هفتمین کنفرانس بین المللی انجمن رمز ایران

۲۴ و ۲۵ شهریورماه ۱۳۸۹، دانشگاه صنعتی خواجه نصیرالدین طوسی

!

انجمن رمزایران

دانشگاه صنعتی خواجه نصیرالدین طوسی

تحلیل صوری خودکار یک پروتکل رأی‌گیری الکترونیکی با استفاده از حساب پی کاربردی

حسین پورمرادیان، حمیدرضا محروقی، رسول جلیلی

مرکز امنیت شبکه شریف

دانشکده مهندسی کامپیوتر

دانشگاه صنعتی شریف

{pourmoradian@cert., mahrooghi@ce., jalili@}sharif.edu

چکیده

هزینه‌ی زیاد و تبعات منفی اجرای طرح‌های ناقص و آسیب‌پذیر، باعث شده است که قبل از اجرای هر سیستم حساس و مهمی، همچون سیستمهای رأی‌گیری الکترونیکی، درستی‌یابی عملکرد آن به امری ضروری و پیش‌نیاز تبدیل شود. اثبات‌های شهودی، از درجه‌ی دقت و اطمینان بالای مورد نیاز برخوردار نیستند، بنابراین همواره استفاده از روش‌های صوری برای این‌گونه اثبات‌ها پیشنهاد می‌شود. در این مقاله، با استفاده از حساب پی کاربردی، که یک زبان صوری مبتنی بر جبر پردازه‌ای بوده و توسعه‌ای بر حساب پی می‌باشد، یک مدل صوری برای پروتکل رأی‌گیری الکترونیکی لین و همکارانش ارائه شده است. همچنین برقراری ویژگی امنیتی صحت، که شامل خصوصیات غیر قابل تغییر بودن، عدم استفاده مجدد و مجاز بودن می‌باشد، در این پروتکل با استفاده از برهان‌های مطابقت و به کمک ابزار پروورِیف، وارسی و اثبات شده است.

واژه های! کلیدی

رأی‌گیری الکترونیکی، تحلیل صوری، حساب پی کاربردی، برهان‌های مطابقت، خصوصیت صحت.

۱- مقدمه !

سیستم‌های رأی‌گیری الکترونیکی در چند دهه اخیر بسیار مورد توجه قرار گرفته و در کشورهای مختلف معرفی و به اجرا گذاشته شده‌اند. به همین منظور پروتکل‌های متعددی جهت ارضای نیازهای امنیتی سیستم‌های رأی‌گیری الکترونیکی ارائه شده‌اند که روند ثبت‌نام، رأی‌گیری، ذخیره آراء و شمارش آنها را بصورت خودکار انجام می‌دهند. اگرچه، طراحی پروتکل‌های امنیتی معمولا با خطاهایی همراه است که تشخیص آنها تنها با آزمون امکان‌پذیر نیست و بسیاری از خطاها با حضور مهاجم بدخواه نمایان می‌شوند. روش‌های صوری، یعنی مدل‌کردن سیستم به صورت ریاضی و امکان بررسی و ارزیابی دقیق ویژگی‌های آن، راه حل مناسبی برای توصیف پروتکل‌ها و تحلیل خصوصیات امنیتی آنها هستند. این روش‌ها امکان تحلیل انتزاعی سیستم‌های رأی‌گیری الکترونیکی را بمنظور وارسی خصوصیات امنیتی مورد نظر بوجود می‌آورند. در این مقاله پروتکل رأی‌گیری الکترونیکی لین و همکارانش [12] با استفاده از حساب پی کاربردی[1][1]، که توسعه‌ای بر حساب پی می‌باشد، به صورت صوری توصیف شده است. برای این منظور، توسعه‌ای از این حساب که شامل رویدادها می‌باشد بکار برده شده و از برهان‌های مطابقت[2]، برای توصیف صوری خصوصیت امنیتی صحت[3] استفاده شده است. در ادامه با کمک ابزار پروورِیف[4]،

[1] Applied pi calculus

[2] Correspondence assertions

[3] Soundness

[4] ProVerif

نشست‌های نامحدودی[1] از پروتکل لین مدل شده و برقرای این خصوصیات بصورت خودکار وارسی و اثبات گردیده است.

۱-۱. خصوصیات امنیتی پروتکل‌های رأی‌گیری الکترونیکی

برخی از خصوصیات امنیتی پروتکل‌های رأی‌گیری الکترونیکی به شرح زیر هستند:

غیر قابل تغییر بودن[2]: هیچکس نتواند رأی شخص دیگری را تغییر دهد.

مجاز بودن[3]: فقط رأی‌دهندگان مجاز قادر به رأی دادن باشند.

عدم استفاده مجدد رأی[4]: هر رأی‌دهنده فقط یکبار بتواند رأی بدهد.

انصاف[5]: نتایج میانی انتخابات فاش نشود، زیرا این نتایج بر روی عملکرد رأی‌دهندگان باقی‌مانده تاثیر خواهد داشت.

قابلیت وارسی[6]: بتوان وارسی نمود که رأی هر شخص بدرستی شمارش شده باشد. همچنین خروجی منتشر شده واقعا حاصل جمع تمامی آراء باشد.

حریم خصوصی[7]: هیچکس متوجه نشود که یک رأی دهنده چگونه و به چه کسی رأی داده است.

مجموع سه خصوصیت اول، تحت عنوان خصوصیت صحت قلمداد می‌شود که از پایه‌ای ترین خصوصیات امنیتی مورد انتظار در هر سیستم رأی‌گیری الکترونیکی می‌باشند.

۱-۲. کارهای مرتبط

تا کنون مقالات متعددی بمنظور توصیف صوری پروتکل‌های رأی‌گیری الکترونیکی و وارسی صوری خصوصیات امنیتی آنها ارائه شده‌اند. در ابتدا آقای کرمر و آقای رایان در [11] خصوصیات انصاف، مجاز بودن و حریم خصوصی را در پروتکل فو [10] مورد تحلیل قرار دادند. سپس خانم دیلان و همکارانش در [9,6,8] علاوه بر خصوصیات فوق به بررسی خصوصیات مبتنی بر حریم خصوصی پرداختند. بعد از آن خانم دیلان و همکارانش در [7] روشی را برای وارسی خودکار این خصوصیات ارائه نمودند. پس از آن در [2,5] یک روش تمام خودکار برای وارسی مطابقت در پروتکل‌های امنیتی ارائه گردید. آقای اسمیت و همکارانش نیز در [13] یک روش خودکار برای وارسی خصوصیت قابلیت وارسی انتخابات ارائه نمودند. همچنین آقای بیکز در [3] به صوری‌سازی خصوصیت صحت پرداخت که ما از این روش برای مدل کردن پروتکل مورد بررسی استفاده می‌کنیم.

