte que $fv(\alpha) \subseteq dom(A)$ et $bn(\alpha) \cap fn(B) = \emptyset$, alors $B \longrightarrow^* \xrightarrow{\alpha} \longrightarrow^* B'$ et $A'\mathcal{R}B'$ pour B' quelconque.

La démonstration est détaillée en Annexe B.

Chapitre 6

ProVerif

Ce chapitre est une petite extension à l'étude du protocole de vote Norvégien au travers de ProVerif. On y présente rapidement l'outil ainsi que les résultats obtenus sur le protocole, montrant d'ailleurs les limites d'un tel outil.

6.1 A Propos de ProVerif

ProVerif[1] est un outil de vérification automatique de protocoles cryptographiques. Ces principales fonctionnalités sont les suivantes :

- Il peut manipuler différentes primitives cryptographiques comme la cryptographie à clef publique ou partagée (chiffrement et signatures), les fonctions de hachages, ...
- Il manipule un nombre non limité de sessions du protocole (même en parallèle) et une taille de message non limitée grâce à des approximations bien choisies. Il en découle que ProVerif peut donner de fausses attaques, mais s'il affirme qu'une propriété est vérifiée, alors c'est bel et bien le cas. Lorsque ProVerif trouve une attaque, il tente d'en reconstruire l'exécution qui permet de mettre à mal la propriété désirée.

ProVerif peut prouver les propriétés suivantes :

- Le secret. (Si l'adversaire peut obtenir ou non un certain secret.)
- Le secret « renforcé ». (L'adversaire ne voit pas la différence lorsque la valeur du secret change.)
- Authentification et propriétés de correspondance.
- Equivalences entre processus qui ne diffèrent que par des termes. (C'est l'utilisation qu'on en a faite pour ce stage.)

6.2 Notes sur la modélisation

Pour modéliser véritablement le protocole Norvégien, il a fallu utiliser un outil supplémentaire, en plus de ProVerif, car ce dernier ne peut modéliser le mélange réalisé par le déchiffreur. Or, sans mélange, l'équivalence n'est pas vérifiée. Pour introduire

1. On pourra trouver davantage d'informations sur **http ://www.proverif.ens.fr/**.

cette particularité du protocole dans le langage de ProVerif, il a fallu utiliser un outil, développé par Ben Smyth et Mark Ryan : ProSwapper [2].

Les descriptions des modélisations ne sont pas fournies dans ce mémoire car elles coïncident modulo quelques réécritures syntaxiques avec la description des protocoles en pi-calcul appliqué.

6.3 Résultats

Le tableau 6.1 regroupe les résultats obtenus en utilisant ProVerif. Il indique en fonction des participants corrompus, si l'outil a trouvé une attaque (X), s'il a terminé sans trouver d'attaque (✓), ou s'il n'a pas terminé (-). On lit le tableau en double entrée, les lignes indiquant les infrastructures corrompues et les colonnes le nombre de votants corrompus dans chaque cas, en plus des deux votants honnêtes présents dans chaque cas.

Corrompus	0 Votant	1 Votant	2 Votants
Tous honnêtes	✓	✓	✓
Urne	–	–	–
Générateur de reçus	✓	✓	✓
Urne et Générateur	–	–	–

FIGURE 6.1 – Résultats fournis par ProVerif.

Comme on peut le voir, ProVerif ne termine que dans la moitié des cas testés. Il prouve néanmoins que si tous les participants sont honnêtes, la confidentialité des votes est conservée. Ce qui est le résultat que nous avons prouvé à la main. Il permet aussi de voir que si le générateur de reçus est corrompu la confidentialité est tout de même conservée. Néanmoins, impossible pour lui de dire, lorsque l'urne est corrompu, ou lorsque l'urne et le générateur de reçus sont corrompus, si la confidentialité reste de mise.

Ces résultats montrent les limites d'un tel outil et la nécessité, parfois, de réaliser les preuves de confidentialité à la main, même si elles s'avèrent difficiles et pénibles. Le développement d'un outil de vérification automatique, spécifique aux protocoles de vote, serait éventuellement une solution pour éviter d'avoir à réaliser ces preuves. Ce sera notamment l'un des enjeux de ma thèse à venir.

2. On retrouvera toutes les précisions sur ProSwapper sur le site web de Ben Smyth : **http ://www.bensmyth.com/proswapper.php**.

Conclusion

On peut donc conclure ce mémoire sur une réponse assez satisfaisante concernant le protocole de vote Norvégien, puisque l'on a démontré la confidentialité du vote sans auditeur. Ce qui est, rappelons le, un résultat plus fort, dans un sens, que de démontrer la confidentialité avec auditeur. Néanmoins, au vu de la démonstration et de la modélisation du protocole, on peut voir que l'ajout d'un processus modélisant l'auditeur ne fera qu'augmenter la taille de la relation $\mathcal{R}$ sans influer, de part la nature privée des canaux reliant l'auditeur aux infrastructures du bureau de vote virtuel, sur l'équivalence statique à démontrer. En particulier, on peut donc obtenir, avec cette modélisation formelle, le résultat de confidentialité concernant le protocole global, ce qui n'était pas une chose évidente dans l'article qui le présentait et visait à établir sa sécurité. Il reste néanmoins encore à terminer la preuve de l'équivalence statique, plus complexe que prévue, mais qui devrait, non sans peine, être réalisée au début de ma futur thèse, encadrée par Véronique Cortier, la directrice de mon stage. On pourra également s'intéresser aux différents cas où des infrastructures particulières sont corrompues et où ProVerif n'a pas réussi à donner un résultat. (Urne Corrompue, Urne et Générateur de Reçus corrompus,...)

En termes d'expérience professionnelle, ce stage au LORIA a, pour moi, revalorisé l'optique de la recherche comme plan de carrière. En effet, je n'aurai pas réellement imaginé me lancer dans thèse après l'obtention de mon Master, davantage décidé à me lancer directement dans le monde du travail dans un côté un peu plus « pratique », pourtant, la connivence évidente entre réalité pratique et théorie présente dans ce stage m'a réellement donné goût à la recherche. En effet, se dire qu'une absence d'études (ou une mauvaise) peut conduire à des brèches de sécurité potentiellement exploitables et dangereuses, permet de voir davantage la connexion entre le travail théorique et ses incidences sur la pratique, les protocoles n'en étant qu'un exemple parmi de nombreux autres. J'ai également pu étendre mon champ de connaissances en apprenant à maitriser des éléments de théorie que l'on n'avait pas forcément détaillé lors des différents enseignements.

Pour terminer, je dirai que ce stage est annonciateur d'une thèse passionnante que j'espère bien pouvoir être en mesure de mener à son terme tout en apportant les contributions attendues, comme, par exemple, un outil de vérifications automatiques pour les protocoles de vote qui permette ainsi d'éviter que des stagiaires n'aient des preuves aussi longues et difficiles, comme celle que j'ai été amené à faire, à réaliser à la main.

Annexe A

Preuves du Chapitre 4

Dans cette annexe sont regroupées les différents lemmes et preuves nécessaires pour démontrer la décidabilité de la déduction. Ces preuves et lemmes sont rédigés en anglais car ils ont été initialement rédigés dans cette langue en vue d'une publication. Compte-tenu du contenu purement informatif et l'anglais utilisé ne gênant pas la compréhension d'un lecteur éventuellement intéressé, elles n'ont pas été traduites.

Lemme A.1.

$$St_E(St_E(\mathsf{penc}(M_1, M_2, \mathsf{pk}(M_3)))) = St_E(\mathsf{penc}(M_1, M_2, \mathsf{pk}(M_3)))$$
$$St_E(St_E(\mathsf{blind}(\mathsf{penc}(M_1, M_2, M_3), M_4))) = St_E(\mathsf{blind}(\mathsf{penc}(M_1, M_2, M_3), M_4))$$

Preuve.

$$\begin{aligned}
St_E(St_E(\mathsf{penc}(M_1, M_2, \mathsf{pk}(\mathsf{f}(M_3, M_4))))) &= St_E(\mathsf{penc}(M_1, M_2, \mathsf{pk}(\mathsf{f}(M_3, M_4)))) \\
&\cup\ St_E(\{\mathsf{penc}(M_1, M_2, \mathsf{pk}(M)) \mid M \in St_E(\mathsf{f}(M_3, M_4))\}) \\
&\cup\ St_E(\mathsf{pk}(\mathsf{f}(M_3, M_4))) \cup St_E(M_1) \cup St_E(M_2) \\
&= St_E(\mathsf{penc}(M_1, M_2, \mathsf{pk}(\mathsf{f}(M_3, M_4)))) \\
&\cup\ \{\mathsf{penc}(M_1, M_2, \mathsf{pk}(M)) \mid M \in St_E(\mathsf{f}(M_3, M_4))\} \\
&\cup\ St_E(\mathsf{pk}(M) \mid M \in St_E(\mathsf{f}(M_3, M_4))) \\
&\cup\ St_E(M_1) \cup St_E(M_2) \\
&= St_E(\mathsf{penc}(M_1, M_2, \mathsf{pk}(\mathsf{f}(M_3, M_4))))
\end{aligned}$$

$$\begin{aligned}
St_E(St_E(\mathsf{blind}(\mathsf{penc}(M_1, M_2, M_3), M_4))) &= St_E(\mathsf{blind}(\mathsf{penc}(M_1, M_2, M_3), M_4)) \\
&\cup\ St_E(\mathsf{blind}(M_1, M_4)) \\
&\cup\ St_E(\mathsf{penc}(M_1, M_2, M_3)) \cup St_E(M_4) \\
&= St_E(\mathsf{blind}(\mathsf{penc}(M_1, M_2, M_3), M_4)) \\
&\cup\ \{\mathsf{blind}(M_1, M_4)\} \\
&\cup\ St_E(M_1) \cup St_E(M_4) \\
&= St_E(\mathsf{blind}(\mathsf{penc}(M_1, M_2, M_3), M_4))
\end{aligned}$$

■

The following lemma is inspired from [BBC09].

Lemme A.2. *(Locality) Let $\phi = \nu\tilde{n}.\sigma$ be a frame in normal form, M be a closed term in normal form. If $\phi \vdash_E M$ then there exists a term ζ_M, called local recipe, such that :*

- *$fn(\zeta_M) \cap \tilde{n} = \emptyset$ and $\zeta_M\sigma =_E M$.*
- *$\forall \zeta' \in St_E(\zeta_M),\ \forall \zeta'' \in St_E(\zeta')$ we have $\zeta''\sigma\downarrow \in St_E(\phi, \zeta'\sigma\downarrow) \cup \{\Sigma_0\}$.*
  Moreover, if $\zeta'' = F(\zeta_1, \ldots, \zeta_k)$ and $F(\zeta_1\sigma\downarrow, \ldots, \zeta_k\sigma\downarrow) \xrightarrow{h} \zeta''\sigma\downarrow$ by applying a subterm
  rule then we have $\zeta''\sigma\downarrow \in St_E(\phi) \cup \{\Sigma_0\}$.

The next lemmas will be used in the proof of locality lemma.

Lemme A.3. *Let $\mathcal{R}$ be the convergent rewriting system associated to E. Let $M_1, \ldots, M_k$ be terms in normal form. If $F(M_1, \ldots, M_k)$ is not in normal form, then we have $M = F(M_1, \ldots, M_k)\downarrow$ iff $F(M_1, \ldots, M_k) \xrightarrow{h} M$.*

Preuve. Let $M_1, \ldots, M_k$ be in normal form, $F(M_1, \ldots, M_k)$ is not in normal form and $F(M_1, \ldots, M_k) \longrightarrow^* M$. Since $M_1, \ldots, M_k$ are in normal form, then the first step of reduction is in head. If the rule applied is different from (4) and (6), then it is clear that the term obtained is a proper subterm and then, is in normal form. If the rule (4) is applied, then $F = \mathsf{dec}$ and $M_1 = \mathsf{blind}(\mathsf{penc}(M_1', M_2', \mathsf{pk}(M_2)), M_3')$ with M_1 in normal form. Since $\mathsf{dec}(M_1, M_2) \xrightarrow{h} M = \mathsf{blind}(M_1', M_3')$, if M is not in normal form, then M_1' or M_3' are not in normal form. In that case, there is a contradiction with the fact that M_1 is in normal form. If the rule (6) is applied, then $F = \mathsf{renc}$ and $M_1 = \mathsf{penc}(M_1', M_2', M_3')$ with M_1 in normal form. Since $\mathsf{renc}(M_1, M_2) \xrightarrow{h} M = \mathsf{penc}(M_1', M_2', \mathsf{pk}(\mathsf{f}(M_3', M_2)))$, if M is not in normal form, then M_1', M_2', or $\mathsf{f}(M_3', M_2))$ are not in normal form. In that case, there is a contradiction with the fact that M_1 and M_2 are in normal form. Then, whatever the rule applied : $M = F(M_1, \ldots, M_k)\downarrow$ iff $F(M_1, \ldots, M_k) \xrightarrow{h} M$.

■

The following proposition provides a characterization of deduction [AC04].

Proposition A.4. *Let M be a closed term and $\phi = \nu\tilde{n}.\sigma$ be a frame. Then $\phi \vdash_E M$ iff there exists a term ζ such that $fn(\zeta) \cap \tilde{n} = \emptyset$ and $\zeta\sigma =_E M$.*

Définition A.5. Let ζ be a recipe. the length of ζ, denoted by $L(\zeta)$, is defined as follows :

$$L(\zeta) = |\zeta| + \mathrm{card}\{\mathsf{renc} \in \zeta\} + \sum_{M \in U(\zeta)} \mathrm{card}\{\mathsf{blind} \in M\}$$

where $|.|$ is the usual notion of length ($|u| = 1$, if u is a constant or a variable, and $f(a_1, \ldots, a_n) = 1 + \sum_{i=1}^{n} |a_i|$) and $U(\zeta) = \{M \in St_E(\zeta) \mid head(M) = \mathsf{dec}\}$.

Now let's prove Lemma A.2.

PREUVE. By proposition of characterization above, there exists a term ζ_M satisfying the first condition. We choose one which is minimal for the notion of length above. The second condition of Lemma A.2 is proved by induction on the size of ζ_M.

Base case : ζ_M is a variable or a name, then the second condition hold since $St_E(\zeta_M) = \{\zeta_M\}$.

Induction step : Let $\zeta_M = F(\zeta_1, \ldots, \zeta_k)$ with ζ_i are the minimal recipes of $\zeta_i\sigma\downarrow$. By induction hypothesis we have :

$$\forall \zeta' \in St_E(\zeta_i)_{i=1..k},\ \forall \zeta'' \in St_E(\zeta'),\ \zeta''\sigma\downarrow \in St_E(\phi, \zeta'\sigma\downarrow) \cup \{\Sigma_0\}.$$

To conclude that :

$$\forall \zeta' \in St_E(\zeta_M),\ \forall \zeta'' \in St_E(\zeta'),\ \zeta''\sigma\downarrow \in St_E(\phi, \zeta'\sigma\downarrow) \cup \{\Sigma_0\},$$

$$\text{or } \zeta''\sigma\downarrow \in St_E(\phi) \cup \{\Sigma_0\}, \text{(if the applied rule is a subterm rule)}$$

it is sufficient to show :

$$\forall \zeta'' \in St_E(\zeta_M),\ \zeta''\sigma\downarrow \in St_E(\phi, M) \cup \{\Sigma_0\}$$

$$\text{(respectively } \zeta''\sigma\downarrow \in St_E(\phi) \cup \{\Sigma_0\}).$$

For this, it is sufficient to prove that :

$$\forall i = 1..k,\ \zeta_i\sigma\downarrow \in St_E(\phi, M) \cup \{\Sigma_0\}$$

$$\text{(respectively } \zeta_i\sigma\downarrow \in St_E(\phi) \cup \{\Sigma_0\}),$$

since

$$\zeta_i\sigma\downarrow \in St_E(\phi, M) \cup \{\Sigma_0\} \Rightarrow \forall \zeta'' \in St_E(\zeta_i),\ \zeta''\sigma\downarrow \in St_E(\phi, \zeta_i\sigma\downarrow) \cup \{\Sigma_0\} \subseteq St_E(\phi, M) \cup \{\Sigma_0\}$$

$$\text{(respectively } \zeta_i\sigma\downarrow \in St_E(\phi) \cup \{\Sigma_0\} \Rightarrow \forall \zeta'' \in St_E(\zeta_i),\ \zeta''\sigma\downarrow \in St_E(\phi, \zeta_i\sigma\downarrow) \cup \{\Sigma_0\} \subseteq St_E(\phi) \cup \{\Sigma_0\}).$$

- If $F(\zeta_1\sigma\downarrow, \ldots, \zeta_k\sigma\downarrow)$ is in normal form, then $\forall i = 1..k$, $\zeta_i\sigma\downarrow \in St_E(F(\zeta_1\sigma\downarrow, \ldots, \zeta_k\sigma\downarrow)) = St_E(M)$ and we conclude.