۱-۳. ساختار مقاله

در بخش ۲ پروتکل لین و همکارانش توصیف شده است. بخش ۳ مروری بر حساب پی کاربردی دارد و در بخش ۴ با بهره‌گیری از این حساب، یک تعریف صوری برای مدل کردن پردازه انتخابات ارائه شده است. در بخش ۵ خصوصیت صحت بصورت صوری توصیف می‌شود. بخش ۶ به ارائه مدل صوری پروتکل لین و همکارانش اختصاص داشته و بخش ۷ به وارسی خصوصیت صحت در این مدل می‌پردازد. در بخش ۸ نیز جمع بندی مطالب بیان شده است.

۲- پروتکل لین و همکارانش

پروتکل لین یک پروتکل رأی‌گیری الکترونیکی مبتنی بر طرح امضای کور[8] می‌باشد. این پروتکل در موارد متعدد مورد تحلیل قرار گرفته و حملات متعددی بر روی خصوصیات امنیتی آن گزارش شده است. ساختار این پروتکل و تعدد مراجع آن و نوع تعامل آن‌ها با یکدیگر، موجب گردیده که به عنوان مطالعه موردی، مرجع مناسبی برای وارسی خصوصیات امنیتی باشد. این پروتکل شامل سه مرحله ثبت‌نام، رأی‌گیری و شمارش آراء بوده و دارای پنج مرجع رأی‌دهنده V، احراز اصالت AS، رأی-گیری VS، شمارش TCS، و صدور گواهی CA می‌باشد. در اینجا به توصیف مراحل مختلف این پروتکل می‌پردازیم:

۲-۱-مرحله اول: ثبت نام و ایجاد تعرفه رأی‌گیری

این مرحله شامل بدست آوردن یک گواهی کلید عمومی و بدست آوردن تعرفه رأی‌گیری می‌شود:

- در ابتدا رأی‌دهنده یک جفت کلید عمومی و خصوصی سیستم رمزنگاری RSA برای خود تولید کرده و کلید عمومی را به همراه شناسه خود برای CA ارسال می‌کند. CA پس از بررسی شناسه رأی‌دهنده و در صورت مجاز بودن وی، یک گواهی برای کلید عمومی او صادر می‌نماید.

- رأی‌دهنده یک عامل کورکننده b برای حفظ گمنامی خود و دو مقدار شبه تصادفی r و k_1 انتخاب می‌کند.

- سپس بر روی دو مقدار r و k_1، توابع $g(r)$ و $g(k_1)$ را اعمال کرده و با عامل b روی آن‌ها امضای کور انجام می‌دهد. پس از آن درخواست تعرفه رأی‌گیری را به فرم $\left\{V, AS, Cert_V, \left\{Blinded\ Request\right\}_{sk_V}\right\}$ ایجاد و برای AS ارسال می‌نماید.

[1] Unbounded sessions
[2] Inalterability
[3] Eligibility
[4] Non-reuseability
[5] Fairness
[6] Verifiability
[7] Privacy
[8] Blind signature

- مرجع AS پس از دریافت درخواست از رأی‌دهنده، صحت امضاء و اعتبار گواهی وی را بررسی نموده و در صورت صحت یک تعرفه رأی‌گیری کورسازی و امضاء شده ایجاد می‌کند. برای این منظور یک مقدار تصادفی k_2 که برای هر رأی‌دهنده یکتا است را انتخاب کرده و آن را به همراه تعرفه رأی‌گیری کورسازی شده و امضا شده به فرم $\left\{k_2,\left\{Blinded\,Request,g\left(k_2\right),AS\right\}_{sk_{AS}}\right\}_{pk_V}$ برای رأی‌دهنده ارسال می‌نماید. همچنین مقدار k_1 را به همراه شناسه رأی‌دهنده در یک پایگاه داده ذخیره می‌کند.
- با دریافت پیغام از AS ، رأی‌دهنده k_2 را بدست می‌آورد. سپس توسط مقادیر k_1 و k_2 کلیدهای عمومی و خصوصی خود را تولید می‌کند. همچنین مقدار امضاء شده توسط AS را آشکارسازی کرده و امضای AS را روی پارامترهای خود و AS بدست می‌آورد. پس از آن رأی خود را با کلید خصوصی خود رمزکرده و تعرفه رأی‌گیری را به فرم زیر ایجاد می‌نماید:

$$\{vote, g(r), publicKey, \{vote\}_{secretKey},\ \{g(r), g(k_1), g(k_2)\}sk_{AS}\}$$

۲-۲- مرحله دوم: رأی‌گیری

این مرحله شامل ارسال تعرفه رأی‌گیری و بررسی صحت آن می‌شود:

- V تعرفه رأی‌گیری تولید شده را از یک کانال گمنام به مرجع رأی‌گیری VS ارسال می‌کند.
- VS صحت امضای AS را بررسی می‌نماید.
- سپس صحت کلید عمومی و توابع $g(r)$، $g(k_1)$ و $g(k_2)$ را مورد وارسی قرار می‌دهد.
- پس از آن رأی رمز شده را با کلید عمومی رمزگشایی کرده و صحت رأی داده شده را بررسی می‌کند.
- درصورت صحت تمامی موارد فوق، VS تعرفه رأی‌گیری معتبر را در پایگاه داده خود ذخیره می‌کند.

۲-۳-مرحله سوم: شمارش آراء

در این مرحله کنترل و شمارش آراء و اعلام نتایج نهایی صورت می‌گیرد:

- VS تعرفه رأی‌گیری معتبر را به مرجع شمارش آراء ارسال می‌نماید.
- مرجع TCS امکان رأی مجدد در تعرفه‌های بدست آمده را بررسی کرده و رأی‌دهندگان متخلف را با کمک AS شناسایی می‌کند.
- سپس آراء بدست آمده را شمارش کرده و نتایج را منتشر می‌کند.