- If $F(\zeta_1\sigma\downarrow, \ldots, \zeta_k\sigma\downarrow)$ is not in normal form. Since $\zeta_1\sigma\downarrow, \ldots, \zeta_k\sigma\downarrow$ are in normal form then by Lemma A.3 we have $M = F(M_1, \ldots, M_k)\downarrow$ iff $F(M_1, \ldots, M_k) \xrightarrow{h} M$. In this case we distinguish six cases according to F :

  - $F \in \{\mathsf{fst}, \mathsf{snd}\}$. This implies $k = 1$, we have $\zeta_M = \mathsf{fst}(\zeta_1)$ (or $\mathsf{snd}(\zeta_1)$) and since $\zeta_M\sigma$ can be reduced to its normal form, we have $head(\zeta_1\sigma\downarrow) = \mathsf{pair}$. Since the applied head rule is a subterm rule, we need to prove that $\zeta_1\sigma\downarrow \in St_E(\phi) \cup \{\Sigma_0\}$. We consider several cases for ζ_1 :

* If ζ_1 is a variable, so we have $\zeta_1\sigma{\downarrow} \in St_E(\phi)$ and since the applied head rule is a subterm rule, then $M \in St_E(\zeta_1\sigma{\downarrow}) \subseteq St_E(\phi)$ and we conclude.

* If $\zeta_1 = g(\zeta_1', \ldots, \zeta_k')$ and $g(\zeta_1'\sigma{\downarrow}, \ldots, \zeta_k'\sigma{\downarrow}) \xrightarrow{h} \zeta_1\sigma{\downarrow}$ by applying a rule different from (6), (7), (9), (10) or (11). Then, since the applied rule is a subterm rule, by induction hypothesis, we have $\zeta_1\sigma{\downarrow} \in St_E(\phi) \cup \{\Sigma_0\}$. Since the applied head rule is a subterm rule, $M \in St_E(\zeta_1\sigma{\downarrow}) \subseteq St_E(\phi) \cup \{\Sigma_0\}$ and we conclude. In the case were another rule is applied, we have a contradiction with the fact that $\zeta_M\sigma$ can be reduced since we have $head(\zeta_1\sigma{\downarrow}) \neq \mathsf{pair}$, so this case cannot appear.

* If $\zeta_1 = g(\zeta_1', \ldots, \zeta_k')$ and $g(\zeta_1'\sigma{\downarrow}, \ldots, \zeta_k'\sigma{\downarrow})$ is in normal form with $g \neq \mathsf{pair}$. This case cannot appear since it is in contradiction with the reduction of $\zeta_M\sigma$.

* If $\zeta_1 = \mathsf{pair}(\zeta_1', \zeta_2')$. This case cannot appear by minimality of ζ_M since ζ_1' (or ζ_2' if $F = \mathsf{snd}$) would be a smaller recipe for M than ζ_M.

- $F \in \{\mathsf{checksign}, \mathsf{checkpfk}_1, \mathsf{checkpfk}_2\}$. This case cannot appear by minimality of ζ_M since OK would be a smaller recipe that ζ_M.

- $F \in \{\mathsf{extract}_1, \mathsf{extract}_2\}$. This implies $k = 2$, we have $\zeta_M = \mathsf{extract}_1(\zeta_1, \zeta_2)$ (or $\mathsf{extract}_2(\zeta_1, \zeta_2)$) and since $\zeta_M\sigma$ can be reduced to its normal form, we have $head(\zeta_1\sigma{\downarrow}) = \mathsf{f}$ and $\zeta_2\sigma{\downarrow} \in St_E(\zeta_1\sigma{\downarrow})$. Since the applied head rule is a subterm rule, it is sufficient to prove that $\zeta_1\sigma{\downarrow} \in St_E(\phi) \cup \{\Sigma_0\}$. We consider several cases for ζ_1 :

  * If ζ_1 is a variable, so we have $\zeta_1\sigma{\downarrow} \in St_E(\phi)$ and since the applied head rule is a subterm rule, then $M \in St_E(\zeta_1\sigma{\downarrow}) \subseteq St_E(\phi)$ and we conclude.

  * If $\zeta_1 = g(\zeta_1', \ldots, \zeta_k')$ and $g(\zeta_1'\sigma{\downarrow}, \ldots, \zeta_k'\sigma{\downarrow}) \xrightarrow{h} \zeta_1\sigma{\downarrow}$ by applying a rule different from (6), (7), (9), (10) or (11). Then, since the applied rule is a subterm rule, by induction hypothesis, we have $\zeta_1\sigma{\downarrow} \in St_E(\phi) \cup \{\Sigma_0\}$. Since the applied head rule is a subterm rule, $M \in St_E(\zeta_1\sigma{\downarrow}) \subseteq St_E(\phi) \cup \{\Sigma_0\}$ and we conclude. In the case were another rule is applied, we have a contradiction with the fact that $\zeta_M\sigma$ can be reduced since we have $head(\zeta_1\sigma{\downarrow}) \neq \mathsf{f}$, so this case cannot appear.

  * If $\zeta_1 = g(\zeta_1', \ldots, \zeta_k')$ and $g(\zeta_1'\sigma{\downarrow}, \ldots, \zeta_k'\sigma{\downarrow})$ is in normal form with $g \neq \mathsf{pair}$. This case cannot appear since it is in contradiction with the reduction of $\zeta_M\sigma$.

  * If $\zeta_1 = \mathsf{f}(\zeta_1', \zeta_2')$. This case cannot appear by minimality of ζ_M since ζ_1' (or ζ_2' if $F = \mathsf{extract}_2$) would be a smaller recipe for M than ζ_M.

- $F = \mathsf{dec}$. This implies $k = 2$, we have $\zeta_M = \mathsf{dec}(\zeta_1, \zeta_2)$ and since $\zeta_M\sigma$ can be reduced to its normal form, we have $head(\zeta_1\sigma{\downarrow}) \in \{\mathsf{blind}, \mathsf{penc}\}$ and $\zeta_2\sigma{\downarrow} \in St_E(\zeta_1\sigma{\downarrow})$. Since the applied head rule is a subterm rule, it is sufficient

to prove that $\zeta_1\sigma\downarrow \in St_E(\phi) \cup \{\Sigma_0\}$. We consider several cases for ζ_1 :

* $head(\zeta_1\sigma\downarrow) = \mathsf{penc}$:

  - If ζ_1 is a variable, so we have $\zeta_1\sigma\downarrow \in St_E(\phi)$ and since the applied head rule is a subterm rule, then $M \in St_E(\zeta_1\sigma\downarrow) \subseteq St_E(\phi)$ and we conclude.

  - If $\zeta_1 = g(\zeta'_1, \ldots, \zeta'_k)$ and $g(\zeta'_1\sigma\downarrow, \ldots, \zeta'_k\sigma\downarrow) \xrightarrow{h} \zeta_1\sigma\downarrow$ by applying a rule different from (6), (7), (9), (10) or (11). Then, since the applied rule is a subterm rule, by induction hypothesis, we have $\zeta_1\sigma\downarrow \in St_E(\phi) \cup \{\Sigma_0\}$. Since the applied head rule is a subterm rule, $M \in St_E(\zeta_1\sigma\downarrow) \subseteq St_E(\phi) \cup \{\Sigma_0\}$ and we conclude. If the rule (6) is applied, then $g = \mathsf{renc}$ and $\mathsf{renc}(\zeta'_1\sigma\downarrow, \zeta'_2\sigma\downarrow) \xrightarrow{h} \zeta_1\sigma\downarrow$. This case cannot appear by minimality of ζ_M since $\mathsf{dec}(\zeta'_1, \mathsf{extract}_1(\zeta_2, \zeta'_2))$ would be a smaller recipe than $\zeta_M = \mathsf{dec}(\mathsf{renc}(\zeta'_1, \zeta'_2), \zeta_2)$ (There is one more renc in ζ_M). In the case were another rule is applied, we have a contradiction with the fact that $\zeta_M\sigma$ can be reduced since we have $head(\zeta_1\sigma\downarrow) \neq \mathsf{penc}$, so this case cannot appear.

  - If $\zeta_1 = g(\zeta'_1, \ldots, \zeta'_k)$ and $g(\zeta'_1\sigma\downarrow, \ldots, \zeta'_k\sigma\downarrow)$ is in normal form with $g \neq \mathsf{penc}$. This case cannot appear since it is in contradiction with the reduction of $\zeta_M\sigma$.

  - If $\zeta_1 = \mathsf{penc}(\zeta'_1, \zeta'_2, \zeta_3)$. This case cannot appear by minimality of ζ_M since ζ'_1 would be a smaller recipe for M than ζ_M.

* $head(\zeta_1\sigma\downarrow) = \mathsf{blind}$:

  - If ζ_1 is a variable, so we have $\zeta_1\sigma\downarrow \in St_E(\phi)$ and since the applied head rule is a subterm rule, then $M \in St_E(\zeta_1\sigma\downarrow) \subseteq St_E(\phi)$ and we conclude.

  - If $\zeta_1 = g(\zeta'_1, \ldots, \zeta'_k)$ and $g(\zeta'_1\sigma\downarrow, \ldots, \zeta'_k\sigma\downarrow) \xrightarrow{h} \zeta_1\sigma\downarrow$ by applying a rule different from (7), (9), (10) or (11).. Then, since the applied rule is a subterm rule, by induction hypothesis, we have $\zeta_1\sigma\downarrow \in St_E(\phi) \cup \{\Sigma_0\}$. Since the applied head rule is a subterm rule, $M \in St_E(\zeta_1\sigma\downarrow) \subseteq St_E(\phi) \cup \{\Sigma_0\}$ and we conclude. In the case were another rule is applied, we have a contradiction with the fact that $\zeta_M\sigma$ can be reduced since we have $head(\zeta_1\sigma\downarrow) \neq \mathsf{blind}$, so this case cannot appear.

  - If $\zeta_1 = g(\zeta'_1, \ldots, \zeta'_k)$ and $g(\zeta'_1\sigma\downarrow, \ldots, \zeta'_k\sigma\downarrow)$ is in normal form with $g \neq \mathsf{blind}$. This case cannot appear since it is in contradiction with the reduction of $\zeta_M\sigma$.

  - If $\zeta_1 = \mathsf{blind}(\zeta'_1, \zeta'_2)$. This case cannot appear by minimality of ζ_M since $\mathsf{blind}(\mathsf{dec}(\zeta'_1, \zeta_2), \zeta'_2)$ would be a smaller recipe for M (its sum of symbols blind under symbols dec is smaller) than $\zeta_M = \mathsf{dec}(\mathsf{blind}(\zeta'_1, \zeta'_2), \zeta_2)$.

- $F = \mathsf{renc}$. This implies $k = 2$, we have $\zeta_M = \mathsf{renc}(\zeta_1, \zeta_2)$ and since $\zeta_M\sigma$ can be reduced to its normal form, we have $head(\zeta_1\sigma\downarrow) = \mathsf{penc}$. By definition of extended subterms, we know that $\zeta_1\sigma\downarrow, \zeta_2\sigma\downarrow \in St_E(M)$, then we conclude.

- $F = \mathsf{unblind}$. This implies $k = 2$, we have $\zeta_M = \mathsf{unblind}(\zeta_1, \zeta_2)$ and since $\zeta_M\sigma$ can be reduced to its normal form, we have $head(\zeta_1\sigma\downarrow) = \mathsf{blind}$ and $\zeta_2\sigma\downarrow \in St_E(\zeta_1\sigma\downarrow)$. Since the applied head rule is a subterm rule, it is sufficient to prove that $\zeta_1\sigma\downarrow \in St_E(\phi) \cup \{\Sigma_0\}$. We consider several cases for ζ_1 :

  * If ζ_1 is a variable, so we have $\zeta_1\sigma\downarrow \in St_E(\phi)$ and since the applied head rule is a subterm rule, then $M \in St_E(\zeta_1\sigma\downarrow) \subseteq St_E(\phi)$, thus we conclude.

  * If $\zeta_1 = g(\zeta'_1, \ldots, \zeta'_k)$ and $g(\zeta'_1\sigma\downarrow, \ldots, \zeta'_k\sigma\downarrow) \xrightarrow{h} \zeta_1\sigma\downarrow$ by applying a rule different from (7), (9), (10) or (11). Then, since the applied rule is a subterm rule, by induction hypothesis, we have $\zeta_1\sigma\downarrow \in St_E(\phi) \cup \{\Sigma_0\}$. Since the applied head rule is a subterm rule, $M \in St_E(\zeta_1\sigma\downarrow) \subseteq St_E(\phi) \cup \{\Sigma_0\}$ and we conclude. In the case were another rule is applied, we have a contradiction with the fact that $\zeta_M\sigma$ can be reduced since we have $head(\zeta_1\sigma\downarrow) \neq \mathsf{blind}$, so this case cannot appear.

  * If $\zeta_1 = g(\zeta'_1, \ldots, \zeta'_k)$ and $g(\zeta'_1\sigma\downarrow, \ldots, \zeta'_k\sigma\downarrow)$ is in normal form with $g \neq \mathsf{blind}$. This case cannot appear since it is in contradiction with the reduction of $\zeta_M\sigma$.

  * If $\zeta_1 = \mathsf{blind}(\zeta'_1, \zeta'_2)$. This case cannot appear by minimality of ζ_M since ζ'_1 would be a smaller recipe for M than ζ_M.

■

Proposition A.6. [BBC09] *Let $\phi = \nu\tilde{n}\sigma$ be a frame such that $\sigma = \{^{M_1}/_{x_1}, \ldots, ^{M_k}/_{x_k}\}$ is in normal form, M be a term in normal form and T be the set computed by the Algorithm 1.*

1. *$\forall M' \in St_E(\phi, M)$, we have $\phi \vdash_E M'$ iff there exists a pair $(M', \zeta_{M'}) \in T$.*

2. *Moreover, the recipe $\zeta_{M'}$ computed by the algorithm is minimal and local.*

Preuve. See [BBC09].

■

Corollaire A.7. [BBC09] *For every frame ϕ in normal form and for every closed term M in normal form, $\phi \vdash_E M$ is decidable.*

Preuve. See [BBC09].