۳- حساب پی کاربردی

حساب پی کاربردی زبانی برای شرح فرایندهای همروند و ارتباط بین آنهاست. این زبان توسعه‌ای از حساب پی[1] بوده و از آن برای تحلیل انواع گوناگونی از پروتکل‌های امنیتی مبتنی بر رمزنگاری استفاده می‌شود. خصوصیات امنیتی پردازه‌های توصیف شده در این حساب را می‌توان هم با استفاده از تکنیکهای اثبات دستی و هم با استفاده از ابزارهای خودکار همانند پروِوریف [4] وارسی نمود.

۳-۱-مفاهیم حساب پی کاربردی

یک پردازه در حساب پی کاربردی از مجموعه‌ای از نام‌ها a,b,c,k,m,n,...، مجموعه‌ای از متغیرها v,x,y,z، و یک امضاء $\sum$ تشکیل شده است. این امضاء شامل مجموعه‌ای از نمادهای تابعی[2] است که برای توصیف عبارات بکار می‌روند و دارای آرگومان می‌باشند. هر نماد تابعی بدون آرگومان یک ثابت نامیده می‌شود. عبارات بصورت نام‌ها، متغیرها و یا نمادهای تابعی که روی عبارات دیگر اعمال شده‌اند توصیف می‌شوند. فرامتغیرها[3] برای توصیف نام‌ها و متغیرها استفاده می‌شوند. از مفهوم تئوری هم‌ارزی[4] E، برای توصیف هم‌ارزی‌های بین عبارات استفاده می‌گردد. نماد $=_E$ رابطه هم‌ارزی است که از E استنتاج می‌شود. دو عبارت فقط وقتی توسط $=_E$ با یکدیگر در ارتباط هستند که این واقعیت از تساوی‌های موجود در E اشتقاق شود.

در این حساب پردازه‌ها شامل دو دسته ساده و توسعه‌یافته می‌باشند. پردازه‌های توسعه‌یافته شامل جایگزینی‌های فعال[5] و محدودیت[6] روی متغیرها می‌شوند. حساب پردازه‌ای که در اینجا بیان می‌شود، توسعه‌ای بر روی حساب پی کاربردی می‌باشد که به منظور اثبات برهان‌های مطابقت[7]، رویداد نیز به آن اضافه شده است [5]. گرامر این حساب در شکل ۱ توصیف شده است. در این گرامر M و N عبارت هستند، n نام بوده، x متغیر و u فرامتغیر است. پردازه 0 هیچ عملی انجام نمی‌دهد. P | Q به معنی ترکیب موازی دو پردازه P و Q است و !P تعداد

[1] Pi calculus
[2] Function symbols
[3] Meta-variables
[4] Equational theory
[5] Active substitution
[6] Restriction
[7] Correnpondence assertions

نامحدودی از کپی‌های پردازه P را بصورت موازی اجرا می‌نماید. پردازه $\nu n.P$ نام جدید و خصوصی n را ایجاد کرده و سپس رفتارش شبیه پردازه P می‌شود. $u(x).P$ یک ورودی از کانال u دریافت کرده و سپس P را با پیغامی که با پارامتر صوری x جایگزین کرده اجرا می‌نماید. $\overline{u}\langle M\rangle.P$ نیز عبارت M را بر نموده و سپس رفتارش شبیه پردازه P می‌شود.

$P,Q :=$	processes
0	null process
$u(x).P$	message input
$\overline{u}\langle M\rangle.P$	message output
$P \mid Q$	parallel composition
$!P$	replication
$\nu n.P$	name restriction
$if\ M = N\ then\ P\ else\ Q$	condition
$A,B :=$	extended processes
P	plain process
$A \mid B$	parallel composition
$\nu n.P$	name restriction
$\nu x.P$	variable restriction
$let\ x = M\ in\ P$	active substitution
$event(M).P$	event

شکل ۱ – گرامر حساب پی کاربردی

برای مجموعه متغیرهای آزاد و مقید و مجموعه نام‌های آزاد و مقید پردازه A به ترتیب *fv*(A) ، *bv*(A) ، *fn*(A) و *bn*(A) را می‌نویسیم. نماد $\tilde{M}$ نشان دهنده دنباله عبارات $M_1,...,M_l$، $\nu\tilde{n}$ نمایانگر دنباله‌ای از محدودیت روی نام به فرم $\nu n_1,...,\nu n_l$ و $\overline{c}\langle\tilde{M}\rangle$ و $c(\tilde{M})$ نیز به ترتیب نشان دهنده ارسال و دریافت یک چندتایی از عبارات می‌باشند. مفهوم جایگزینی فعال $let\ x = M\ in\ P$ را القاء کرده و به معنی جایگزینی متغیر x با عبارت M می‌باشد، و معادل صوری آن برابر با $\nu a.(\overline{a}\langle M\rangle \mid a(x).P)$ است. event(M).P به معنی اجرای رویداد M می‌باشد. رویدادها شامل اعمال[1] مهم انجام شده توسط پروتکل می‌باشند که روی رفتار مدل تاثیر می‌گذارند و در اثبات خصوصیت مورد تحلیل ما دارای اهمیت هستند.

یک پردازه بسته، پردازه‌ای است که تمامی متغیرهای آن یا محدود بوده و یا با یک جایگزینی فعال توصیف شوند. یک زمینه ساده[2]، یک عبارت (بصورت پردازه ساده یا توسعه‌یافته) است که دارای یک حفره می‌باشد. زمینه‌های ترتیبی[3]، زمینه‌های ساده‌ای هستند که شامل انتشار و ترکیب موازی نباشند.

۴-صوری‌سازی پروتکل انتخابات

در مدل صوری ارائه شده، همانند [3,13] یک پردازه انتخابات تعریف می‌شود، بطوریکه در آن تعداد نامحدودی از رأی دهندگان و مجریان درستکار انتخابات بصورت موازی در حال اجرا هستند. در اینجا از مدل‌سازی مجریان فاسد انتخابات خودداری شده و فرض می‌شود که آنها بخشی از محیط فاسد هستند. دو گونه رأی‌دهنده فرض می‌شوند: رأی‌دهندگان درستکار و رأی‌دهندگان فاسد. رأی‌دهندگان درستکار از مرجع توزیع کلید، شناسه خود را دریافت کرده و توسط آن از مرجع صدور گواهی، گواهی خود را دریافت می‌کنند، سپس درست مطابق توصیف پروتکل رفتار کرده و به وظایف خود عمل می‌کنند. رأی‌دهندگان فاسد پس از دریافت گواهی، کلیه اسرار خود را بر روی کانال عمومی منتشر می‌کنند تا مهاجم بتواند بر اساس این اطلاعات حمله‌ای را بر روی پروتکل صورت دهد. بر این اساس پردازه انتخابات بصورت زیر تعریف می‌شود:

تعریف ۱. پردازه انتخابات: یک پردازه انتخابات، یک پردازه ساده بسته به فرم زیر است:

$$EP \equiv \nu\tilde{n}.\left(!V^{hon} \mid !V^{cor} \mid !A_1,...,!A_k \mid !Q \mid R\right)$$

در اینجا پردازه Q وظیفه تخصیص شناسه به رأی‌دهندگان و مجریان انتخابات و همچنین اعلان عمومی شناسه‌ها را بر عهده دارد. نام‌های محدود $\tilde{n}$ شامل رازهایی می‌باشند که بین رأی‌دهندگان و مجریان انتخابات به اشتراک گذاشته شده‌اند (برای مثال کانال‌های خصوصی مشترک). هر رأی‌دهنده درستکار V^{hon}، شناسه خود را از طریق کانال $ch_{idv} \in \tilde{n}$ دریافت می‌کند، سپس ثبت نام کرده و رأی می‌دهد. هر رأی دهنده فاسد V^{cor}، پس از دریافت شناسه و ثبت نام، کلیه اسرار خود را منتشر می‌نماید. توصیف صوری اجزای این پردازه بصورت زیر است:

- دو زمینه ترتیبی V^{reg} و V^{vote} وجود دارند بطوریکه $V^{hon} \equiv ch_{idv}(x_{idv}).V^{reg}\left[ch_v(x_v).\ let\ vote = x_v\ in\ V^{vote}\right]$ ، به این معنی که رأی‌دهنده درستکار شناسه خود را از کانال ch_{idv} دریافت کرده، سپس ثبت نام می‌کند و رأی می‌دهد. در اینجا داریم $x_v \in \tilde{v}$ که $\tilde{v}$ لیست کاندیدهایی است که رأی‌دهندگان مجاز به رأی دادن به آنها می‌باشند.
- پردازه رای‌دهنده فاسد به فرم $V^{cor} \equiv c_{idv}(x_{idv}).V^c$ می‌باشد.
- پردازه توزیع کلید Q، بصورت زیر می‌باشد:

$$Q \equiv (\nu idv.\overline{ch_{idv}}\langle idv\rangle.\overline{ch_{pub}}\langle idv\rangle) \mid (\nu ida_1.ch_{ida_1}\langle ida_1\rangle.\overline{ch_{pub}}\langle ida_1\rangle \mid ... \mid \nu ida_k.ch_{ida_k}\langle ida_k\rangle.\overline{ch_{pub}}\langle ida_k\rangle)$$

[1] Action

[2] Plain context

[3] Sequential context

این پردازه به ازای هر رأی‌دهنده، شناسه او و تمامی مجریانی که با وی در تعامل هستند را تولید و به آن‌ها ارسال می‌نماید.

- یک پردازه تولید رأی وجود دارد بطوریکه $R \equiv !\nu v.\left(\left(!\overline{ch_v}\langle v\rangle\right) \mid \overline{ch_{pub}}(v)\right)$. این پردازه مجموعه‌ای از آراء (لیست کاندیدهای مجاز) را تولید می‌کند و در اختیار رأی‌دهندگان قرار می‌دهد. این لیست همچنین بر روی کانال عمومی ch_{pub} منتشر می‌شود.
- به ازای برخی از مجریان انتخابات که آنها را با $A_1,...,A_k$ نمایش می‌دهیم، یک زمینه C وجود دارد بطوریکه $A_i \equiv C\left[\overline{ch_{vote}}\langle x\rangle.P\right]$. به این معنی که رأی‌های نهایی را منتشر می‌کند.

۵- صوری‌سازی خصوصیت صحت

تعریف غیرصوری این خصوصیت در بخش ۱-۱ گفته شد، هم اکنون به بیان صوری آن می‌پردازیم. بمنظور صوری‌سازی خصوصیت صحت، ما رویدادهای زیر را به پردازه انتخابات اضافه نموده و پردازه انتخابات تکمیلی را تعریف می‌کنیم: پردازه توزیع کلید Q، رویداد $\texttt{newid}(idv)$ را بعد از تخصیص شناسه idv به رأی‌دهنده اجرا می‌کند. رویدادهای $\texttt{startid}(idv)$ و $\texttt{startcorid}(idv)$ به هنگام شروع فاز ثبت نام به ترتیب توسط رأی‌دهنده درستکار و رأی‌دهنده فاسد اجرا می‌شوند. رویداد $\texttt{beginvote}(idv,v)$ نمایانگر شروع فاز رأی‌گیری برای رأی‌دهنده با شناسه idv است که به کاندید v رأی می‌دهد. رویداد $\texttt{endvote}(v)$ نیز با شمارش رأی v توسط مرجع شمارش اجرا می‌شود.