■

Annexe B

Preuves du Chapitre 5

Corollaire B.1. *Pour $i = 0 \dots n$, $j = 3 \dots n$, et $N_k\theta_k\Sigma$ des votes id_k-valides pour $\Sigma \in \{\Sigma_L, \Sigma_R\}$ et $k = 3 \dots n$, on a :*

$$\nu\tilde{n}.\theta_i\sigma_N^i\Sigma_L \approx_s \nu\tilde{n}.\theta_i\sigma_N^i\Sigma_R$$
$$\nu\tilde{n}.\theta'_j\sigma_N\Sigma_L \approx_s \nu\tilde{n}.\theta'_j\sigma_N\Sigma_R$$

PREUVE. A venir.

Définition B.2. Let $id \in \{id_1, \dots, id_n\}$. A term N is said to be a *id - valid ballot* if $\phi_{\mathsf{b}}^{id}(N) = \mathsf{true}$, equivalently :

$$\begin{cases} N & = & (N_1, N_2, N_3) \\ \mathsf{checksign}((N_1, N_2), \mathsf{vk}(id), N_3) & =_E & \mathsf{OK} \\ \mathsf{checkpfk}_1(\mathsf{vk}(id), N_1, N_2) & =_E & \mathsf{OK} \end{cases}.$$

Lemme B.3. *Let N be a id-valid ballot, thus $N = (N_1, N_2, N_3)$. Let*

$$\begin{array}{ll} N_{renc} = \mathsf{renc}(N_1, a_2), & N_{pfk}^1 = \mathsf{pfk}_2(id, a_2, N_1, N_{renc}), \\ N_{blind} = \mathsf{blind}(N_{renc}, \mathsf{s}(id)), & N_{pfk}^2 = \mathsf{pfk}_2(id, \mathsf{s}(id), N_{renc}, N_{blind}), \\ N_{hash} = \mathsf{hash}((\mathsf{vk}(id), N_1, N_2, N_3)), & R_{sign} = \mathsf{sign}(N_{hash}, id_R), \\ N' = (N, N_{renc}, N_{pfk}^1, N_{blind}, N_{pfk}^2). & \end{array}$$

Then we have $\phi_{\mathsf{b}}(id, N) =_E \phi_{\mathsf{r}}(idp, N') =_E \phi_{\mathsf{s}}^{idp_R}(N_{hash}, R_{sign}) =_E \mathsf{true}$ with $idp = \mathsf{vk}(id)$.

PREUVE. Let N be a id-valid ballot. By definition, we have that $\phi_{\mathsf{b}}(id, N) = \mathsf{true}$. According to the definition of N', we have that $N' = (N'_1, N'_2, N'_3, N'_4, N'_5)$ with $N'_1 = N = (N_1, N_2, N_3)$. Moreover, we know that $\mathsf{checkpfk}_1(idp, N_1, N_2) =_E \mathsf{checksign}((N_1, N_2), idp, N_3) =_E \mathsf{OK}$ since $\phi_{\mathsf{b}}(id, N) = \mathsf{true}$. In addition, we have :

$$\begin{aligned} \mathsf{checkpfk}_2(idp, N'_2, N'_3) &=_E \mathsf{checkpfk}_2(\mathsf{vk}(id), N_{renc}, N_{pfk}^1) \\ &=_E \mathsf{checkpfk}_2(\mathsf{vk}(id), \mathsf{renc}(N_1, a_2), \mathsf{pfk}_2(id, a_2, N_1, \mathsf{renc}(N_1, a_2))) \\ &=_E \mathsf{OK} \end{aligned}$$

and

$$\begin{aligned}\mathsf{checkpfk}_2(idp, N'_4, N'_5) &=_E \mathsf{checkpfk}_2(\mathsf{vk}(id), N_{blind}, N^2_{pfk})\\ &=_E \mathsf{checkpfk}_2(\mathsf{vk}(id), N_{blind}, \mathsf{pfk}_2(id, \mathsf{s}(id), N_{renc}, N_{blind}))\\ &\text{where } N_{blind} = \mathsf{blind}(N_{renc}, \mathsf{s}(id))\\ &=_E \mathsf{OK}.\end{aligned}$$

Then, we have $\phi_{\mathsf{r}}(idp, N') = \mathsf{true}$. Finally, we have :

$$\begin{aligned}\mathsf{checksign}(N_{hash}, idp_R, R_{sign}) &=_E \mathsf{checksign}(N_{hash}, idp_R, \mathsf{sign}(N_{hash}, id_R))\\ &=_E \mathsf{OK}\end{aligned}$$

which prove that $\phi^{idp_R}_{\mathsf{s}}(N_{hash}, R_{sign}) = \mathsf{true}$.

■

Lemme B.4. *Let N be a term such that, for some N_{rand} :*

$$\begin{aligned}N &= (N_1, N_2, N_3)\\ N_1 &= \mathsf{penc}(v, N_{rand}, \mathsf{pk}(a_1))\\ N_2 &= \mathsf{pfk}_1(id, N_{rand}, v, N_1)\\ N_3 &= \mathsf{sign}((N_1, N_2), id).\end{aligned}$$

Let $R_{rec} = \mathsf{dec}(N_{blind}, a_3)$, with N_{blind}, R_{sign} and N_{hash} the same as in Lemma B.3. Then, N is a id-valid ballot and we have :

$$\phi^{idp_R}_{\mathsf{v}}(id, N_{hash}, R_{sign}, v, N_{rec}) = \mathsf{true}.$$

PREUVE. Let N be this term. Then, N clearly satisfies Definition B.2, and :

$$\begin{aligned}\mathsf{unblind}(\mathsf{dec}(N_{blind}, a_3), \mathsf{s}(id)) &=_E \mathsf{unblind}(\mathsf{dec}(\mathsf{blind}(N_{renc}, \mathsf{s}(id)), a_3), \mathsf{s}(id))\\ &= \mathsf{unblind}(\mathsf{dec}(\mathsf{blind}(\mathsf{renc}(N_1, a_2), \mathsf{s}(id)), a_3), \mathsf{s}(id))\\ &= \mathsf{unblind}(\mathsf{dec}(\mathsf{blind}(\mathsf{renc}(\mathsf{penc}(v, N_{rand}, \mathsf{pk}(a_1)), a_2), \mathsf{s}(id)), a_3), \mathsf{s}(id))\\ &\overset{(6)}{=}_E \mathsf{unblind}(\mathsf{dec}(\mathsf{blind}(\mathsf{penc}(v, N_{rand}, \mathsf{pk}(a_1 + a_2)), \mathsf{s}(id)), a_3), \mathsf{s}(id))\\ &= \mathsf{unblind}(\mathsf{dec}(\mathsf{blind}(\mathsf{penc}(v, N_{rand}, \mathsf{pk}(a_3)), \mathsf{s}(id)), a_3), \mathsf{s}(id))\\ &\overset{(4)}{=}_E \mathsf{unblind}(\mathsf{blind}(v, \mathsf{s}(id)), \mathsf{s}(id)\\ &\overset{(7)}{=}_E v.\end{aligned}$$

Moreover :

$$\begin{aligned}\mathsf{checksign}(N_{hash}, idp_R, R_{sign}) &=_E \mathsf{checksign}(N_{hash}, idp_R, \mathsf{sign}(N_{hash}, id_R))\\ &=_E \mathsf{OK}\end{aligned}$$

which prove that $\phi^{idp_R}_{\mathsf{v}}(id, N_{hash}, R_{sign}, v, N_{rec}) = \mathsf{true}$.

■

Preuve de la relation

INTERNAL REDUCTIONS : We must show for all extended processes A and B, where $A \mathcal{R} B$, that if $A \longrightarrow A'$ for some A', then $B \longrightarrow B'$ and $A' \mathcal{R} B'$ for some B'. We observe that if $A \mathcal{R} B$ by (2), (8), (11), (15), (21), (24), (28), (33), (35), or (38) to (42) then there is no extended process A' such that $A \longrightarrow A'$. We proceed by case analysis on the remaining cases.

(1) We have $A \equiv A_0 \left[V^1_{1,1}|V^1_{2,2}|B^1_{1,n}|R^1_{1,n}|D^1_{1,n}\right]$ and $B \equiv A_0 \left[V^1_{1,2}|V^1_{2,1}|B^1_{1,n}|R^1_{1,n}|\overline{D}^1_{1,n}\right]$. If $A \longrightarrow A'$, then it must be the case that $A \equiv A_0 \left[\nu t_1.\bar{c}_1\langle ballot_1\rangle.V^2_{1,1}|V^1_{2,2}|c_1(x_1).B^{1.1}_{1,n}|R^1_{1,n}|D^1_{1,n}\right] \sigma$ and $A' \equiv A_1 \left[V^2_{1,1}|V^1_{2,2}|B^{1.1}_{1,n}|R^1_{1,n}|D^1_{1,n}\right] \sigma$. It follows from $B \equiv A_0 \left[\nu t_1.\bar{c}_1\langle ballot_1\rangle.V^2_{1,2}|V^1_{2,1}|c_1(x_1).B^{1.1}_{1,n}|R^1_{1,n}|\overline{D}^1_{1,n}\right] \tau$ that $B \longrightarrow B'$ where $B' = A_1 \left[V^2_{1,2}|V^1_{2,1}|B^{1.1}_{1,n}|R^1_{1,n}|\overline{D}^1_{1,n}\right] \tau$. Since $A' = A_1 \left[V^2_{1,1}|V^1_{2,2}|B^{1.1}_{1,n}|R^1_{1,n}|D^1_{1,n}\right] \sigma \mathcal{R} B'$, we derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(3) We have $A \equiv A_2 \left[V^2_{1,1}|V^1_{2,2}|B^{1.2}_{1,n}|R^1_{1,n}|D^1_{1,n}\right] \sigma$ and $B \equiv A_2 \left[V^2_{1,2}|V^1_{2,1}|B^{1.2}_{1,n}|R^1_{1,n}|\overline{D}^1_{1,n}\right] \tau$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C \left[V^2_{1,1}|V^1_{2,2}|\mathsf{If}\ \phi^{id_1}_{\mathsf{b}}(x_1)\ \mathsf{Then}\ B^{1.3}_{1,n}\ \mathsf{Else}\ 0|D^1_{1,n}\right] \sigma$. Since x_1 refers to $ballot_1$, it follows from Lemma B.4 applied to $ballot_1$ that it is a id_1-valid ballot and from Lemma B.3 that $\phi^{id_1}_{\mathsf{b}}(x_1)\{^{ballot_1}/_{x_1}\}\sigma = \mathsf{true}$ and $A' \equiv C \left[V^2_{1,1}|V^1_{2,2}|B^{1.3}_{1,n}|D^1_{1,n}\right] \sigma$ where $C[\_] = A_2 \left[\_|R^1_{1,n}\right]$. It follows from $B \equiv C \left[V^2_{1,2}|V^1_{2,1}|\mathsf{If}\ \phi^{id_1}_{\mathsf{b}}(x_1)\ \mathsf{Then}\ B^{1.3}_{1,n}\ \mathsf{Else}\ 0|\overline{D}^1_{1,n}\right] \tau$ and from Lemma B.4 and Lemma B.3, since $\phi^{id_1}_{\mathsf{b}}(x_1)\{^{ballot_1}/_{x_1}\}\tau = \mathsf{true}$, that $B \longrightarrow B'$ where $B' = A_2 \left[V^2_{1,2}|V^1_{2,1}|B^{1.3}_{1,n}|R^1_{1,n}|\overline{D}^1_{1,n}\right] \tau$. Since $A' \equiv A_2 \left[V^2_{1,1}|V^1_{2,2}|B^{1.3}_{1,n}|R^1_{1,n}|D^1_{1,n}\right] \sigma \mathcal{R} B'$, we derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(4) We have $A \equiv A_2 \left[V^2_{1,1}|V^1_{2,2}|B^{1.3}_{1,n}|R^1_{1,n}|D^1_{1,n}\right] \sigma$ and $B \equiv A_2 \left[V^2_{1,2}|V^1_{2,1}|B^{1.3}_{1,n}|R^1_{1,n}|\overline{D}^1_{1,n}\right] \tau$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C \left[V^2_{1,1}|V^1_{2,2}|\bar{c}_{BR}\langle ballot'_1\rangle.B^2_{1,n}|c_{BR}(p_1).R^2_{1,n}|D^1_{1,n}\right] \sigma$ and $A' \equiv C \left[V^2_{1,1}|V^1_{2,2}|B^2_{1,n}|R^2_{1,n}|D^1_{1,n}\right] \sigma$ where $C[\_] = A_2[\_]$. It follows from $B \equiv C \left[V^2_{1,2}|V^1_{2,1}|\bar{c}_{BR}\langle ballot'_1\rangle.B^2_{1,n}|c_{BR}(p_1).R^2_{1,n}|\overline{D}^1_{1,n}\right] \tau$ that $B \longrightarrow B'$ where $B' = A_2 \left[V^2_{1,2}|V^1_{2,1}|B^2_{1,n}|R^2_{1,n}|\overline{D}^1_{1,n}\right] \tau$. Since $A' \equiv A_2 \left[V^2_{1,1}|V^1_{2,2}|B^2_{1,n}|R^2_{1,n}|D^1_{1,n}\right] \sigma \mathcal{R} B'$, we derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(5) We have $A \equiv A_2 \left[V^2_{1,1}|V^1_{2,2}|B^2_{1,n}|R^2_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma$ and $B \equiv A_2 \left[V^2_{1,2}|V^1_{2,1}|B^2_{1,n}|R^2_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right] \tau$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C \left[V^2_{1,1}|V^1_{2,2}|\mathsf{If}\ \phi^{idp_1}_{\mathsf{r}}(p_1)\{^{ballot'_1}/_{p_1}\}\ \mathsf{Then}\ R^{2.1}_{1,n}\{^{ballot'_1}/_{p_1}\}\ \mathsf{Else}\ 0|D^1_{1,n}\right] \sigma$ and it follows from Lemma B.3, since $ballot_1$ is verifying Lemma B.4, that $\phi^{idp_1}_{\mathsf{r}}(p_1)\{^{ballot'_1}/_{p_1}\}\sigma = \mathsf{true}$, thus $A' \equiv C \left[V^2_{1,1}|V^1_{2,2}|R^{2.1}_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma$ where $C[\_] = A_2 \left[\_|B^2_{1,n}\right]$. It follows from $B \equiv C \left[V^2_{1,2}|V^1_{2,1}|\mathsf{If}\ \phi^{idp_1}_{\mathsf{r}}(p_1)\{^{ballot'_1}/_{p_1}\}\ \mathsf{Then}\ R^{2.1}_{1,n}\{^{ballot'_1}/_{p_1}\}\ \mathsf{Else}\ 0|\overline{D}^1_{1,n}\right] \tau$ and the lemmas that $\phi^{idp_1}_{\mathsf{r}}(p_1)\{^{ballot'_1}/_{p_1}\}\tau = \mathsf{true}$ thus $B \longrightarrow B'$ where $B' = A_2 \left[V^2_{1,2}|V^1_{2,1}|B^2_{1,n}|R^{2.1}_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right] \tau$. Since $A' \equiv A_2 \left[V^2_{1,1}|V^1_{2,2}|B^2_{1,n}|R^{2.1}_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma \mathcal{R} B'$,

we derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(6) We have $A \equiv A_2 \left[V^2_{1,1}|V^1_{2,2}|B^2_{1,n}|R^{2.1}_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma$ and $B \equiv A_2 \left[V^2_{1,2}|V^1_{2,1}|B^2_{1,n}|R^{2.1}_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right] \tau$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[V^2_{1,1}|V^1_{2,2}|c_{BR}(q_1).B^3_{1,n}|(\overline{c}_{BR}\langle sig^R_1\rangle.R^3_{1,n})\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma$ and $A' \equiv C\left[V^2_{1,1}|V^1_{2,2}|B^3_{1,n}\{^{sig^R_1}/_{q_1}\}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma$ where $C[\_] = A_2[\_]$. It follows from $B \equiv C\left[V^2_{1,2}|V^1_{2,1}|c_{BR}(q_1).B^3_{1,n}|(\overline{c}_{BR}\langle sig^R_1\rangle.R^3_{1,n})\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right] \tau$ that $B \longrightarrow B'$ where $B' = A_2\left[V^2_{1,2}|V^1_{2,1}|B^3_{1,n}\{^{sig^R_1}/_{q_1}\}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right] \tau$. Since $A' \equiv A_2\left[V^2_{1,1}|V^1_{2,2}|B^3_{1,n}\{^{sig^R_1}/_{q_1}\}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma \mathcal{R} B'$, we derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(7) We have $A \equiv A_2\left[V^2_{1,1}|V^1_{2,2}|B^3_{1,n}\{^{sig^R_1}/_{q_1}\}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma$ and $B \equiv A_2\left[V^2_{1,2}|V^1_{2,1}|B^3_{1,n}\{^{sig^R_1}/_{q_1}\}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right] \tau$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[V^2_{1,1}|V^1_{2,2}|\mathsf{If}\ \phi^{idp_R}_{\mathsf{s}}(q_1)\{^{sig^R_1}/_{q_1}\}\ \mathsf{Then}\ B^{3.1}_{1,n}\{^{sig^R_1}/_{q_1}\}\ \mathsf{Else}\ 0|D^1_{1,n}\right] \sigma$ and it follows from Lemma B.3 and Lemma B.4 applied to $ballot_1$ that $\phi^{idp_R}_{\mathsf{s}}(q_1)\{^{sig^R_1}/_{q_1}\}\sigma = \mathsf{true}$ and $A' \equiv C\left[V^2_{1,1}|V^1_{2,2}|B^{3.1}_{1,n}\{^{sig^R_1}/_{q_1}\}|D^1_{1,n}\right] \sigma$ where $C[\_] = A_2\left[\_|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}\right]$. It follows from $B \equiv C\left[V^2_{1,2}|V^1_{2,1}|\mathsf{If}\ \phi^{idp_R}_{\mathsf{s}}(q_1)\{^{sig^R_1}/_{q_1}\}\ \mathsf{Then}\ B^{3.1}_{1,n}\{^{sig^R_1}/_{q_1}\}\ \mathsf{Else}\ 0|\overline{D}^1_{1,n}\right] \tau$ and the lemmas that $\phi^{idp_R}_{\mathsf{s}}(q_1)\{^{sig^R_1}/_{q_1}\}\tau = \mathsf{true}$ and $B \longrightarrow B'$ where $B' = A_2\left[V^2_{1,2}|V^1_{2,1}|B^{3.1}_{1,n}\{^{sig^R_1}/_{q_1}\}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|\ \overline{D}^1_{1,n}\right] \tau$. Since $A' \equiv A_2\left[V^2_{1,1}|V^1_{2,2}|B^{3.1}_{1,n}\{^{sig^R_1}/_{q_1}\}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma \mathcal{R} B'$, we derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(9) We have $A \equiv A_3\left[V^2_{1,1}|V^1_{2,2}|B^{3.2}_{1,n}\{^{sig^R_1}/_{q_1}\}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma$ and $B \equiv A_3\left[V^2_{1,2}|V^1_{2,1}|B^{3.2}_{1,n}\{^{sig^R_1}/_{q_1}\}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right] \tau$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[c_1(x^1).V^3_{1,1}|V^1_{2,2}|\overline{c}_1(q_1).B^{3.3}_{1,n}|D^1_{1,n}\right] \sigma$ and $A' \equiv C\left[V^3_{1,1}|V^1_{2,2}|B^{3.3}_{1,n}|D^1_{1,n}\right] \sigma$ where $C[\_] = A_2\left[\_|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}\right]$. It follows from $B \equiv C\left[c_1(x^1).V^3_{1,2}|V^1_{2,1}|\overline{c}_1(q_1).B^{3.3}_{1,n}|\overline{D}^1_{1,n}\right] \tau$ that $B \longrightarrow B'$ where $B' = A_3\left[V^3_{1,2}|V^1_{2,1}|B^{3.3}_{1,n}|R^3_{1,n}|\overline{D}^1_{1,n}\right] \tau$. Since $A' \equiv A_3\left[V^3_{1,1}|V^1_{2,2}|B^{3.3}_{1,n}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma \mathcal{R} B'$, we derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(10) We have $A \equiv A_3\left[V^3_{1,1}|V^1_{2,2}|B^{3.3}_{1,n}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma$ and $B \equiv A_3\left[V^3_{1,2}|V^1_{2,1}|B^{3.3}_{1,n}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right] \tau$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[V^3_{1,1}|V^1_{2,2}|\overline{c}_{BR}\langle\mathsf{OK}\rangle.B^4_{1,n}|c_{BR}(sy_1).R^4_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma$ and $A' \equiv C\left[V^3_{1,1}|V^1_{2,2}|B^4_{1,n}|R^4_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma$ where $C[\_] = A_3[\_]$. It follows from $B \equiv C\left[V^3_{1,2}|V^1_{2,1}|\overline{c}_{BR}\langle\mathsf{OK}\rangle.B^4_{1,n}|c_{BR}(sy_1).R^4_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right] \tau$ that $B \longrightarrow B'$ where $B' = A_3\left[V^3_{1,2}|V^1_{2,1}|B^4_{1,n}|R^4_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right] \tau$. Since $A' \equiv A_3\left[V^3_{1,1}|V^1_{2,2}|B^4_{1,n}|R^4_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma \mathcal{R} B'$,

we derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(12) We have $A \equiv A_4 \left[V^3_{1,1}|V^1_{2,2}|B^4_{1,n}|R^{4.1}_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma$ and $B \equiv A_4 \left[V^3_{1,2}|V^1_{2,1}|B^4_{1,n}|R^{4.1}_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right] \tau$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[c_{RV_1}(y_1)|V^1_{2,2}|\overline{c}_{RV_1}\langle r_1\rangle.R^{4.2}_{1,n}|D^1_{1,n}\right] \sigma$ and $A' \equiv C\left[V^1_{2,2}|R^{4.2}_{1,n}|D^1_{1,n}\right] \sigma$ where $C[\_] = A_2\left[\_|B^4_{1,n}\right]$. It follows from $B \equiv C\left[c_{RV_1}(y_1)|V^1_{2,1}|\overline{c}_{RV_1}\langle r_1\rangle.R^{4.2}_{1,n}|\overline{D}^1_{1,n}\right] \tau$ that $B \longrightarrow B'$ where $B' = A_4\left[V^1_{2,1}|B^4_{1,n}|R^{4.2}_{1,n}|\overline{D}^1_{1,n}\right] \tau$. Since $A' \equiv A_4\left[V^1_{2,2}|B^4_{1,n}|R^{4.2}_{1,n}|D^1_{1,n}\right] \sigma \mathcal{R} B'$, we derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(13) We have $A \equiv A_4\left[V^1_{2,2}|B^4_{1,n}|R^{4.2}_{1,n}|D^1_{1,n}\right] \sigma$ and $B \equiv A_4\left[V^1_{2,1}|B^4_{1,n}|R^{4.2}_{1,n}|\overline{D}^1_{1,n}\right] \tau$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[V^1_{2,2}|c_{BR}(sy_1).B^1_{2,n}|\overline{c}_{BR}\langle \mathsf{OK}\rangle.R^1_{2,n}|D^1_{1,n}\right] \sigma$ and $A' \equiv C\left[V^1_{2,2}|B^1_{2,n}|R^1_{2,n}|D^1_{1,n}\right] \sigma$ where $C[\_] = A_4[\_]$. It follows from $B \equiv C\left[V^1_{2,1}|c_{BR}(sy_1).B^1_{2,n}|\overline{c}_{BR}\langle \mathsf{OK}\rangle.R^1_{2,n}|\overline{D}^1_{1,n}\right] \tau$ that $B \longrightarrow B'$ where $B' = A_4\left[V^1_{2,1}|B^1_{2,n}|R^1_{2,n}|\overline{D}^1_{1,n}\right] \tau$. Since $A' \equiv A_4\left[V^1_{2,2}|B^1_{2,n}|R^1_{2,n}|D^1_{1,n}\right] \sigma \mathcal{R} B'$, we derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(14) We have $A \equiv A_4\left[V^1_{2,2}|B^1_{2,n}|R^1_{2,n}|D^1_{1,n}\right] \sigma$ and $B \equiv A_4\left[V^1_{2,1}|B^1_{2,n}|R^1_{2,n}|\overline{D}^1_{1,n}\right] \tau$. If $A \longrightarrow A'$, then it must be the case that $A \equiv A_4\left[\nu t_2.\overline{c}_2\langle ballot_2\rangle.V^2_{2,2}|c_2(x_2).B^{1.1}_{2,n}|R^1_{2,n}|D^1_{1,n}\right] \Sigma_L$ and $A' \equiv A_5\left[V^2_{2,2}|B^{1.1}_{2,n}|R^1_{2,n}|D^1_{1,n}\right] \Sigma_L$. It follows from $B \equiv A_4\left[\nu t_2.\overline{c}_2\langle ballot_2\rangle.V^2_{2,1}|c_2(x_2).B^{1.1}_{2,n}|R^1_{2,n}|\overline{D}^1_{1,n}\right] \Sigma_R$ that $B \longrightarrow B'$ where $B' = A_5\left[V^2_{2,1}|B^{1.1}_{2,n}|R^1_{2,n}|\overline{D}^1_{1,n}\right] \Sigma_R$. Since $A' = A_5\left[V^2_{2,2}|B^{1.1}_{2,n}|R^1_{2,n}|D^1_{1,n}\right] \Sigma_L \mathcal{R} B'$, we derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(16) We have $A \equiv A_6\left[V^2_{2,2}|B^{1.2}_{2,n}|R^1_{2,n}|D^1_{1,n}\right] \Sigma_L$ and $B \equiv A_6\left[V^2_{2,1}|B^{1.2}_{2,n}|R^1_{2,n}|\overline{D}^1_{1,n}\right] \Sigma_R$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[V^2_{2,2}|\mathsf{If}\ \phi^{id_2}_{\mathsf{b}}(x_2)\ \mathsf{Then}\ B^{1.3}_{2,n}\ \mathsf{Else}\ 0|D^1_{1,n}\right] \Sigma_L$. Since x_2 refers to $ballot_2$, it follows from Lemma B.4 applied to $ballot_2$ that it is a id_2-valid ballot and from Lemma B.3 that $\phi^{id_2}_{\mathsf{b}}(x_2)\{^{ballot_2}/_{x_2}\}\Sigma_L = \mathsf{true}$ and $A' \equiv C\left[V^2_{2,2}|B^{1.3}_{2,n}|D^1_{1,n}\right] \Sigma_L$ where $C[\_] = A_6\left[\_|R^1_{2,n}\right]$. It follows from $B \equiv C\left[V^2_{2,1}|\mathsf{If}\ \phi^{id_2}_{\mathsf{b}}(x_2)\ \mathsf{Then}\ B^{1.3}_{2,n}\ \mathsf{Else}\ 0|\overline{D}^1_{1,n}\right] \Sigma_R$ from Lemma B.4 and Lemma B.3, since $\phi^{id_2}_{\mathsf{b}}(x_2)\{^{ballot_2}/_{x_2}\}\Sigma_R = \mathsf{true}$, that $B \longrightarrow B'$ where $B' = A_6\left[V^2_{2,1}|B^{1.3}_{2,n}|R^1_{2,n}|\overline{D}^1_{1,n}\right] \Sigma_R$. Since $A' \equiv A_6\left[V^2_{2,2}|B^{1.3}_{2,n}|R^1_{2,n}|D^1_{1,n}\right] \Sigma_L \mathcal{R} B'$, we derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(17) We have $A \equiv A_6\left[V^2_{2,2}|B^{1.3}_{2,n}|R^1_{2,n}|D^1_{1,n}\right] \Sigma_L$ and $B \equiv A_6\left[V^2_{2,1}|B^{1.3}_{2,n}|R^1_{2,n}|\overline{D}^1_{1,n}\right] \Sigma_R$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[V^2_{2,2}|\overline{c}_{BR}\langle ballot'_2\rangle.B^2_{2,n}|c_{BR}(p_2).R^2_{2,n}|D^1_{1,n}\right] \Sigma_L$ and $A' \equiv C\left[V^2_{2,2}|B^2_{2,n}|R^2_{2,n}|D^1_{1,n}\right] \Sigma_L$ where $C[\_] = A_6[\_]$. It follows from $B \equiv C\left[V^2_{2,1}|\overline{c}_{BR}\langle ballot'_2\rangle.B^2_{2,n}|c_{BR}(p_2).R^2_{2,n}|\overline{D}^1_{1,n}\right] \Sigma_R$ that $B \longrightarrow B'$ where $B' = A_6\left[V^2_{2,1}|B^2_{2,n}|R^2_{2,n}|\overline{D}^1_{1,n}\right] \Sigma_R$. Since $A' \equiv A_6\left[V^2_{2,2}|B^2_{2,n}|R^2_{2,n}|D^1_{1,n}\right] \Sigma_L \mathcal{R} B'$,

we derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(18) We have $A \equiv A_6 \left[V^2_{2,2}|B^2_{2,n}|R^2_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right] \Sigma_L$ and $B \equiv A_6 \left[V^2_{2,1}|B^2_{2,n}|\right.$ $\left.R^2_{2,n}\{^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right] \Sigma_R$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[V^2_{2,2}|\right.$ $\left.\mathsf{If}\ \phi^{idp_2}_{\mathsf{r}}(p_2)\{^{ballot'_2}/_{p_2}\}\ \mathsf{Then}\ R^{2.1}_{2,n}\{^{ballot'_2}/_{p_2}\}\ \mathsf{Else}\ 0|D^1_{1,n}\right] \Sigma_L$ and it follows from Lemma B.3, since $ballot_2$ is verifying Lemma B.4, that $\phi^{idp_2}_{\mathsf{r}}(p_2)\{^{ballot'_2}/_{p_2}\}\Sigma_L = \mathsf{true}$, thus $A' \equiv C\left[V^2_{2,2}|R^{2.1}_{2,n}|D^1_{1,n}\right] \Sigma_L$ where $C[\_] = A_6\left[\_|B^2_{2,n}\right]$. It follows from $B \equiv C\left[V^2_{2,1}|\mathsf{If}\ \phi^{idp_2}_{\mathsf{r}}(p_2)\{^{ballot'_2}/_{p_2}\}\ \mathsf{Then}\ R^{2.1}_{2,n}\{^{ballot'_2}/_{p_2}\}\ \mathsf{Else}\ 0|\overline{D}^1_{1,n}\right] \Sigma_R$ and the lemmas that $\phi^{idp_2}_{\mathsf{r}}(p_2)\{^{ballot'_2}/_{p_2}\}\Sigma_R = \mathsf{true}$ thus $B \longrightarrow B'$ where $B' = A_6\left[V^2_{2,1}|B^2_{2,n}|\right.$ $\left.R^{2.1}_{2,n}\{^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right] \Sigma_R$. Since $A' \equiv A_6\left[V^2_{2,2}|B^2_{2,n}|R^{2.1}_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right] \Sigma_L \mathcal{R} B'$, we derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(19) We have $A \equiv A_6\left[V^2_{2,2}|B^2_{2,n}|R^{2.1}_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right] \Sigma_L$ and $B \equiv A_6\left[V^2_{2,1}|B^2_{2,n}|\right.$ $\left.R^{2.1}_{2,n}\{^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right] \Sigma_R$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[V^2_{2,2}|\right.$ $\left.c_{BR}(q_2).B^3_{2,n}\{^{sig^R_2}/_{q_2}\}|\overline{c}_{BR}\langle sig^R_2\rangle.R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right] \Sigma_L$ and $A' \equiv C\left[V^2_{2,2}|\right.$ $\left.B^3_{2,n}\{^{sig^R_2}/_{q_2}\}|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right] \Sigma$ where $C[\_] = A_6[\_]$. It follows from $B \equiv$ $C\left[V^2_{2,1}|c_{BR}(q_2).B^3_{2,n}\{^{sig^R_2}/_{q_2}\}|\overline{c}_{BR}\langle sig^R_2\rangle.R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right] \Sigma_R$ that $B \longrightarrow B'$ where $B' = A_6\left[V^2_{2,1}|B^3_{2,n}\{^{sig^R_2}/_{q_2}\}|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right] \Sigma_R$. Since $A' \equiv A_6\left[V^2_{2,2}|\right.$ $\left.B^3_{2,n}\{^{sig^R_2}/_{q_2}\}|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right] \Sigma_L \mathcal{R} B'$, we derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(20) We have $A \equiv A_6\left[V^2_{2,2}|B^3_{2,n}\{^{sig^R_2}/_{q_2}\}|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right] \Sigma_L$ and $B \equiv A_6\left[V^2_{2,1}|\right.$ $\left.B^3_{2,n}\{^{sig^R_2}/_{q_2}\}|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right] \Sigma_R$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[V^2_{2,2}|\mathsf{If}\ \phi^{idp_R}_{\mathsf{s}}(q_2)\{^{sig^R_2}/_{q_2}\}\ \mathsf{Then}\ B^{3.1}_{2,n}\{^{sig^R_2}/_{q_2}\}\ \mathsf{Else}\ 0|D^1_{1,n}\right] \Sigma_L$ and it follows from Lemma B.3 and Lemma B.4 applied to $ballot_2$ that $\phi^{idp_R}_{\mathsf{s}}(q_2)\{^{sig^R_2}/_{q_2}\}\Sigma_L =$ true and $A' \equiv C\left[V^2_{2,2}|B^{3.1}_{2,n}\{^{sig^R_2}/_{q_2}\}|D^1_{1,n}\right] \Sigma_L$ where $C[\_] = A_6\left[\_|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}\right]$. It follows from $B \equiv C\left[V^2_{2,1}|\mathsf{If}\ \phi^{idp_R}_{\mathsf{s}}(q_2)\{^{sig^R_2}/_{q_2}\}\ \mathsf{Then}\ B^{3.1}_{2,n}\{^{sig^R_2}/_{q_2}\}\ \mathsf{Else}\ 0|\overline{D}^1_{1,n}\right] \Sigma_R$ and the lemmas that $\phi^{idp_R}_{\mathsf{s}}(q_2)\{^{sig^R_2}/_{q_2}\}\Sigma_R = \mathsf{true}$ and $B \longrightarrow B'$ where $B' = A_6\left[V^2_{2,1}|B^{3.1}_{2,n}\{^{sig^R_2}/_{q_2}\}|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right] \Sigma_R$. Since $A' \equiv A_6\left[V^2_{2,2}|B^{3.1}_{2,n}\{^{sig^R_2}/_{q_2}\}|\right.$ $\left.R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right] \Sigma_L \mathcal{R} B'$, we derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(22) We have $A \equiv A_7\left[V^2_{2,2}|B^{3.2}_{2,n}\{^{sig^R_2}/_{q_2}\}|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right] \Sigma_L$ and $B \equiv A_7\left[V^2_{2,1}|\right.$ $\left.B^{3.2}_{2,n}\{^{sig^R_2}/_{q_2}\}|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right] \Sigma_R$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[c_2(x^2).V^3_{2,2}|\overline{c}_2(q_2).B^{3.3}_{2,n}|D^1_{1,n}\right] \Sigma_L$ and $A' \equiv C\left[V^3_{2,2}|B^{3.3}_{2,n}|D^1_{1,n}\right] \Sigma_L$ where $C[\_] = A_7\left[\_|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}\right]$. It follows from $B \equiv C\left[c_2(x^2).V^3_{2,1}|\overline{c}_2(q_2).B^{3.3}_{2,n}|\overline{D}^1_{1,n}\right] \Sigma_R$

that $B \longrightarrow B'$ where $B' = A_7\left[V^3_{2,1}|B^{3.3}_{2,n}|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right]\Sigma_R$. Since $A' \equiv A_7\left[V^3_{2,2}|B^{3.3}_{2,n}|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right]\Sigma_L\ \mathcal{R}\ B'$, we derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(23) We have $A \equiv A_7\left[V^3_{2,2}|B^{3.3}_{2,n}|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right]\Sigma_L$ and $B \equiv A_7\left[V^3_{2,1}|B^{3.3}_{2,n}|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right]\Sigma_R$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[V^3_{2,2}|\overline{c}_{BR}\langle\mathsf{OK}\rangle.B^4_{2,n}|c_{BR}(sy_2).R^4_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right]\Sigma_L$ and $A' \equiv C\left[V^3_{2,2}|B^4_{2,n}|R^4_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right]\Sigma_L$ where $C[\_] = A_7[\_]$. It follows from $B \equiv C\left[V^3_{2,1}|\overline{c}_{BR}\langle\mathsf{OK}\rangle.B^4_{2,n}|c_{BR}(sy_2).R^4_{2,n}\{^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right]\Sigma_R$ that $B \longrightarrow B'$ where $B' = A_7\left[V^3_{2,1}|B^4_{2,n}|R^4_{2,n}\{^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right]\Sigma_R$. Since $A' \equiv A_7\left[V^3_{2,2}|B^4_{2,n}|R^4_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right]\Sigma_R\ \mathcal{R}\ B'$, we derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(25) We have $A \equiv A_8\left[V^3_{2,2}|B^4_{2,n}|R^{4.1}_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right]\Sigma_L$ and $B \equiv A_8\left[V^3_{2,1}|B^4_{2,n}|R^{4.1}_{2,n}\{^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right]\Sigma_R$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[c_{RV_2}(y_2)|\overline{c}_{RV_2}\langle r_2\rangle.R^{4.2}_{2,n}|D^1_{1,n}\right]\Sigma_L$ and $A' \equiv C\left[R^{4.2}_{2,n}|D^1_{1,n}\right]\Sigma_L$ where $C[\_] = A_8\left[\_|B^4_{2,n}\right]$. It follows from $B \equiv C\left[c_{RV_2}(y_2)|\overline{c}_{RV_2}\langle r_2\rangle.R^{4.2}_{2,n}|\overline{D}^1_{1,n}\right]\Sigma_R$ that $B \longrightarrow B'$ where $B' = A_8\left[B^4_{2,n}|R^{4.2}_{2,n}|\overline{D}^1_{1,n}\right]\Sigma_R$. Since $A' \equiv A_8\left[B^4_{2,n}|R^{4.2}_{2,n}|D^1_{1,n}\right]\Sigma_L\ \mathcal{R}\ B'$, we derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(26) We have $A \equiv A_8\left[B^4_{2,n}|R^{4.2}_{2,n}|D^1_{1,n}\right]\Sigma_L$ and $B \equiv A_8\left[B^4_{2,n}|R^{4.2}_{2,n}|\overline{D}^1_{1,n}\right]\Sigma_R$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[c_{BR}(sy_2).B^1_{3,n}|\overline{c}_{BR}\langle\mathsf{OK}\rangle.R^1_{3,n}|D^1_{1,n}\right]\Sigma_L$ and $A' \equiv C\left[B^1_{3,n}|R^1_{3,n}|D^1_{1,n}\right]\Sigma_L$ where $C[\_] = A_8[\_]$. It follows from $B \equiv C\left[c_{BR}(sy_2).B^1_{3,n}|\overline{c}_{BR}\langle\mathsf{OK}\rangle.R^1_{3,n}|\overline{D}^1_{1,n}\right]\Sigma_R$ that $B \longrightarrow B'$ where $B' = A_8\left[B^1_{3,n}|R^1_{3,n}|\overline{D}^1_{1,n}\right]\Sigma_R$. Since $A' \equiv A_8\left[B^1_{3,n}|R^1_{3,n}|D^1_{1,n}\right]\Sigma_L\ \mathcal{R}\ B'$, we derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(28) We have $A \equiv A_8\left[B^{1.2}_{j,n}\{^{M_j}/_{x_j}\}|R^1_{j,n}|D^1_{1,n}|\Lambda_j\right]\Sigma_L$ and $B \equiv A_8\left[B^{1.2}_{j,n}\{^{M_j}/_{x_j}\}|R^1_{j,n}|\overline{D}^1_{1,n}|\Lambda_j\right]\Sigma_R$ for some $j \in \{3,\ldots,n\}$, $N_3,\ldots,N_{j-1}$ such that $N_k\theta_{k-1}\sigma_N^{k-1}\Sigma_L$ are id_k-valid ballots for $k = 3\ldots j-1$, and a term M_j such that $fv(M_j) \cup \bigcup_{3\leq i\leq j-1} fv(N_i) \subseteq dom(A^8)$ and $(fn(M_j)\cup\bigcup_{3\leq i\leq j-1} fn(N_i))\cap bn(A^8) = \emptyset$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[\mathsf{If}\ \phi^{id_j}_{\mathsf{b}}(x_j)\{^{M_j}/_{x_j}\}\ \mathsf{Then}\ P\ \mathsf{Else}\ 0|D^1_{1,n}\right]\Sigma_L$ with $P = B^{1.3}_{j,n}$. We also have $B \equiv C\left[\mathsf{If}\ \phi^{id_j}_{\mathsf{b}}(x_j)\{^{M_j}/_{x_j}\}\ \mathsf{Then}\ P\ \mathsf{Else}\ 0|\overline{D}^1_{1,n}\right]\Sigma_R$, where $C[\_] = A_8[\_|R^1_{j,n}|\Lambda_j]$. We procced by case analysis on the structure of A' :

- If $A' \equiv C\left[P|D^1_{1,n}\right]\Sigma_L$, then $M_j\theta_{j-1}\sigma_N^{j-1}\Sigma_L$, which is equal to $M_j\Lambda_j\Sigma_L$ in the A_8 context, must have passed $\phi^{id_j}_{\mathsf{b}}$, is a id_j-valid ballot. From Corrolary B.1, since we deduce that $M_j\Lambda_j\Sigma_R$ is also a valid ballot and then $B \longrightarrow B' = A_8\left[B^{1.3}_{j,n}|R^1_{j,n}|\overline{D}^1_{1,n}|\Lambda'_j\right]\Sigma_R$. Since $A' \equiv A_8\left[B^{1.3}_{j,n}|R^1_{j,n}|D^1_{1,n}|\Lambda'_j\right]\Sigma_L\ \mathcal{R}\ B'$, we de-

rive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

- If $A' \equiv C\left[0|D^1_{1,n}\right]\Sigma_L$, then $M_j\Lambda_j\Sigma_L$ must not have passed $\phi^{id_j}_{\mathsf{b}}$, then $M_j\Lambda_j\Sigma_L$ is not id_j-valid ballot. From Corrolary B.1, we deduce that $M_j\Lambda_j\Sigma_R$ is not a valid ballot either and then $B \longrightarrow B' = A_8\left[0|R^1_{j,n}|\overline{D}^1_{1,n}|\Lambda_j|\{^{M_j}/_{x_j}\}\right]\Sigma_R$. Since $A' \equiv A_8\left[0|R^1_{j,n}|D^1_{1,n}|\Lambda_j|\{^{M_j}/_{x_j}\}\right]\Sigma_L \mathcal{R} B'$, we derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(29) We have $A \equiv A_8\left[B^{1.3}_{j,n}|R^1_{j,n}|D^1_{1,n}|\Lambda'_j\right]\Sigma_L$ and $B \equiv A_8\left[B^{1.3}_{j,n}|R^1_{j,n}|\overline{D}^1_{1,n}|\Lambda'_j\right]\Sigma_R$ for some $j \in \{3,\ldots,n\}$, $N_3,\ldots,N_j$ such that $N_k\theta_{k-1}\sigma_N^{k-1}\Sigma_L$ are id_k-valid ballots for $k = 3\ldots j$ and $\bigcup_{3\leq i\leq j} fv(N_i) \subseteq dom(A^8)$ and $\bigcup_{3\leq i\leq j} fn(N_i)\cap bn(A^8) = \emptyset$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[\overline{c}_{BR}\langle ballot'_j\rangle.B^2_{j,n}|c_{BR}(p_j).R^2_{j,n}\{^{ballot'_j}/_{p_j}\}|D^1_{1,n}\right]\Sigma_L$ and $A' \equiv C\left[B^2_{j,n}|R^2_{j,n}\{^{ballot'_j}/_{p_j}\}|D^1_{1,n}\right]\Sigma_L$ where $C[\_] = A_8\left[\_|\Lambda'_j\right]$. It follows from $B \equiv C\left[\overline{c}_{BR}\langle ballot'_j\rangle.B^2_{j,n}|c_{BR}(p_j).R^2_{j,n}\{^{ballot'_j}/_{p_j}\}|\overline{D}^1_{1,n}\right]\Sigma_R$ that $B \longrightarrow B'$ where $B' = A_8\left[B^2_{j,n}|R^2_{j,n}\{^{ballot'_j}/_{p_j}\}|\overline{D}^1_{1,n}|\Lambda'_j\right]\Sigma_R$. Since $A' \equiv A_8\left[B^2_{j,n}|R^2_{j,n}\{^{ballot'_j}/_{p_j}\}|D^1_{1,n}|\Lambda'_j\right]\Sigma_L \mathcal{R} B'$, we derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(30) We have $A \equiv A_8\left[B^2_{j,n}|R^2_{j,n}\{^{ballot'_j}/_{p_j}\}|D^1_{1,n}|\Lambda'_j\right]\Sigma_L$ and $B \equiv A_8\left[B^2_{j,n}|R^2_{j,n}\{^{ballot'_j}/_{p_j}\}|\overline{D}^1_{1,n}|\Lambda'_j\right]\Sigma_R$ for some $j \in \{3,\ldots,n\}$, $N_3,\ldots,N_j$ such that $N_k\theta_{k-1}\sigma_N^{k-1}\Sigma_L$ are id_k-valid ballots for $k = 3\ldots j$ and $\bigcup_{3\leq i\leq j} fv(N_i) \subseteq dom(A^8)$ and $\bigcup_{3\leq i\leq j} fn(N_i)\cap bn(A^8) = \emptyset$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[\mathsf{If}\ \phi^{idp_j}_{\mathsf{r}}(p_j)\{^{ballot'_j}/_{p_j}\}\ \mathsf{Then}\ R^{2.1}_{j,n}\{^{ballot'_j}/_{p_j}\}\ \mathsf{Else}\ 0|D^1_{1,n}\right]\Sigma_L$ and it follows from Lemma B.3, since $N_j\theta_{j-1}\sigma_N^{j-1}\Sigma_L$ is a id_j-valid ballot, that $\phi^{idp_j}_{\mathsf{r}}(p_j)\{^{ballot'_j}/_{p_j}\}\Sigma_L = \mathsf{true}$, thus $A' \equiv C[R^{2.1}_{j,n}\{^{ballot'_j}/_{p_j}\}|D^1_{1,n}]\Sigma_L$ where $C[\_] = A_8\left[\_|B^2_{j,n}|\Lambda'_j\right]$. It follows from $B \equiv C\left[\mathsf{If}\ \phi^{idp_j}_{\mathsf{r}}(p_j)\{^{ballot'_j}/_{p_j}\}\ \mathsf{Then}\ R^{2.1}_{j,n}\{^{ballot'_j}/_{p_j}\}\ \mathsf{Else}\ 0|\overline{D}^1_{1,n}\right]\Sigma_R$ and from Lemma B.3 that $\phi^{idp_j}_{\mathsf{r}}(p_j)\{^{ballot'_j}/_{p_j}\}\Sigma_R = \mathsf{true}$ since $N_j\theta_{j-1}\sigma_N^{j-1}\Sigma_R$ is a id_j-valid ballot. Thus $B \longrightarrow B'$ where $B' = A_8\left[B^2_{j,n}|R^{2.1}_{j,n}\{^{ballot'_j}/_{p_j}\}|\overline{D}^1_{1,n}|\Lambda'_j\right]\Sigma_R$. Since $A' \equiv A_8\left[B^2_{j,n}|R^{2.1}_{j,n}\{^{ballot'_j}/_{p_j}\}|D^1_{1,n}|\Lambda'_j\right]\Sigma_L \mathcal{R} B'$, we derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(31) We have $A \equiv A_8\left[B^2_{j,n}|R^{2.1}_{j,n}\{^{ballot'_j}/_{p_j}\}|D^1_{1,n}|\Lambda'_j\right]\Sigma_L$ and $B \equiv A_8\left[B^2_{j,n}|R^{2.1}_{j,n}\{^{ballot'_j}/_{p_j}\}|\overline{D}^1_{1,n}|\Lambda'_j\right]\Sigma_R$ for some $j \in \{3,\ldots,n\}$, $N_3,\ldots,N_j$ such that $N_k\theta_{k-1}\sigma_N^{k-1}\Sigma_L$ are id_k-valid ballots for $k = 3\ldots j$ and $\bigcup_{3\leq i\leq j} fv(N_i) \subseteq dom(A^8)$ and $\bigcup_{3\leq i\leq j} fn(N_i)\cap bn(A^8) = \emptyset$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[c_{BR}(q_j).B^3_{j,n}\{^{sig^R_j}/_{q_j}\}|\right.$