تعریف ۲. پردازه انتخابات تکمیلی: یک پردازه انتخابات تکمیلی، یک پردازه انتخابات EP است که بصورت زیر تکمیل شده است:

- $V^{hon} \equiv ch_{idv}(x_{idv}).\texttt{startid}(x_{idv}).$ $V^{reg}[ch_v(x_v).\ let\ vote = x_v\ in\ \texttt{beginvote}().V^{vote}]$
- $V^{cor} \equiv c_{idv}(x_{idv}).\texttt{startcorid}(idv).V^c$
- $Q \equiv (\nu idv.\texttt{newid}(idv).\overline{ch_{idv}}\langle idv\rangle.\overline{ch_{pub}}\langle idv\rangle) \mid (\nu ida_1.ch_{ida_1}\langle ida_1\rangle.\overline{ch_{pub}}\langle ida_1\rangle \mid ... \mid \nu ida_k.ch_{ida_k}\langle ida_k\rangle.\overline{ch_{pub}}\langle ida_k\rangle)$
- $A_i \equiv C\left[\texttt{endvote}(x).\overline{ch_{vote}}\langle x\rangle.P\right].$

تعریف ۳. صحت: دنباله t صحت را تضمین می‌کند اگر شروط زیر برقرار باشند:

۱. به ازای هر t_1، t_2 و v که $t = t_1 :: \texttt{endvote}(v) :: t_2$، یکی از دو حالت زیر را داشته باشیم:

(الف) $t_1 = t' :: \texttt{startid}(idv) :: t'' :: \texttt{beginvote}(idv,v) :: t'''$

(ب) $t_1 = t' :: \texttt{startcorid}(idv) :: t''$

۲. ، یکی از دو حالت زیر را داشته باشیم:

(الف) $t_1 = t' :: \texttt{startid}(idv) :: t'' :: \texttt{beginvote}(idv,v) :: t'''$

(ب) $t_1 = t' :: \texttt{startcorid}(idv) :: t''$

۳. به ازای هر t_1، t_2 و idv یکی از دو حالت زیر را داشته باشیم:

$t = t_1 :: \texttt{startid}(idv) :: t_2$

$t = t_1 :: \texttt{startcorid}(idv) :: t_2$

و رویدادهای $\texttt{startid}(idv)$ و $\texttt{startcorid}(idv)$ در $t_2 :: t_1$ رخ ندهند.

۴. به ازای هر t_1، t_2 و idv یکی از دو حالت زیر را داشته باشیم:

$t = t_1 :: \texttt{startid}(idv) :: t_2$

$t = t_1 :: \texttt{startcorid}(idv) :: t_2$

و رویداد $\texttt{newid}(idv)$ در t_1 رخ دهد.

نماد :: به عنوان اجرای مجموعه‌ای از اعمال تعریف می‌شود که اهمیتی در وارسی خصوصیت ندارند (همانند نماد τ در حساب پی). می‌گوئیم یک پردازه انتخابات تکمیلی صحت را تضمین می‌کند، اگر و تنها اگر تمامی دنباله‌های آن صحت را تضمین کنند. در توصیف فوق خصوصیت غیر قابل تغییر بودن به این صورت مدل می‌شود که هر رأی شمرده شده، با رأیی که توسط یک رأی‌دهنده داده شده، مطابقت داشته باشد (رأی-دهنده می‌تواند درستکار (شرط ۱-الف) یا فاسد (شرط ۱-ب) باشد). خصوصیت عدم استفاده مجدد به این صورت مدل می‌شود که رویدادهای $\texttt{endvote}(v)$ و $\texttt{beginvote}(idv,v)$ با یکدیگر مطابقت یک‌به‌یک[1] داشته باشند (شرط۲). درنهایت خصوصیت مجاز بودن به این‌صورت مدل می‌شود که فقط رأی دهنده‌ای فاز ثبت نام را آغاز کند که قبلا شناسه idv به او تخصیص داده شده باشد.

۶-صوری‌سازی پروتکل لین در حساب پی کاربردی

برای صوری‌سازی پروتکل لین و همکارانش از حساب پی کاربردی و ابزار پروِریف استفاده شده است. در اینجا نیازی به

[1] Injective

مدل کردن مهاجم بصورت مجزا نخواهیم داشت. ابزار پروِریف یک مهاجم به صورت پیش‌فرض و به فرم دالو-یااو[1] درنظر می‌گیرد که تمامی پیغام‌های ارسال شده بر روی کانال‌های عمومی را شنود می‌کند.

۶-۱-تئوری‌های هم‌ارزی

تئوری‌های هم‌ارزی مدل لین در پردازه ۱ نشان داده شده است.

```
(*Equational Theory*)
equation decrypt(encrypt(x,pk(y)),y) = x.
equation checksign(sign(x,y),pk(y)) = x.
equation dec(enc(x,skey(k1,k2)),
                    pkey(skey(k1,k2))) = x.
equation
unblind(sign((blind((g(r),g(k1)),b),g(k2),i
d)
      ,y),b) =
sign((g(r),g(k1),g(k2),id),y).
equation checkg(g(k1),g(k2),
                    pkey(skey(k1,k2))) =
true.
```

پردازه ۱- تئوری‌های هم‌ارزی

پروتکل لین بصورت انتزاعی مدل‌سازی شده است، به این معنی که از بیان جزئیات پیغام‌ها و محاسباتی که تاثیری در وارسی خصوصیات ندارند صرف نظر می‌شود. توابع encrypt و decrypt نمایانگر رمزنگاری کلید عمومی هستند. تابع sig عمل امضای رقمی را انجام داده و با checksign می‌توان مقدار امضا شده را استخراج نمود. در مدل ارائه شده نوع خاصی از امضای کور مطرح شده است، به اینصورت که اگر مقادیر کورسازی شده‌ای که با پارامترهای دیگر ترکیب شده و بر روی آنها امضای رقمی صورت گرفته است آشکارسازی شوند، می‌توان به امضای رقمی بر روی کلیه پارامترها دست یافت. توابع enc و dec نیز به ترتیب برای رمزگذاری و رمزگشایی با کلید خصوصی و عمومی رأی‌گیری بکار می‌روند. بدلیل اینکه ماهیت این دو تابع با توابع encrypt و decrypt متفاوت است، تحت تئوری‌های تساوی جداگانه توصیف می‌گردند. تابع checkg عمل وارسی صحت کلید عمومی رأی‌گیری را انجام می‌دهد و g نیز نمایانگر تابع یکطرفه بکار رفته در پروتکل است. همچنین بوسیله تابع pk، از روی کلید خصوصی، کلید عمومی بدست می‌آید.

۶-۲-پردازه محیط

در پردازه اصلی (پردازه ۲) کانال‌های خصوصی ایجاد شده و مشخص می‌شود که پردازه‌ها چگونه با یکدیگر ترکیب شوند. اکثر کانال‌های خصوصی تعریف شده بمنظور توزیع کلید هستند. در روش‌های وارسی مدل بدلیل مشکل انفجار فضای حالت[2]، می‌بایست پروتکل را با تعداد پردازه محدود اجرا و وارسی نمود. اما بوسیله پروِریف می‌توان تعداد نسخه‌های نامحدود از هر پردازه را اجرا کرده و تعامل آنها با یکدیگر را نمایش داد. همانطور که ملاحظه می‌شود مدل‌سازی پروتکل لین بر اساس توصیف صوری پردازه انتخابات که در تعریف ۱ مطرح شد انجام می‌شود. بدلیل اینکه پردازه مرجع شمارش آراء در وارسی خصوصیات مورد نظر تاثیری ندارد، از مدل‌سازی آن خودداری شده است. همچنین از مدل‌سازی ذخیره شناسه‌ها توسط AS و ذخیره تعرفه‌ها توسط VS صرف‌نظر شده است. در اینجا از کانال c بعنوان کانال عمومی استفاده شده و کانال‌های voteCh و tallingCh نیز دو کانال آزاد گمنام هستند که مهاجم به آنها دسترسی ندارد.

```
(* Private Channels *)
νskvCh. νskcaCh. νskasCh. νpkasCh1. νpkasCh2.
νpkcaCh. νprivCh. νcertCh. νauthCh. νlistCh.
(* Processes *)
( !Q | !CA | !AS | !VS | !V | !CorruptV | R
)
```

پردازه ۲- پردازه اصلی

۶-۳- همگام‌سازی پردازه‌ها

جهت همگام‌سازی پردازه‌ها در مراحل مختلف پروتکل لین، از عملگر phase در پروِریف استفاده کرده‌ایم. هنگامی که یک پردازه به phase n برسد، منتظر می‌ماند تا کلیه پردازه‌های دیگر که امکان رسیدن به phase n را دارند نیز به این نقطه برسند.

۶-۴-پردازه مرجع توزیع کلید

مدل شامل یک پردازه اختصاصی برای توزیع کلید می‌باشد (پردازه ۳) که یک PKI را مدل می‌کند. این پردازه همچنین رأی‌دهندگان مجاز را ثبت‌نام کرده و کلیدهای عمومی مجریان انتخابات را برای دیگر پردازه‌ها ارسال می‌کند. این کار با استفاده از کانال‌های محدود شده صورت می‌گیرد و به این ترتیب مهاجم امکان ایجاد کلیدهای عمومی جعلی و ارسال آنها را نخواهد داشت. در اینجا تابع idnt از کلید عمومی، شناسه را تولید می‌کند. بر طبق پردازه انتخابات تکمیلی (تعریف ۲)، پس از تخصیص شناسه به رأی‌دهنده رویداد NEWID اجرا می‌شود که از آن بعدا در وارسی خصوصیات مورد تحلیل استفاده شده است.

[1] Dolev-Yao

[2] State space explosion

```
let Q =
(* Construct Private Keys *)
νskv. νskca. νskas.
(* Construct Correspondig Public Keys *)
let (pkv,pkca,pkas) =
                (pk(skv),pk(skca),pk(skas))
in
(* Public Keys on Public Channels *)
out(c,pkca); out(c,pkas); out(c,pkv);
(* Register Legitimate Voters *)
let idv = idnt(pkv) in
event NEWID(idv);
(out(privCh,(n2,nonceCA2,idv,pkv))|
(* Keys on Private Channels *)
out(skvCh,skv)|out(skcaCh,skca)|
out(skasCh,skas)|
out(pkasCh1,pkas)|out(pkasCh2,pkas)|out(pkc
aCh,pkca)).
```

پردازه ۳- پردازه مرجع توزیع کلید

۶-۵-پردازه تولید رأی

پردازه تولید رأی (پردازه ۴) لیست کاندیدهایی که رأی دهندگان مجاز به رأی‌دادن به آنها هستند را تولید کرده، برای هر رأی‌دهنده یک رأی را انتخاب و برای وی ارسال می‌کند. همچنین این لیست را بر روی کانال عمومی منتشر می‌نماید.

```
let    R    =    (!νs.((!out(listCh,s))    |
out(c,s))).
```

پردازه ۴- پردازه تولید رأی

۶-۶-پردازه رای‌دهنده

در پردازه رأی‌دهنده (پردازه ۵)، وی در ابتدا کلید خصوصی خود را دریافت کرده، سپس درخواست صدور گواهی را به مرجع CA ارسال می‌کند. پس از دریافت گواهی، درخواست دیگری به مرجع احراز اصالت ارسال کرده و یک تعرفه رأی‌گیری کور شده و امضا شده را از وی دریافت می‌نماید. با آشکارسازی این پیغام، امضای AS را بر روی پارامترهای مورد نظر بدست آورده، رأی خود را با کلید خصوصی رأی‌گیری امضا می‌کند و تعرفه رأی‌گیری نهایی را شکل می‌دهد. سپس از طریق کانال گمنام *voteCh* آن را برای مرجع رأی‌گیری ارسال می‌نماید. در این پردازه به هنگام شروع فاز ثبت‌نام، رویداد STARTID و به هنگام شروع فاز رأی‌گیری، رویداد BEGINVOTE اجرا می‌شود.

```
let V =
(* Voter Private Key *)
in(skvCh,seckv);
(* Authorities Public Keys *)
in(pkasCh1,pubkas);
let identas = idnt(pubkas) in
(* Voter Certificate *)
let idv = idnt(pk(seckv)) in
out(certCh,(idv,pk(seckv)));
in(certCh,certv);
phase 1;
event STARTID(idv);
νb. νr. νk1.
(* Construct Request for Authentication *)
let blcommit = blind((g(r),g(k1)),b) in
out(authCh,(idv,identas,certv,

sign((blcommit,idv),seckv)));
(* Get AS Signature on Blinded *)
in(authCh,T');
let (k22,assigonb) = decrypt(T',seckv) in
let assig = unblind(assigonb,b) in
(* Voting Public and Private Keys *)
let skeyv = skey(k1,k22) in
let pkeyv = pkey(skeyv) in
(* Construct Ticket *)
in(listCh,vote);
let T =
   (vote,g(r),pkeyv,enc(vote,skeyv),assig)
in
phase 2;
event BEGINVOTE(idv,vote);
out(voteCh,T).
```