$\overline{c}_{BR}\langle sig_R^j\rangle.R_{j,n}^3\{^{ballot'_j}/_{p_j}\}|D_{1,n}^1\Big]\,\Sigma_L$ and $A' \equiv C\left[B_{j,n}^3\{^{sig_j^R}/_{q_j}\}|R_{j,n}^3\{^{ballot'_j}/_{p_j}\}|D_{1,n}^1\right]\Sigma_L$ where $C[\_] = A_8\left[\_|\Lambda_j^{'}\right]$. It follows from $B \equiv C\left[c_{BR}(q_j).B_{j,n}^3\{^{sig_j^R}/_{q_j}\}|\overline{c}_{BR}(sig_R^j).\right.$ $\left.R_{j,n}^3\{^{ballot'_j}/_{p_j}\}|\overline{D}_{1,n}^1\right]\Sigma_R$ that $B \longrightarrow B'$ where $B' = A_8\left[B_{j,n}^3\{^{sig_j^R}/_{q_j}\}|R_{j,n}^3\{^{ballot'_j}/_{p_j}\}|\right.$ $\left.\overline{D}_{1,n}^1|\Lambda_j^{'}\right]\Sigma_R$. Since $A' \equiv A_8\left[B_{j,n}^3\{^{sig_j^R}/_{q_j}\}|R_{j,n}^3\{^{ballot'_j}/_{p_j}\}|D_{1,n}^1|\Lambda_j^{'}\right]\Sigma_L\ \mathcal{R}\ B'$, we derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(32) We have $A \equiv A_8\left[B_{j,n}^3\{^{sig_j^R}/_{q_j}\}|R_{j,n}^3\{^{ballot'_j}/_{p_j}\}|D_{1,n}^1|\Lambda_j^{'}\right]\Sigma_L$ and $B \equiv A_8\left[B_{j,n}^3\{^{sig_j^R}/_{q_j}\}|\right.$ $\left.R_{j,n}^3\{^{ballot'_j}/_{p_j}\}|\overline{D}_{1,n}^1|\Lambda_j^{'}\right]\Sigma_R$ for some $j \in \{3,\ldots,n\}$, $N_3,\ldots,N_j$ such that $N_k\theta_{k-1}\sigma_N^{k-1}\Sigma_L$ are id_k-valid ballots for $k = 3\ldots j$ and $\bigcup_{3\leq i\leq j} fv(N_i) \subseteq dom(A^8)$ and $\bigcup_{3\leq i\leq j} fn(N_i)\cap$ $bn(A^8) = \emptyset$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\Big[\mathsf{If}\ \phi_{\mathsf{s}}^{idp_R}(q_j)\{^{sig_j^R}/_{q_j}\}$ $\mathsf{Then}\ B_{j,n}^{3.2}\{^{sig_j^R}/_{q_j}\}\ \mathsf{Else}\ 0|D_{1,n}^1\Big]\Sigma_L$ and it follows from Lemma B.3, since $N_j\theta_{j-1}$ $\sigma_N^{j-1}\Sigma_L$ is a id_j-valid ballot, that $\phi_{\mathsf{s}}^{idp_R}(q_j)\{^{sig_j^R}/_{q_j}\}\Sigma_L = \mathsf{true}$, thus $A' \equiv C\left[B_{j,n}^{3.2}\{^{sig_j^R}/_{q_j}\}|\right.$ $\left.D_{1,n}^1\right]\Sigma_L$ where $C[\_] = A_8\left[\_|R_{j,n}^3\{^{ballot'_j}/_{p_j}\}|\Lambda_j^{'}\right]$. It follows from $B \equiv C[\mathsf{If}$ $\phi_{\mathsf{s}}^{idp_R}(q_j)\{^{sig_j^R}/_{q_j}\}\ \mathsf{Then}\ B_{j,n}^{3.2}\{^{sig_j^R}/_{q_j}\}\ \mathsf{Else}\ 0|\overline{D}_{1,n}^1\Big]\Sigma_R$ and from Lemma B.3 that $\phi_{\mathsf{s}}^{idp_R}(q_j)\{^{sig_j^R}/_{q_j}\}\Sigma_R = \mathsf{true}$ since $N_j\theta_{j-1}\sigma_N^{j-1}\Sigma_R$ is a id_j-valid ballot. Thus $B \longrightarrow$ B' where $B' = A_8\left[B_{j,n}^{3.2}\{^{sig_j^R}/_{q_j}\}|R_{j,n}^3\{^{ballot'_j}/_{p_j}\}|\overline{D}_{1,n}^1|\Lambda_j^{'}\right]\Sigma_R$. Since $A' \equiv A_8\left[B_{j,n}^{3.2}\{^{sig_j^R}/_{q_j}\}|\right.$ $\left.R_{j,n}^3\{^{ballot'_j}/_{p_j}\}|D_{1,n}^1|\Lambda_j^{'}\right]\Sigma_L\ \mathcal{R}\ B'$, we derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(34) We have $A \equiv A_8\left[B_{j,n}^{3.3}|R_{j,n}^3\{^{ballot'_j}/_{p_j}\}|D_{1,n}^1|\Lambda_j^{''}\right]\Sigma_L$ and $B \equiv A_8\left[B_{j,n}^{3.3}|R_{j,n}^3\{^{ballot'_j}/_{p_j}\}|\right.$ $\left.\overline{D}_{1,n}^1|\Lambda_j^{''}\right]\Sigma_R$ for some $j \in \{3,\ldots,n\}$, $N_3,\ldots,N_j$ such that $N_k\theta_{k-1}\sigma_N^{k-1}\Sigma_L$ are id_k-valid ballots for $k = 3\ldots j$ and $\bigcup_{3\leq i\leq j} fv(N_i) \subseteq dom(A^8)$ and $\bigcup_{3\leq i\leq j} fn(N_i)\cap$ $bn(A^8) = \emptyset$. If $A \longrightarrow A'$, then it must be the case that $A \equiv C\left[\overline{c}_{BR}\langle\mathsf{OK}\rangle.B_{j,n}^4|c_{BR}(sy_j).\right.$ $\left.R_{j,n}^{4.1}\{^{ballot'_j}/_{p_j}\}|D_{1,n}^1\right]\Sigma_L$ and $A' \equiv C\left[B_{j,n}^4|R_{j,n}^{4.1}\{^{ballot'_j}/_{p_j}\}|D_{1,n}^1\right]\Sigma_L$ where $C[\_] =$ $A_8\left[\_|\Lambda_j^{''}\right]$. It follows from $B \equiv C\left[\overline{c}_{BR}\langle\mathsf{OK}\rangle.B_{j,n}^4|c_{BR}(sy_j).R_{j,n}^{4.1}\{^{ballot'_j}/_{p_j}\}|\overline{D}_{1,n}^1\right]\Sigma_R$ that $B \longrightarrow B'$ where $B' = A_8\left[B_{j,n}^4|R_{j,n}^{4.1}\{^{ballot'_j}/_{p_j}\}|\overline{D}_{1,n}^1|\Lambda_j^{''}\right]\Sigma_R$. Since $A' \equiv$ $A_8\left[B_{j,n}^4|R_{j,n}^{4.1}\{^{ballot'_j}/_{p_j}\}|D_{1,n}^1|\Lambda_j^{''}\right]\Sigma_L\ \mathcal{R}\ B'$, we derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(36) We have $A \equiv A_8\left[B_{j,n}^4|R_{j,n}^{4.2}|D_{1,n}^1|\Lambda_{j+1}\right]\Sigma_L$ and $B \equiv A_8\left[B_{j,n}^4|R_{j,n}^{4.2}|\overline{D}_{1,n}^1|\Lambda_{j+1}\right]\Sigma_R$. for some $j \in \{3,\ldots,n\}$, $N_3,\ldots,N_j$ such that $N_k\theta_{k-1}\sigma_N^{k-1}\Sigma_L$ are id_k-valid ballots for $k = 3\ldots j$ and $\bigcup_{3\leq i\leq j} fv(N_i) \subseteq dom(A^8)$ and $\bigcup_{3\leq i\leq j} fn(N_i) \cap bn(A^8) = \emptyset$. There are two subcases :

- If $A \longrightarrow A'$ and $j < n$, then it must be the case that $A \equiv C\left[c_{BR}(sy_j).B_{j+1,n}^1|\right.$

$\overline{c}_{BR}\langle \mathsf{OK}\rangle.R^1_{j+1,n}|D^1_{1,n}\left]\,\Sigma_L\right.$ and $A' \equiv C\left[B^1_{j+1,n}|R^1_{j+1,n}|D^1_{1,n}\right]\Sigma_L$ where $C\,[\_] = A_8\,[\_|\Lambda_{j+1}]$. It follows from $B \equiv C\left[c_{BR}(sy_j).B^1_{j+1,n}|\overline{c}_{BR}\langle \mathsf{OK}\rangle.R^1_{j+1,n}|\overline{D}^1_{1,n}\right]\Sigma_R$ that $B \longrightarrow B'$ where $B' = A_8\left[B^1_{j+1,n}|R^1_{j+1,n}|\overline{D}^1_{1,n}|\Lambda_{j+1}\right]\Sigma_R$. Since $A' \equiv A_8\left[B^1_{j+1,n}|R^1_{j+1,n}|D^1_{1,n}|\Lambda_{j+1}\right]\Sigma_L\ \mathcal{R}\ B'$, we derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

- If $A \longrightarrow A'$ and $j = n$, then it must be the case that $A \equiv C\left[c_{BR}(sy_n).B^5_{1,n}|\overline{c}_{BR}\langle \mathsf{OK}\rangle|D^1_{1,n}\right]\Sigma_L$ and $A' \equiv C\left[B^5_{1,n}|D^1_{1,n}\right]\Sigma_L$ where $C\,[\_] = A_8\,[\_|\Lambda_{n+1}]$. It follows from $B \equiv C\left[c_{BR}(sy_n).B^5_{1,n}|\overline{c}_{BR}\langle \mathsf{OK}\rangle|\overline{D}^1_{1,n}\right]\Sigma_R$ that $B \longrightarrow B'$ where $B' = A_8\left[B^5_{1,n}|\overline{D}^1_{1,n}|\Lambda_{n+1}\right]\Sigma_R$. Since $A' \equiv A_8\left[B^5_{1,n}|D^1_{1,n}|\Lambda_{n+1}\right]\Sigma_L\ \mathcal{R}\ B'$, we derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(37) We have $A \equiv A_8\left[B^5_{k,n}|D^1_{k,n}|\Lambda_{n+1}\right]\Sigma_L$ and $B \equiv A_8\left[B^5_{k,n}|\overline{D}^1_{k,n}|\Lambda_{n+1}\right]\Sigma_R$ for some $j \in \{3,\ldots,n\}$, $N_3,\ldots,N_j$ such that $N_k\theta_{k-1}\sigma_N^{k-1}\Sigma_L$ are id_k-valid ballots for $k = 3\ldots j$ and $\bigcup_{3\le i\le j} fv(N_i) \subseteq dom(A^8)$ and $\bigcup_{3\le i\le j} fn(N_i)\cap bn(A^8) = \emptyset$. There are two subcases :

- If $A \longrightarrow A'$ and $k < n$, then it must be the case that $A \equiv C\left[\overline{c}_{BD}\langle \Pi_1(x_k)\rangle.B^5_{k+1,n}|c_{BD}(d_k).D^1_{k+1,n}\right]\Sigma_L$ and $A' \equiv C\left[B^5_{k+1,n}|D^1_{k+1,n}\right]\Sigma_L$ where $C\,[\_] = A_8[\_|\Lambda_{n+1}]$. It follows from $B \equiv C[\overline{c}_{BD}\langle \Pi_1(x_k)\rangle.B^5_{k+1,n}|c_{BD}(d_k).\overline{D}^1_{k+1,n}]\Sigma_R$ that $B \longrightarrow B'$ where $B' = A_8\left[B^5_{k+1,n}|\overline{D}^1_{k+1,n}|\Lambda_{n+1}\right]\Sigma_R$. Since $A' \equiv A_8\left[B^5_{k+1,n}|D^1_{k+1,n}|\Lambda_{n+1}\right]\Sigma_L\ \mathcal{R}\ B'$, we derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

- If $A \longrightarrow A'$ and $k = n$, then it must be the case that $A \equiv C\left[\overline{c}_{BD}\langle \Pi_1(x_n)\rangle|c_{BD}(d_n).D^2_{1,n}\right]\Sigma_L$ and $A' \equiv C\left[D^2_{1,n}\right]\Sigma_L$ where $C\,[\_] = A_8[\_|\Lambda_{n+1}]$. It follows from $B \equiv C[\overline{c}_{BD}\langle \Pi_1(x_n)\rangle|c_{BD}(d_n).\overline{D}^2_{1,n}]\Sigma_R$ that $B \longrightarrow B'$ where $B' = A_8\left[\overline{D}^2_{1,n}|\Lambda_{n+1}\right]\Sigma_R$. Since $A' \equiv A_8\left[D^2_{1,n}|\Lambda_{n+1}\right]\Sigma_L\ \mathcal{R}\ B'$, we derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

Labelled Reductions : We must show for all extended processes A and B, where $A\ \mathcal{R}\ B$, that if $A \xrightarrow{\alpha} A'$ for some A', then $B \longrightarrow^* \xrightarrow{\alpha} \longrightarrow^* B'$ and $A'\ \mathcal{R}\ B'$ for some B'. We observe that if $A\ \mathcal{R}\ B$ by an other relation than (2), (8), (11), (15), (21), (24), (27), (33), (35), (38), (39) or (40) then there is no extended process A' such that $A \xrightarrow{\alpha} A'$. We proceed by case analysis on the remaining cases.

(2) We have $A \equiv A_1\left[V^2_{1,1}|V^1_{2,2}|B^{1.1}_{1,n}|R^1_{1,n}|D^1_{1,n}\right]\sigma$ and $B \equiv A_1\left[V^2_{1,2}|V^1_{2,1}|B^{1.1}_{1,n}|R^1_{1,n}|\overline{D}^1_{1,n}\right]\tau$ If $A \xrightarrow{\alpha} A'$ such that $fv(\alpha) \subseteq dom(A)$ and $bn(\alpha)\cap bn(B) = \emptyset$, then it must be the case that $A \equiv A_1\left[V^2_{1,1}|V^1_{2,2}|\overline{c}_{out}\langle x_1\rangle.B^{1.2}_{1,n}|R^1_{1,n}|D^1_{1,n}\right]\sigma$ and $A' \equiv A_1\left[V^2_{1,1}|V^1_{2,2}|B^{1.2}_{1,n}|R^1_{1,n}|D^1_{1,n}\right]\sigma$ where $\alpha = \nu b_1.\overline{c}_{out}\langle b_1\rangle$ and $b_1 \notin fv(\Gamma)$ where $\Gamma = \{{}^{\mathsf{pk}(a_1)}/_{g_1},\ {}^{\mathsf{pk}(a_2)}/_{g_2},\ {}^{\mathsf{pk}(a_3)}/_{g_3},\ {}^{\mathsf{vk}(id_1)}/_{idp_1},\ldots,{}^{\mathsf{vk}(id_n)}/_{idp_n},\ {}^{\mathsf{vk}(id_R)}/_{idp_R}\}$. It follows from $B \equiv A_1\left[V^2_{1,2}|V^1_{2,1}|\overline{c}_{out}\langle x_1\rangle.B^{1.2}_{1,n}|R^1_{1,n}|\overline{D}^1_{1,n}\right]\tau$ that $B \xrightarrow{\alpha} B'$ where $B' \equiv A_1\left[V^2_{1,2}|V^1_{2,1}|B^{1.2}_{1,n}|R^1_{1,n}|\overline{D}^1_{1,n}\right]\tau$.

We have $A_1 \left[V^2_{1,1}|V^1_{2,2}|B^{1.2}_{1,n}|R^1_{1,n}|D^1_{1,n}\right] \sigma \mathcal{R} B'$, and derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(8) We have $A \equiv A_2 \left[V^2_{1,1}|V^1_{2,2}|B^{3.1}_{1,n}\{^{sig^R_1}/_{q_1}\}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma$ and $B \equiv A_2 \left[V^2_{1,2}|V^1_{2,1}|B^{3.1}_{1,n}\{^{sig^R_1}/_{q_1}\}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right] \tau$ If $A \xrightarrow{\alpha} A'$ such that $fv(\alpha) \subseteq dom(A)$ and $bn(\alpha) \cap bn(B) = \emptyset$, then it must be the case that $A \equiv A_2 \left[V^2_{1,1}|V^1_{2,2}|\overline{c}_{out}\langle q_1\rangle.B^{3.2}_{1,n}\{^{sig^R_1}/_{q_1}\}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma$ and $A' \equiv A_3 \left[V^2_{1,1}|V^1_{2,2}|B^{3.2}_{1,n}\{^{sig^R_1}/_{q_1}\}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma$ where $\alpha = \nu z_1.\overline{c}_{out}\langle z_1\rangle$ and $z_1 \notin \{b_1\} \cup fv(\Gamma)$. It follows from $B \equiv A_2 \left[V^2_{1,2}|V^1_{2,1}|\overline{c}_{out}\langle q_1\rangle.B^{3.2}_{1,n}\{^{sig^R_1}/_{q_1}\}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right] \tau$ that $B \xrightarrow{\alpha} B'$ where $B' \equiv A_3 \left[V^2_{1,2}|V^1_{2,1}|B^{3.2}_{1,n}\{^{sig^R_1}/_{q_1}\}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right] \tau$. We have $A_3 \left[V^2_{1,1}|V^1_{2,2}|B^{3.2}_{1,n}\{^{sig^R_1}/_{q_1}\}|R^3_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma \mathcal{R} B'$, and derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(11) We have $A \equiv A_3 \left[V^3_{1,1}|V^1_{2,2}|B^4_{1,n}|R^4_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma$ and $B \equiv A_3 \left[V^3_{1,2}|V^1_{2,1}|B^4_{1,n}|R^4_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right] \tau$ If $A \xrightarrow{\alpha} A'$ such that $fv(\alpha) \subseteq dom(A)$ and $bn(\alpha) \cap bn(B) = \emptyset$, then it must be the case that $A \equiv A_3 \left[V^3_{1,1}|V^1_{2,2}|B^4_{1,n}|\overline{c}_{out}\langle r_1\rangle.R^{4.1}_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma$ and $A' \equiv A_4 \left[V^3_{1,1}|V^1_{2,2}|B^4_{1,n}|R^{4.1}_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma$ where $\alpha = \nu y_1.\overline{c}_{out}\langle y_1\rangle$ and $y_1 \notin \{b_1, z_1\} \cup fv(\Gamma)$. It follows from $B \equiv A_3 \left[V^3_{1,2}|V^1_{2,1}|B^4_{1,n}|\overline{c}_{out}\langle r_1\rangle.R^{4.1}_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right] \tau$ that $B \xrightarrow{\alpha} B'$ where $B' \equiv A_4 \left[V^3_{1,2}|V^1_{2,1}|B^4_{1,n}|R^{4.1}_{1,n}\{^{ballot'_1}/_{p_1}\}|\overline{D}^1_{1,n}\right] \tau$. We have $A_4 \left[V^3_{1,1}|V^1_{2,2}|B^4_{1,n}|R^{4.1}_{1,n}\{^{ballot'_1}/_{p_1}\}|D^1_{1,n}\right] \sigma \mathcal{R} B'$, and derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(15) We have $A \equiv A_5 \left[V^2_{2,2}|B^{1.1}_{2,n}|R^1_{2,n}|D^1_{1,n}\right] \Sigma_L$ and $B \equiv A_5 \left[V^2_{2,1}|B^{1.1}_{2,n}|R^1_{2,n}|\overline{D}^1_{1,n}\right] \Sigma_R$ If $A \xrightarrow{\alpha} A'$ such that $fv(\alpha) \subseteq dom(A)$ and $bn(\alpha) \cap bn(B) = \emptyset$, then it must be the case that $A \equiv A_5 \left[V^2_{2,2}|\overline{c}_{out}\langle x_2\rangle.B^{1.2}_{2,n}|R^1_{2,n}|D^1_{1,n}\right] \Sigma_L$ and $A' \equiv A_6 \left[V^2_{2,2}|B^{1.2}_{2,n}|R^1_{2,n}|D^1_{1,n}\right] \Sigma_L$ where $\alpha = \nu b_2.\overline{c}_{out}\langle b_2\rangle$ and $b_2 \notin \{b_1, z_1, y_1\} \cup fv(\Gamma)$. It follows from $B \equiv A_5 \left[V^2_{2,1}|\overline{c}_{out}\langle x_2\rangle.B^{1.2}_{2,n}|R^1_{2,n}|\overline{D}^1_{1,n}\right] \Sigma_R$ that $B \xrightarrow{\alpha} B'$ where $B' \equiv A_6 \left[V^2_{2,1}|B^{1.2}_{2,n}|R^1_{2,n}|\overline{D}^1_{1,n}\right] \Sigma_R$. We have $A_6 \left[V^2_{2,2}|B^{1.2}_{2,n}|R^1_{2,n}|D^1_{1,n}\right] \Sigma_L \mathcal{R} B'$, and derive $A' \mathcal{R} B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(21) We have $A \equiv A_6 \left[V^2_{2,2}|B^{3.1}_{2,n}\{^{sig^R_2}/_{q_2}\}|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right] \Sigma_L$ and $B \equiv A_6 \left[V^2_{2,1}|B^{3.1}_{2,n}\{^{sig^R_2}/_{q_2}\}|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right] \Sigma_R$ If $A \xrightarrow{\alpha} A'$ such that $fv(\alpha) \subseteq dom(A)$ and $bn(\alpha) \cap bn(B) = \emptyset$, then it must be the case that $A \equiv A_6 \left[V^2_{2,2}|\overline{c}_{out}\langle q_2\rangle.B^{3.2}_{2,n}\{^{sig^R_2}/_{q_2}\}|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right] \Sigma_L$ and $A' \equiv A_7 \left[V^2_{2,2}|B^{3.2}_{2,n}\{^{sig^R_2}/_{q_2}\}|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right] \Sigma_L$ where $\alpha = \nu z_2.\overline{c}_{out}\langle z_2\rangle$ and $z_2 \notin \{b_1, z_1, y_1, b_2\} \cup fv(\Gamma)$. It follows from $B \equiv A_6 \left[V^2_{2,1}|\overline{c}_{out}\langle q_2\rangle.B^{3.2}_{2,n}\{^{sig^R_2}/_{q_2}\}|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right] \Sigma_R$ that $B \xrightarrow{\alpha} B'$ where $B' \equiv A_7 \left[V^2_{2,1}|B^{3.2}_{2,n}\{^{sig^R_2}/_{q_2}\}|R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right] \Sigma_R$. We have $A_7 \left[V^2_{2,2}|B^{3.2}_{2,n}\{^{sig^R_2}/_{q_2}\}|\right.$

$R^3_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right]\Sigma_L\ \mathcal{R}\ B'$, and derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(24) We have $A \equiv A_7\left[V^3_{2,2}|B^4_{2,n}|R^4_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right]\Sigma_L$ and $B \equiv A_7\left[V^3_{2,1}|B^4_{2,n}|\right.$ $\left.R^4_{2,n}\{^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right]\Sigma_R$ If $A \xrightarrow{\alpha} A'$ such that $fv(\alpha) \subseteq dom(A)$ and $bn(\alpha) \cap bn(B) = \emptyset$, then it must be the case that $A \equiv A_7\left[V^3_{2,2}|B^4_{2,n}|\overline{c}_{out}\langle r_2\rangle.R^{4.1}_{2,n}\{^{ballot'_2}/_{p_2}\}|\right.$ $\left.D^1_{1,n}\right]\Sigma_L$ and $A' \equiv A_8\left[V^3_{2,2}|B^4_{2,n}|R^{4.1}_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right]\Sigma_L$ where $\alpha = \nu y_2.\overline{c}_{out}\langle y_2\rangle$ and $y_2 \notin \{b_1, z_1, y_1, b_2, z_2\} \cup fv(\Gamma)$. It follows from $B \equiv A_7\left[V^3_{2,1}|B^4_{2,n}|\overline{c}_{out}\langle r_2\rangle.\right.$ $\left.R^{4.1}_{2,n}\{^{ballot'_2}/_{p_2}\}|\overline{D}^1_{1,n}\right]\Sigma_R$ that $B \xrightarrow{\alpha} B'$ where $B' \equiv A_8\left[V^3_{2,1}|B^4_{2,n}|R^{4.1}_{2,n}\{^{ballot'_2}/_{p_2}\}|\right.$ $\left.\overline{D}^1_{1,n}\right]\Sigma_R$. We have $A_8\left[V^3_{2,2}|B^4_{2,n}|R^{4.1}_{2,n}\{^{ballot'_2}/_{p_2}\}|D^1_{1,n}\right]\Sigma_R\ \mathcal{R}\ B'$, and derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(27) We have $A \equiv A_8\left[B^1_{j,n}|R^1_{j,n}|D^1_{1,n}|\Lambda_j\right]\Sigma_L$ and $B \equiv A_8\left[B^1_{j,n}|R^1_{j,n}|\overline{D}^1_{1,n}|\Lambda_j\right]\Sigma_R$ for some $j \in \{3, \ldots, n\}$, $N_3, \ldots, N_j$ such that $N_k\theta_{k-1}\sigma_N^{k-1}\Sigma_L$ are id_k-valid ballots for $k = 3 \ldots j$ and $\bigcup_{3\leq i\leq j} fv(N_i) \subseteq dom(A^8)$ and $\bigcup_{3\leq i\leq j} fn(N_i) \cap bn(A^8) = \emptyset$. If $A \xrightarrow{\alpha} A'$ such that $fv(\alpha) \subseteq dom(A)$ and $bn(\alpha) \cap bn(B) = \emptyset$, then it must be the case that $A \equiv A_8\left[c_j(x_j).B^{1.2}_{j,n}|R^1_{j,n}|D^1_{1,n}|\Lambda_j\right]\Sigma_L$ and $A' \equiv A_8\left[B^{1.2}_{j,n}\{^{M_j}/_{x_j}\}|R^1_{j,n}|D^1_{1,n}|\Lambda_j\right]\Sigma_L$ where $\alpha = c_j(M_j)$ for some term M_j such that $fn(\alpha) \cap bn(A^8) = \emptyset$. It follows from $B \equiv A_8\left[c_j(x_j).B^{1.2}_{j,n}|R^1_{j,n}|\overline{D}^1_{1,n}|\Lambda_j\right]\Sigma_R$ that $B \xrightarrow{\alpha} B'$ where $B' \equiv A_8\left[B^{1.2}_{j,n}\{^{M_j}/_{x_j}\}|R^1_{j,n}|\overline{D}^1_{1,n}|\Lambda_j\right]\Sigma_R$. We have $A_8\left[B^{1.2}_{j,n}\{^{M_j}/_{x_j}\}|R^1_{j,n}|D^1_{1,n}|\Lambda_j\right]\Sigma_L\ \mathcal{R}\ B'$, and derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(33) We have $A \equiv A_8\left[B^{3.2}_{j,n}\{^{sig^R_j}/_{q_j}\}|R^3_{j,n}\{^{ballot'_j}/_{p_j}\}|D^1_{1,n}|\Lambda'_j\right]\Sigma_L$ and $B \equiv A_8\left[B^{3.2}_{j,n}\{^{sig^R_j}/_{q_j}\}|\right.$ $\left.R^3_{j,n}\{^{ballot'_j}/_{p_j}\}|\overline{D}^1_{1,n}|\Lambda'_j\right]\Sigma_R$ for some $j \in \{3, \ldots, n\}$, $N_3, \ldots, N_j$ such that $N_k\theta_{k-1}$ $\sigma_N^{k-1}\Sigma_L$ are id_k-valid ballots for $k = 3 \ldots j$ and $\bigcup_{3\leq i\leq j} fv(N_i) \subseteq dom(A^8)$ and $\bigcup_{3\leq i\leq j} fn(N_i) \cap bn(A^8) = \emptyset$. If $A \xrightarrow{\alpha} A'$ such that $fv(\alpha) \subseteq dom(A)$ and $bn(\alpha) \cap bn(B) = \emptyset$, then it must be the case that $A \equiv A_8\left[\overline{c}_{out}\langle q_j\rangle.B^{3.3}_{j,n}|R^3_{j,n}\{^{ballot'_j}/_{p_j}\}|D^1_{1,n}|\right.$ $\left.\Lambda'_j\right]\Sigma_L$ and $A' \equiv A_8\left[B^{3.3}_{j,n}|R^3_{j,n}\{^{ballot'_j}/_{p_j}\}|D^1_{1,n}|\Lambda''_j\right]\Sigma_L$ where $\alpha = \nu z_j.\overline{c}_{out}\langle z_j\rangle$ and $z_j \notin \{b_1, b_2, z_1, y_1, \ldots z_{j-1}, y_{j-1}\} \cup fv(\Gamma)$. It follows from $B \equiv A_8\left[\overline{c}_{out}\langle q_j\rangle.B^{3.3}_{j,n}|\right.$ $\left.R^3_{j,n}\{^{ballot'_j}/_{p_j}\}|\overline{D}^1_{1,n}|\Lambda'_j\right]\Sigma_R$ that $B \xrightarrow{\alpha} B'$ where $B' \equiv A_8\left[B^{3.3}_{j,n}|R^3_{j,n}\{^{ballot'_j}/_{p_j}\}|\overline{D}^1_{1,n}|\right.$ $\left.\Lambda''_j\right]\Sigma_R$. We have $A_8\left[B^{3.3}_{j,n}|R^3_{j,n}\{^{ballot'_j}/_{p_j}\}|D^1_{1,n}|\Lambda''_j\right]\Sigma_L\ \mathcal{R}\ B'$, and derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(35) We have $A \equiv A_8\left[B^4_{j,n}|R^{4.1}_{j,n}\{^{ballot'_j}/_{p_j}\}|D^1_{1,n}|\Lambda''_j\right]\Sigma_L$ and $B \equiv A_8\left[B^4_{j,n}|R^{4.1}_{j,n}\{^{ballot'_j}/_{p_j}\}|\right.$ $\left.\overline{D}^1_{1,n}|\Lambda''_j\right]\Sigma_R$ for some $j \in \{3, \ldots, n\}$, $N_3, \ldots, N_j$ such that $N_k\theta_{k-1}\sigma_N^{k-1}\Sigma_L$ are id_k-valid ballots for $k = 3 \ldots j$ and $\bigcup_{3\leq i\leq j} fv(N_i) \subseteq dom(A^8)$ and $\bigcup_{3\leq i\leq j} fn(N_i) \cap$ $bn(A^8) = \emptyset$. If $A \xrightarrow{\alpha} A'$ such that $fv(\alpha) \subseteq dom(A)$ and $bn(\alpha) \cap bn(B) = \emptyset$, then it must be the case that $A \equiv A_8\left[B^4_{j,n}|\overline{c}_{out}\langle r_j\rangle.R^{4.2}_{j,n}|D^1_{1,n}|\Lambda''_j\right]\Sigma_L$ and $A' \equiv$