پردازه ۵- پردازه رأی‌دهنده (درستکار)

۶-۷-پردازه رأی‌دهنده فاسد

پردازه رأی‌دهنده فاسد (پردازه ۶) در ابتدا همانند رأی‌دهنده درستکار عمل می‌کند، با این تفاوت که پس از دریافت گواهی، کلیه اسرار خود را از طریق کانال عمومی منتشر کرده و در اختیار مهاجم قرار می‌دهد.

```
let CorruptV =
(* Voter Private Key *)
in(skvCh,seckv);
(* Authorities Public Keys *)
in(pkasCh1,pubkas);
let identas = idnt(pubkas) in
(* Voter Certificate *)
let idv = idnt(pk(seckv)) in
out(certCh,(idv,pk(seckv)));
in(certCh,certv);
event STARTCORID(idv);
νb. νr. νk1.
in(listCh,vote);
(*Reveal Secrets*)
out(c,(seckv,idv,certv,vote,b,r,k1)).
```

پردازه ۶- پردازه رأی‌دهنده فاسد

۶-۸-پردازه مرجع صدور گواهی

پردازه مرجع صدور گواهی (پردازه ۷) پس از دریافت کلید عمومی رأی‌دهنده مجاز از مرجع توزیع کلید، درخواست رأی دهنده را مورد بررسی قرار می‌دهد. درصورت مجاز بودن رأی دهنده، یک گواهی برای کلید عمومی وی صادر کرده و برایش ارسال می‌نماید.

```
let CA =
(*** Private Key ***)
in(skcaCh,seckca);
(*** Legitimate Voters ***)
in(privCh,(idvoter,pkvoter));
(*** Construct Certificate ***)
in(certCh,(=idvoter,=pkvoter));
let  cert  =  sign((idvoter,pkvoter),seckca)
in
out(certCh,cert).
```

پردازه ۷- پردازه مرجع صدور گواهی

۶-۹-پردازه مرجع احراز اصالت

پردازه مرجع احراز اصالت (پردازه ۸)، درخواست صدور تعرفه رأی‌گیری را از رای‌دهنده دریافت می‌کند و درصورت تایید اعتبار گواهی وی، مقدار کورسازی شده توسط رأی‌دهنده را به همراه پارامترهای خود امضا کرده و برای او ارسال می‌نماید.

```
let AS =
(* Private Key *)
in(skasCh,seckas);
let idas = idnt(pk(seckas)) in
(* Authorities Public Keys *)
in(pkcaCh,pubkca);
phase 1;
(* Process Authentication Requst *)
in(authCh,(identvoter,=idas,certificate,req
))
let (idvt,pkvt) =
          checksign(certificate,pubkca) in
let (blindcommit,idvcheck) =
                    checksign(req,pkvt) in
if identvoter = idvt then
  if idvcheck = idvt then
  out(authCh,(encrypt((k2,

sign((blindcommit,g(k2),idas),seckas)),

pkvt))).
```

پردازه ۸- پردازه مرجع احراز اصالت

۶-۱۰-پردازه مرجع رأی‌گیری

پردازه مرجع رأی‌گیری (پردازه ۹)، تعرفه رأی نهایی رأی‌دهنده را از کانال گمنام *voteCh* دریافت کرده و اعتبار آن را بررسی می‌کند. این بررسی شامل کنترل امضای AS بر روی تعرفه، کنترل کلیه پارامترهای بکار رفته در تولید تعرفه و وارسی اعتبار کلید عمومی رأی‌گیری می‌شود. همچنین رأیی که با کلید خصوصی رأی‌گیری امضاء شده است نیز وارسی می‌شود. در صورت صحت موارد مذکور، تعرفه رأی‌گیری معتبر از طریق کانال *tallingCh* به مرجع شمارش آراء ارسال و رویداد ENDVOTE اجرا می‌شود.

```
let VS =
(* Authorities Public Keys *)
in(pkasCh2,pubkeyas);
let identaserver = idnt(pubkeyas) in
phase 1;
phase 2;
in(voteCh,ticket);
(* Check Validation *)
let (vt,g_r,pkeyvoter,msig,assign) =
                                    ticket
in
let (gr,gk1,gk2,idaserver) =
                checksign(assign,pubkeyas)
in
if checkg(gk1,gk2,pkeyvoter) = true then
  if g_r = gr then
    if idaserver = identaserver then
      if dec(msig,pkeyvoter) = vt then
      event ENDVOTE(vt);
      (* Output Voting Ticket *)
      out(tallingCh,ticket).
```

پردازه ۹- پردازه مرجع رأی‌گیری

۷- وارسی خصوصیت صحت در پروتکل لین و همکارانش

مدل صوری مربوط به خصوصیت صحت در پروتکل لین، در پردازه ۱۰ نشان داده شده است. در این مدل پرس‌وجوها بصورت برهان‌های مطابقت یک‌به‌یک و با استفاده از evinj نشان داده شده‌اند.

طبق توصیف صوری (تعریف ۳)، خصوصیت غیر قابل تغییر بودن به این صورت مدل می‌شود که هر رویداد ENDVOTE زمانی اجرا شود که قبل از آن، به ترتیب رویدادهای STARTID و BEGINVOTE متناظر با آن توسط رأی دهنده درستکار اجرا شده باشند، و یا رویداد STARTCORID توسط رأی‌دهنده فاسد اجرا شده باشد. درصورت درست بودن نتیجه این پرس‌وجو، رأی داده شده توسط رأی‌دهنده درستکار تا هنگام شمارش تغییر نکرده است. همچنین درصورتیکه رأی شمارش شده توسط مهاجم داده شده بود، وی از شناسه رأی‌دهنده فاسد برای رأی دادن استفاده کرده است. خصوصیت عدم استفاده مجدد به اینصورت مدل می‌شود که رویداد STARTID یا STARTCORID فقط یک بار در هر دنباله اتفاق بیفتد و با سایر رویدادهای موجود تناظر داشته باشد. بدین ترتیب به ازای هر ثبت‌نام تنها یک بار رأی ارسال می‌گردد. درنهایت خصوصیت مجاز بودن به این صورت مدل می‌شود که درصورتی رویداد STARTID یا STARTCORID اجرا شود که قبل از آن رویداد NEWID اجرا شده باشد. بدین ترتیب فقط رأی‌دهندگان مجازی که شناسه دریافت کرده باشند قادر به شروع فاز ثبت نام هستند.

```
(* INALTERABILITY *)
query evinj:ENDVOTE(v) ==>
     ((evinj:BEGINVOTE(id,v) ==>
          evinj:STARTID(id)) |
      evinj:STARTCORID(corvid)).
(* NON-REUSABILITY *)
query evinj:ENDVOTE(v) ==>
   (((evinj:BEGINVOTE(id,v) ==>
          evinj:STARTID(id)) &
                evinj:NEWID(id))|
     (evinj:STARTCORID(id) ==>
          evinj:NEWID(id))).
(* ELIGIBILITY *)
query evinj:STARTID(id) ==>
evinj:NEWID(id).
query evinj:STARTCORID(id) ==>
          evinj:NEWID(id).
```