$A_8\left[B^4_{j,n}|R^{4.2}_{j,n}|D^1_{1,n}|\Lambda_{j+1}\right]\Sigma_L$ where $\alpha = \nu y_j.\overline{c}_{out}\langle y_j\rangle$ and $y_j \notin \{b_1, b_2, z_1, y_1, \ldots z_{j-1}, y_{j-1}, z_j\} \cup fv(\Gamma)$. It follows from $B \equiv A_8\left[B^4_{j,n}|\overline{c}_{out}\langle r_j\rangle.R^{4.2}_{j,n}|\overline{D}^1_{1,n}|\Lambda''_j\right]\Sigma_R$ that $B \xrightarrow{\alpha} B'$ where $B' \equiv A_8\left[B^4_{j,n}|R^{4.2}_{j,n}|\overline{D}^1_{1,n}|\Lambda_{j+1}\right]\Sigma_R$. We have $A_8\left[B^4_{j,n}|R^{4.2}_{j,n}|D^1_{1,n}|\Lambda_{j+1}\right]\Sigma_R\ \mathcal{R}\ B'$, and derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(38) We have $A \equiv A_8\left[D^2_{1,n}\{^{\Pi_1(x_1)}/_{d_1}\}|\Lambda_{n+1}\right]\Sigma_L$ and $B \equiv A_8\left[\overline{D}^2_{1,n}\{^{\Pi_1(x_2)}/_{d_2}\}|\Lambda_{n+1}\right]\Sigma_R$ for some $N_3, \ldots, N_n$ such that $N_k\theta_{k-1}\sigma_N^{k-1}\Sigma_L$ are id_k-valid ballots for $k = 3 \ldots n$ and $\bigcup_{3\le i\le n} fv(N_i) \subseteq dom(A^8)$ and $\bigcup_{3\le i\le n} fn(N_i) \cap bn(A^8) = \emptyset$. If $A \xrightarrow{\alpha} A'$ such that $fv(\alpha) \subseteq dom(A)$ and $bn(\alpha) \cap bn(B) = \emptyset$, then it must be the case that $A \equiv A_8\left[\overline{c}_{out}\langle dec_1\rangle.D^2_{2,n}\{^{\Pi_1(x_2)}/_{d_2}\}|\Lambda_{n+1}\right]\Sigma_L$ and $A' \equiv A_{9,1}\left[D^2_{2,n}\{^{\Pi_1(x_2)}/_{d_2}\}|\Lambda_{j+1}\right]\Sigma_L$ where $\alpha = \nu\ result_1.\overline{c}_{out}\langle result_1\rangle$ and $result_1 \notin \{b_1, b_2, z_1, y_1, \ldots z_n, y_n\} \cup fv(\Gamma)$. It follows from $B \equiv A_8\left[\overline{c}_{out}\langle dec_2\rangle.\overline{D}^2_{2,n}\{^{\Pi_1(x_1)}/_{d_1}\}|\Lambda_{n+1}\right]\Sigma_R$ that $B \xrightarrow{\alpha} B'$ where $B' \equiv \overline{A}_{9,1}\left[\overline{D}^2_{2,n}\{^{\Pi_1(x_1)}/_{d_1}\}|\Lambda_{n+1}\right]\Sigma_R$. We have $A_{9,1}\left[D^2_{2,n}\{^{\Pi_1(x_2)}/_{d_2}\}|\Lambda_{n+1}\right]\Sigma_R\ \mathcal{R}\ B'$, and derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(39) We have $A \equiv A_8\left[D^2_{2,n}\{^{\Pi_1(x_2)}/_{d_2}\}|\Lambda_{n+1}\right]\Sigma_L$ and $B \equiv A_8\left[\overline{D}^2_{2,n}\{^{\Pi_1(x_1)}/_{d_1}\}|\Lambda_{n+1}\right]\Sigma_R$ for some $N_3, \ldots, N_n$ such that $N_k\theta_{k-1}\sigma_N^{k-1}\Sigma_L$ are id_k-valid ballots for $k = 3 \ldots n$ and $\bigcup_{3\le i\le n} fv(N_i) \subseteq dom(A^8)$ and $\bigcup_{3\le i\le n} fn(N_i) \cap bn(A^8) = \emptyset$. If $A \xrightarrow{\alpha} A'$ such that $fv(\alpha) \subseteq dom(A)$ and $bn(\alpha) \cap bn(B) = \emptyset$, then it must be the case that $A \equiv A_8\left[\overline{c}_{out}\langle dec_2\rangle.D^2_{3,n}\{^{\Pi_1(x_3)}/_{d_3}\}|\Lambda_{n+1}\right]\Sigma_L$ and $A' \equiv A_{9,1}\left[D^2_{3,n}\{^{\Pi_1(x_3)}/_{d_3}\}|\Lambda_{j+1}\right]\Sigma_L$ where $\alpha = \nu\ result_2.\overline{c}_{out}\langle result_2\rangle$ and $result_2 \notin \{b_1, b_2, z_1, y_1, \ldots z_n, y_n, result_1\} \cup fv(\Gamma)$. It follows from $B \equiv A_8\left[\overline{c}_{out}\langle dec_2\rangle.\overline{D}^2_{3,n}\{^{\Pi_1(x_3)}/_{d_3}\}|\Lambda_{n+1}\right]\Sigma_R$ that $B \xrightarrow{\alpha} B'$ where $B' \equiv \overline{A}_{9,1}\left[\overline{D}^2_{3,n}\{^{\Pi_1(x_3)}/_{d_3}\}|\Lambda_{n+1}\right]\Sigma_R$. We have $A_{9,1}\left[D^2_{3,n}\{^{\Pi_1(x_3)}/_{d_3}\}|\Lambda_{n+1}\right]\Sigma_R\ \mathcal{R}\ B'$, and derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

(40) We have $A \equiv A_8\left[D^2_{j,n}\{^{\Pi_1(x_j)}/_{d_j}\}|\Lambda_{n+1}\right]\Sigma_L$ and $B \equiv A_8\left[\overline{D}^2_{j,n}\{^{\Pi_1(x_j)}/_{d_j}\}|\Lambda_{n+1}\right]\Sigma_R$ for some $N_3, \ldots, N_n$ such that $N_k\theta_{k-1}\sigma_N^{k-1}\Sigma_L$ are id_k-valid ballots for $k = 3 \ldots n$ and $\bigcup_{3\le i\le n} fv(N_i) \subseteq dom(A^8)$ and $\bigcup_{3\le i\le n} fn(N_i) \cap bn(A^8) = \emptyset$. If $A \xrightarrow{\alpha} A'$ such that $fv(\alpha) \subseteq dom(A)$ and $bn(\alpha) \cap bn(B) = \emptyset$, then there are two cases :

- If $3 \le j < n$. $A \equiv A_{9,j-1}\left[\overline{c}_{out}\langle dec_j\rangle.D^2_{j+1,n}\{^{\Pi_1(x_{j+1})}/_{d_{j+1}}\}|\Lambda_{n+1}\right]\Sigma_L$ and $A' \equiv A_{9,j}\left[D^2_{j+1,n}\{^{\Pi_1(x_{j+1})}/_{d_{j+1}}\}|\Lambda_{n+1}\right]\Sigma_L$ where $\alpha = \nu\ result_j.\overline{c}_{out}\langle result_j\rangle$ and $result_j \notin \{b_1, b_2, z_1, y_1, \ldots z_n, y_n, result_1, \ldots, result_{j-1}\} \cup fv(\Gamma)$. It follows from $B \equiv \overline{A}_{9,j-1}[\ \overline{c}_{out}\langle dec_j\rangle.\overline{D}^2_{j+1,n}\{^{\Pi_1(x_{j+1})}/_{d_{j+1}}\}|\Lambda_{n+1}\right]\Sigma_R$ that $B \xrightarrow{\alpha} B'$ where $B' \equiv \overline{A}_{9,j}\left[\overline{D}^2_{j+1,n}|\Lambda_{n+1}\right]\Sigma_R$. We have $A_{9,j}\left[D^2_{j+1,n}|\Lambda_{n+1}\right]\Sigma_R\ \mathcal{R}\ B'$, and derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

- If $j = n$. $A \equiv A_{9,n-1}\left[\overline{c}_{out}\langle dec_n\rangle|\Lambda_{n+1}\right]\Sigma_L$ and $A' \equiv A_{9,n}\left[\Lambda_{n+1}\right]\Sigma_L$ where $\alpha = \nu\ result_n.\overline{c}_{out}\langle result_n\rangle$ and $result_n \notin \{b_1, b_2, z_1, y_1, \ldots z_n, y_n, result_1, \ldots, result_{n-1}\} \cup$

$fv(\Gamma)$. It follows from $B \equiv \overline{A}_{9,n-1}\,[\overline{c}_{out}\langle dec_n\rangle|\Lambda_{n+1}]\,\Sigma_R$ that $B \xrightarrow{\alpha} B'$ where $B' \equiv \overline{A}_{9,n}\,[\Lambda_{n+1}]\,\Sigma_R$. We have $A_{9,n}\,[\Lambda_{n+1}]\,\Sigma_R\ \mathcal{R}\ B'$, and derive $A'\ \mathcal{R}\ B'$ by the closure of $\mathcal{R}$ under structural equivalence.

Bibliographie

[AB03] Martin Abadi and Bruno Blanchet. Computer-assisted verification of a protocol for certified email. In *SAS*, pages 316–335, 2003.

[ABB+05] Alessandro Armando, David A. Basin, Yohan Boichut, Yannick Chevalier, Luca Compagna, Jorge Cuéllar, Paul Hankes Drielsma, Pierre-Cyrille Héam, Olga Kouchnarenko, Jacopo Mantovani, Sebastian Mödersheim, David von Oheimb, Michaël Rusinowitch, Judson Santiago, Mathieu Turuani, Luca Viganò, and Laurent Vigneron. The avispa tool for the automated validation of internet security protocols and applications. In *CAV*, pages 281–285, 2005.

[ABF04] Martín Abadi, Bruno Blanchet, and Cédric Fournet. Just fast keying in the pi calculus. pages 340–354. Springer, 2004.

[AC04] Martin Abadi and Véronique Cortier. Deciding knowledge in security protocols under equational theories. In Josep Diaz, Juhani Karhumäki, Arto Lepistö, and Donald Theodore Sannella, editors, *The 31st International Colloquium on Automata, Languages and Programming - ICALP'2004*, volume 3142 of *Lecture Notes in Computer Science*, pages 148–164, Turku, Finland, 2004. Springer. Colloque avec actes et comité de lecture. nationale.

[AF01] Martin Abadi and Cédric Fournet. Mobile values, new names, and secure communication. *SIGPLAN Not.*, 36 :104–115, January 2001.

[AF03] Martin Abadi and Cédric Fournet. Hiding names : Private authentication in the applied pi calculus. In *In Software Security – Theories and Systems. Mext-NSF-JSPS International Symposium (ISSSâ02*, pages 317–338. Springer-Verlag, 2003.

[BBC09] Mouhebeddine Berrima, Narjes Ben Rajeb, and Véronique Cortier. Deciding knowledge in security protocols under some e-voting theories. Research Report RR-6903, INRIA, 2009.

[Bla01] Bruno Blanchet. An efficient cryptographic protocol verifier based on prolog rules. In *In 14th IEEE Computer Security Foundations Workshop (CSFW-14)*, pages 82–96. IEEE Computer Society Press, 2001.

[BN98] Franz Baader and Tobias Nipkow. *Term Rewriting and All That.* Cambridge University Press, 1998.

[CJ97] Hubert Comon and Jean-Pierre Jouannaud. *Les termes en logique et en programmation. Version préliminaire*, 1997.

[Cor11] Véronique Cortier. *Théorie de la Sécurité*, 2011.

[CS11] Véronique Cortier and Ben Smyth. Attacking and fixing helios : An analysis of ballot secrecy. In *Proceedings of the 24th IEEE Computer Security Foundations Symposium (CSF'11), Cernay, France.*, 2011.

[DKR09] Stéphanie Delaune, Steve Kremer, and Mark Ryan. Verifying privacy-type properties of electronic voting protocols. *Journal of Computer Security*, 17 :435–487, 2009.

[DLMS99] N. Durgin, P. Lincoln, J. Mitchell, and A. Scedrov. Undecidability of bounded security protocols. 1999.

[Gjo10] Kristian Gjosteen. Analysis of an internet voting protocol. Cryptology ePrint Archive, Report 2010/380, 2010.

[Hoa78] C. A. R. Hoare. Communicating sequential processes. *Commun. ACM*, 21 :666–677, August 1978.

[Low95] Gavin Lowe. An attack on the needham-schroeder public-key authentication protocol. *Inf. Process. Lett.*, 56 :131–133, November 1995.

[Mil82] R. Milner. *A Calculus of Communicating Systems.* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1982.

[MPW92] Robin Milner, Joachim Parrow, and David Walker. A calculus of mobile processes, parts i and ii. *Inf. Comput.*, 100 :1–77, September 1992.

[NS78] Roger M. Needham and Michael D. Schroeder. Using encryption for authentication in large networks of computers. *Commun. ACM*, 21 :993–999, December 1978.

[RT03] Michaël Rusinowitch and Mathieu Turuani. Protocol insecurity with a finite number of sessions, composed keys is np-complete. *Theor. Comput. Sci.*, 1-3(299) :451–475, 2003.

[Sch96] Bruce Schneier. *Applied cryptography (2nd ed.) : protocols, algorithms, and source code in C.* John Wiley & Sons, Inc., New York, NY, USA, 1996.