پردازه ۱۰- پردازه وارسی خصوصیت صحت

به علت اجرای نسخه‌های نامحدود از پردازه‌ها و همچنین استفاده از برهان‌های مطابقت یک‌به‌یک، به هنگام پیاده سازی پروتکل در پروِوریف از تبادل نانس[1] استفاده شده است. بدین ترتیب هر نسخه از یک پردازه درحال اجرا، به هنگام استفاده از کانال‌های اشتراکی، با پردازه‌های دیگری که متناظر با وی هستند تعامل خواهد داشت. ابزار پروِوریف پروتکل را بصورت مجموعه‌ای از قوانین پرولوگ[2] توصیف می‌کند، سپس در این قوانین به وارسی خصوصیات مورد پرس‌وجو می‌پردازد. در شکل ۲ خروجی این ابزار در وارسی خصوصیت مجاز

[1] Nonce-handshake

[2] Prolog rules

بودن نمایش داده شده است. نتیجه اجرای پردازه ۱۰ در ابزار پروِریف نشان می‌دهند که خصوصیت صحت در پروتکل لین و همکارانش برقرار است.

```
-- Query evinj:STARTCORID(id_150) ==>
        evinj:NEWID(id_150)
Starting query evinj:STARTCORID(id_150) ==>
             evinj:NEWID(id_150)
RESULT evinj:STARTCORID(id_150) ==>
       evinj:NEWID(id_150) is true.
-- Query evinj:STARTID(id_35146) ==>
        evinj:NEWID(id_35146)
Starting query evinj:STARTID(id_35146) ==>
             evinj:NEWID(id_35146)
RESULT evinj:STARTID(id_35146) ==>
       evinj:NEWID(id_35146) is true.
```

شکل ۲ – وارسی خصوصیت مجاز بودن در ابزار پروِریف

۸- نتیجه‌گیری

در این مقاله پروتکل لین و همکارانش با استفاده از حساب پی کاربردی و ابزار پروِریف به صورت صوری مدل گردید. همچنین خصوصیات غیرقابل تغییر بودن، عدم استفاده مجدد و مجاز بودن بصورت صوری توصیف و برقراری آنها در پروتکل مذکور مورد وارسی قرار گرفت. در این مدل‌سازی از برهان‌های مطابقت جهت خودکار کردن روند وارسی و توصیف خصوصیات امنیتی بهره گرفته شد.

تشکر

نویسندگان مقاله مایلند از همکاری آقای محمدهاشم حقیقت[1] در انجام این تحقیق تشکر و قدردانی نمایند.

مراجع

[1] M. Abadi, C. Fournet. "Mobile values, new names, and secure communication," In Proc. 28th ACM Symposium on Principles of Programming Languages–POPL'01, ACM, 2001, pp. 104–115.

[2] M. Abadi, B. Blanchet, C. Fournet. "Just fast keying in the pi calculus," In 13th European Symposium on Programming–ESOP'04, vol. 2986 of LNCS, Springer, 2004, pp. 340–354.

[3] M. Backes, C. Hritcu, M. Maffei. "Automated verification of remote electronic voting protocols in the applied pi-calculus," In Proc. 21st IEEE Symposium on Computer Security Foundations–CSF'08, IEEE, 2008, pp. 195–209.

[4] B. Blanchet. "An efficient cryptographic protocol verifier based on Prolog rules," In Proc. 14th IEEE Computer Security Foundations Workshop–CSFW, IEEE, 2001, pp. 82–96.

[5] B. Blanchet. "Automatic verification of correspondences for security protocols," In Journal of Computer Security, vol. 17, 2009. pp. 363-434.

[6] S. Delaune, S. Kremer, M. Ryan. "Verifying privacy-type properties of electronic voting protocols," Research Report LSV-08-01, 2008.

[7] S. Delaune, M. Ryan, B. Smyth. "Automatic verification of privacy properties in the applied pi calculus," To appear in 2nd Joint iTrust and PST Conferences on Privacy, Trust Management and Security–IFIPTM'08, 2008.

[8] S. Delaune, S. Kremer, M. Ryan. "Coercion-resistance and receipt-freeness in electronic voting," In Proc. 19th Computer Security Foundations Workshop–CSFW'06, IEEE, 2006, pp. 28–39.

[9] S. Delaune, S. Kremer, M. Ryan. "Verifying Properties of Electronic Voting Protocols," In Proc. of IIAVoSS Workshop On Trustworthy Elections–WOTE'06, 2006, pp. 45-52.

[10] A. Fujioka, T. Okamoto, K. Ohta. "A practical secret voting scheme for large scale elections," In Advances in Cryptology – AUSCRYPT '92, vol. 718 of LNCS, Springer, 1992, pp. 244–251.

[11] S. Kremer, M. Ryan. "Analysis of an electronic voting protocol in the applied pi-calculus," In Proc. 14th European Symposium on Programming–ESOP, LNCS, Springer, 2005, pp. 186–200.

[12] I. Lin, M. Hwang, C. Chang, "Security enhancement for anonymous secure e-voting over a network," In Computer Standard and Interfaces, vol. 2, 2003, pp.131–139.

[13] B. Smyth, M. Ryan, S. Kremer, M. Kourjieh. "Election verifiability in electronic voting protocols," In 4th Benelux Workshop on Information and System Security–WISSec'09, 2009.

[1] دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف