احراز هویت مبتنی بر ویژگی سرور ناشناس در محاسبات ابری

ولی الله وحیدی نژاد

دانشجوی کارشناسی ارشد مهندسی نرم افزار گرایش نرم افزار ، دانشگاه پیام نور واحد تهران ، ایران

vahidi.best90@yahoo.com

مریم اسکندر پور

فارغ التحصیل کارشناسی ارشد مهندسی فناوری اطلاعات گرایش سیستمهای چند رسانه ای ، دانشگاه پیام نور واحد عسلویه ، ایران

mohandes.eskandarpour@gmail.com

چکیده

مفهوم اثر مبتنی بر ویژگی یکی از مهم ترین موارد ایمنی برای درک احرازهویت ناشناس می باشد. در اثر مبتنی بر ویژگی (ABS)، کاربران می توانند با ویژگی های خود، اثری بر پیام تولید کنند. با این اثر، هرتصدیق کننده متقاعد خواهد شد که یک اثر از نشان دهنده با این ویژگی ها تولید شده است. هرچندکه، تشخیص نشان دهنده از تصدیق کننده پنهان خواهد ماند. ABS برای طرح احرازهویت ناشناس و سیستم های پیام رسانی مبتنی بر ویژگی سودمند است. اگرچه، وجود عملکرد احرازهویت مبتنی بر ویژگی معمولا" به محاسبات بسیاری حین نشان کردن نیاز دارد، که بنا بر میزان ویژگی ها، به صورت خطی رشد می کند. پس این روش ها در محاسبات سنگین کاربری نتیجه می دهد، اما برای ابزاری که تنها توانایی محاسبات کوچک را دارند، مناسب نمی باشد. محاسبات ابری نوعی محاسبه است که درآن منابع اساسی مصاحبه به صورت خدمات اینترنتی توسعه یافته اند. دراین تحقیق، ما درابتدا چالش ایمنی منابع ABS رابیان کرده ایم، که کاربران را در کاهش هزینه ی محاسبه ی تولید ABS ، توانا می سازد. این مدل امنیتی درحالی که منابع محاسبه نشان گذاری خارجی می باشند، معمولا" جهت حفظ حریم شخصی کاربر تعریف شده است. یک الگوریتم موثر و ایمن ABS نیز پیشنهاد شده است. تحلیل های توسعه یافته نشان می دهند که طرح ما در مدل پیشنهادی ایمنی دارد و بیش از ۹۰٪ از محاسبات کاربری را ذخیره می سازد.

واژگان کلیدی: احرازهویت، محاسبات ابری، تصدیق کننده، خدمات اینترنتی

۱. مقدمه

اثر مبتنی بر ویژگی یک اثر دیجیتال مهم است، که اثر مبتنی بر تشخیص را توسعه می بخشد، دراین اثر نشان کنده با مجموعه ای از ویژگی های بیان شده در تشخیص نشان کننده، تعریف می شوند. در ABS، کاربر برای مجموعه ای از ویژگی های احراز هویت، اعتبار بدست می آورد. تصدیق کننده ی ABS متقاعد است که نشان کننده ای که به پیش بینی مجموعه ی پیچیده ای از ویژگی ها می پردازد، پیام را تایید می کند.

ABS کاربردهای بسیار مهمی دارد و با ویژگی رمزنگاری نیز ارتباط دارد. برای مثال، ABS می تواند برای کنترل دسترسی در احرازهویت ناشناس بکار رود. کاربرد مهم دیگر ABS را می توان در ویژگی مبتنی بر سیستم پیام رسانی یافت، که درآن دریافت کننده ی پیام مطمئن است که پیام دریافت شده توسط کاربری با ویژگی های متناسب ارسال شده است. همچنین سناریوی کاربرد ABS درخدمات آن لاین کتاب الکترونیک حائز اهمیت است، این خدمات به کاربران اجازه ی کاربرد آزاد خدمات به مدت ۳۰ روز را می دهد. راه حل این است که یکی باید اعتبار آدرس ایمیل را با دو پروف مستقل از شناسایی او – مثل حساب بانکی، یک گذرنامه، یا یک کارت اعتباری- توسعه بخشد. لازم به ذکر است که دراین سناریو، حمایت از موضوعاتی با هدف ثبت اعتبار شناسایی، نیاز می باشد. با این تفکر، می توان نشان مشروحی را ایجاد کرد که دو موضوع پشتیبان موردنیاز را با کاربرد تکنیک ABS بیان می کند.

ABS می تواند در کاربردهای مهمی استفاده شود، اگرچه، تولید ABS به دلیل رشد خطی درکنار تعدادی ویژگی، به محاسبات گسترده ای نیاز دارد. همه ی طرح های پیشین ABS این نقص را داشته اند. برای ابزار محاسبه ی سنتی، مثل صفحات کامپیوتری، این محاسبات به سادگی پردازش می شوند. هرچندکه، برای ابزار سیار رایجی چون تلفن های همراه، برخی از محاسبات با چالش همراه می باشند.

محاسبات ابری یک الکوی محاسباتی جدید است و توانایی توسعه ی منابع عظیم و محاسبه را برای کاربران به همراه دارد. با تکنیک محاسبات ابری، کاربران توانایی برون سپاری محاسبات خود و ذخیره ی اعمال خود در سرورهای ابری را خواهندداشت. درنتیجه، این الگوی محاسباتی جدید، حتی برای کاربرانی که تنها با ابزار همراه توانایی محاسبات محدودی را دارند، نیز کمک کننده است. محاسبات ابری، به دلیل این خصوصیات توجه زیادی را به خود جلب کرده است. این الگو هنگامی که کاربر برای محاسبات برون سپاری، به مزیت غیرحقیقی ابری گرایش دارد، با چهار چالش جدید روبه رو می شود. این چالش ها وقتی که کاربر خواستار عملکردهای خصوصی در محیط ابری می شود – مثل تولید اثر- ترسناک تر می شوند.

برای دسترسی کاربر و و ارسال کلید حریم خصوصی به توسعه دهنده ی ابری، یک روش بسیار ساده وجود دارد. با این کلید، سرور ابری می تواند اثری تولید کند و آن را به کاربر بازگرداند. هرچندکه، این امر به اعتماد کامل ابری نیاز دارد. این ابر می تواند اثرات پیام های کاربر را تولید کند. بنابراین، روش پیش رو، وقتی که ابر کاملا" مورد اطمینان نیست، ایمنی ندارد. روش دوم، روش مفیدی ست که در طرح های موثر سرور استفاده می شوند. هرچندکه، طرح های پیشین سرور برای کاهش محاسبات با توسعه ی الگوریتم محاسباتی سرور طراحی شده اند. با کاربرد این روش ها در ABS اثرپذیری بیشتری خواهیم داشت. روش سوم جهت کاربرد روش های برون سپاری، با رمزنگاری یکنواخت یا تبادل سیستم های پروف، می باشد. اما گنتری نشان داده است که تکنیک عملکرد یکنواخت، درحال حاضر عملی نمی باشد و به ۳۰s بر یک ماشین بسیار با کیفیت، نیاز دارد. بنابراین، با این روش ها، حریم خصوصی ورودی و خروجی می تواند با کاربرد رمزنگاری یکنواخت حفظ شود، به طور کلی این تکنیک ها غیرکاربردی ست.برای مشکلات ویژه ی محاسبه نیزعملکردهایی بر محاسبات برون سپاری وجود دارد.

دراین تحقیق، ما بر طرح برون سپاری و توجه به ABS تاکید داریم. ما یک پروتکل ABS -برون سپاری (ABSO) ایمن وموثری را پیشنهاد داده ایم که بیشتر محاسبات تولید اثر را از بین می برد، درحالی که ایمنی و حریم خصوصی کاربر را نیز حفظ می کند. دراین پروتکل، کاربر یک کلید برون سپاری واحد در سرور ابری ایجاد می کند که طی آن سرور ابری توانایی تولید یک نیمه اثر را خواهد داشت. وقتی که کاربر یک نیمه اثر دریافت می کند، می تواند آن را به یک ABS معتبر با چند کاربرد موثر انتقال دهد. برای تحلیل ایمنی، اول از همه، ایمنی قابلیت تصدیق و جعل را به صورت معمولی تعریف می کنیم. سپس ثابت می کنیم که طرح ما در مدل استاندارد، تعاریف ایمنی کافی را دارا می باشد. طرح ABSO ما براساس اثرپذیری طرح ABS پیشنهاد شده توسط لی می باشد، هرچندکه، روش پیشنهاد شده در این تحقیق می تواند به دیگر طرح های ABS توسعه یابد.

این تحقیق، بدین صورت سازمان دهی شده است. دربخش ۲ مقدمات طرح خود را ارائه کرده ایم. در بخش ۳ مدل سیستم و تعریف ایمنی طرح خود را آورده ایم. ساختار طرح ما در بخش ۴ بیان شده است. دربخش ۵ تحلیل موثر و پروف ایمنی طرح خود را نشان داده ایم. درپایان، جمع بندی این تحقیق در بخش ۶ آورده شده است.

۲. مقدمات

۲.۱. نقشه های دو خطی

G و G_T گروه های دوره ای اولیه ی P می باشند، g و g_T به ترتیب تولید کننده های G و G_T می باشند. یک نقشه دو خطی $e: G\times G\rightarrow G_T$ درصورتی که موقعیت های زیر برقرار باشد، یک جفت دو خطی نامیده می شود:

***دوخطی:** برای همه ی $g\epsilon G^*$، g یک تولید کننده از G است، و

$$a,b \epsilon Zp,\ e(g^a, g^b)= e(g, g)^{ab}$$

***انحطاط:** $e(g, g) \neq ۱$. به عبارت دیگر، می توان گفت که اگر gگروه G را تولید کند، پس e(g ،g) به تولید G_T می پردازد.

***اثرپذیری:** دراین مورد یک الگوریتم موثر برای محاسبه ی e (.، .) وجود دارد.

۲.۲. ساختار در دسترس

تعریف ۱: (دسترسی به ساختار). ما مجموعه ی $\{P_۱, P_۲, ... P_N\}$ را به صورت بخش های شامل شده تعریف می کنیم. $\{P_۱, P_۲, ... NP\}$ ۲ CA درصورتی که هر B و C با موقعیت $B\epsilon A$ و CCB باشد یک مونوتن نامیده می شود. بدین ترتیب داریم $C\epsilon A$. ما دسترسی به یک ساختار را به صورت A از $\{P_۱, P_۲, ... P_N\}$ تعریف می کنیم. پس، همه ی مجموعه های بدست آمده در A مجموعه های احراز هویت می باشند.

در سیستم های مبتنی بر ویژگی، این بخش با ویژگی ها تعریف شده است. براساس این تعریف، همه ی کاربران احراز هویت شده در مجموعه ی A حضور دارند. در جزییات بیشتر، این طرح جدید همه ی پیش بینی های Υ شامل مقدار آستانه می شود که به صورت زیر بیان می شود:

$$\Upsilon_{k,\omega^*}(\omega) = \begin{cases} 1 & \text{if } |\omega^* \cap \omega| \geq k \\ 0 & \text{otherwise} \end{cases} \qquad (1)$$

که درآن ω^* و ω مجموعه ویژگی ها و κ تعریف اولیه ی مقدار آستانه می باشد. بنابراین، مجموعه های احراز هویت شده دراین مفهوم شامل همه ی مجموعه ویژگی های ω است که به صورت زیر نشان داده می دهد،

$$\Upsilon_{\kappa,\omega^*}(\omega)=1$$

۲.۳. فرضیه

فرضیه ی محاسبه ی دیفی-هلمن (CDH): فرضیه ی استاندارد CDH به صورت زیر تعریف می شود. برای هر احتمال الگوریتم زمان پلینومینال A، توابع چشم پوشی $gen(.)$ برای همه ی nها وجود دارد.

$$|Pr[\mathcal{A}(1^n, g, g^x, g^y) = g^{xy}]| \leq neg(\cdot)$$

که درآن gژنراتور گروه G از سفارش P است که اولین طول تقریبی $n,x,y\epsilon Z_P$ می باشد. اگر هیچ تقابل A وجودنداشته باشد، می گوییم که CDH -(t, ϵ) در Gحفظ شده است.

۳. تعریف مدل سیستم و ایمنی

۳.۱. مدل

سه کلیت در یک سیستم وجود دارند، ویژگی احرازهویت A، کاربر U و سرور ابری S. ویژگی احراز هویت یک کلید خصوصی برای کاربران است. کاربر بیشتر محاسبات تولید شده را برای سرور ابری Sبرون سپاری می کند. U برای ارائه ی ویژگی کلی سودمند است. به طور خلاصه، این ویژگی ها براساس طرح موثر با برون سپاری توابع الگوریتم های زیر می باشد:

اجرا (λ,U): الگوریتم اجرایی پارامتر ایمنی و ویژگی کلی رابه عنوان ورودی نشان می دهد. این ورودی پارامتر عمومی PK و کلید اصلی SK را برای ویژگی احرازهویت نتیجه می دهد.

استخراج (ω, SK): الگوریتم استخراج کلید اصلی و مجموعه ویژگی های ورودی را نشان می دهد. این ورودی مجموعه های منطبق بر حریم خصوصی SK_ω را برای کاربر تولید می کند.

KeyGenout (skω, T) الگوریتم تولیدی کلید برون سپاری، کلید خصوصی SK_ω کاربر و مجموعه T که به طور تصادفی انتخاب شده است را به عنوان ورودی بیان می کند. این ورودی کلید برون سپاریSK_{tuo} را برای سرور ابری S تولید می کند.

Signout (skout , Y) این الگوریتم کلید برون سپاری و پیش بینی را به عنوان ورودی نشان می دهد. این ورودی نیمه اثر σ_{lah} رابنا بر پیش بینی Υ نتیجه می دهد.

تایید (PK, σ, Υ) پس از دریافت اثر σ بر پیام m و مجموعه ویژگی ω با پیش بینی Υ، تایید کننده درصورتی خروجی را دارد که، $\Upsilon(\omega)=$۱و۰.

۳.۲. تجهیزات ایمنی

جعل: در سیستم ما یکی از تجهیزات اساسی ایمنی این است که هر دشمن، حتی در سرور ابری هم نمی تواند بر پیش بینی های انجام شده اثر داشته باشد. به طور ویژه، ما به طرح پیشنهاد شده ای نیاز داریم که وجود دشمن را در برابر پیش بینی های منتخب و پیام هجوم نشان دهد. این تعریف برای ایمنی از جعل همراه با چالش CوF پیگیری می شود.

*اجرا: پارامتر بزرگ ایمنی 1^{λ} انتخاب می شود. C کلید رمز SK است و یک PK به F ارسال می کند.

*نمایش: نمایش ها موضوعی با ω و (m,Υ) می باشند. علاوه براین F می تواند نمایش یک عدد پلینومینال بر کلید برون سپاری باشد.

جعل اسناد: درپایان، F خروجی اثر معتبر σ^{*} بر پیام m بدون پیش بینی ویژگی Υ^{*} است.

اگر σ^{*} اثر معتبر بر پیام *m برای Υ^{*} باشد، دشمن دراین بازی پیروز است و مقدار $\Upsilon^{*}(\omega) = 1$ برای ویژگی کلید خصوصی استخراج خواهد شد. مزیت این مقدار به صورت یک احتمال پیروزی تعریف می شود.

تعریف ۲: (جعل). یک دشمن F درصورتی که در زمان t کار خود را اجرا کند، برون سپاریABS را می شکند و ویژگی qk و qs برای اثرهای بیان شده نمایش داده می شوند.

دراین تحقیق، پیش بینی هجوم دشمن نیز مورد توجه می باشد. تعریف کلی EUF-sP-CMA نیز می تواند براساس بازی میان چالش C و جعل F باشد. پیش بینی بازی منتخب همانند بازی توصیف شده در بالا می باشد، به استثناء اینکه دشمن می تواند چالش را پیش از شروع اجرای الگوریتم پیش بینی و انتخاب کند. این مدل بسیار رایج است و در بسیاری از ساختارهای دیگر نیز بکار رفته است.

*مقداردهی اولیه: F یک پیش بینی $\Upsilon_{k,\omega*}$ را انتخاب می کند که حاوی اثر جعل است و آن را به چالش C می فرستد.

*اجرا: پس از دریافت چالش پیس بینی F، یک پارامتر امنیتی 1^{λ} انتخاب شده و کار اجرا شروع می شود. C کلید رمز sk را حفظ کرده و pk تولید شده از اجرای F را ارسال می کند.

*نمایش: عدد نمایش داده شده ی ω و (m,Υ) به کلید خصوصی استخراج شده ارسال می شود. علاوه براین، F می تواند عدد پلینومینال را برای کلید برون سپاری به اجرا گذارد.

*جعل اسناد: درپایان، دشمن اثر معتبر σ^{*} را برای پیام m نتیجه می دهد.

تعریف ۳: (جعل انتخاب-پیش بینی). یک جعل اسناد F در صورتی که در زمان t اجرا شود، طرح برون سپاریABS را می شکند، کلید خصوصی qk استخراج می شود، اثر qs نمایش داده می شود، و q_{0} تولید می شود.

اثبات پذیری: اثبات پذیری نیاز دارد که نتایج توسط سرور ابری که می تواند برای کاربر موثر باشد، بازگردانده شود. در سیستم ما، نیاز است که کاربر صحت نیمی از آثار را تضمین کند. علاوه براین، روش اثبات پذیری باید به محاسبات کمتری نیاز داشته باشد. به طور کلی، در سیستم پیشنهادی ما، نیاز به ایمنی اثبات پذیری احتمالی ست که طی آن سرور ابری در تولید اثری که قابل چشم پوشی باشد، توانایی لازم را دارد.

۴. ساختار

همان طور که در بخش های پیشین ذکر شد، طرح ما شامل شش الگوریتم می شود. همه ی این عملکردها، با گروه G و G_T انجام می شوند. ساختار این طرح می تواند به صورت زیر باشد:

*اجرا (λ،U): ابتدا، ویژگی احراز هویت با ویژگی کلیت U به صورت عناصر Z_P تعریف می شود.۱-Ad مجموعه ویژگی Z_P است. یک ژنراتور تصادفی gϵG را انتخاب کرده، سپس آن را محاسبه کنید. الگوریتم اجرا شده نیز با دو تابع هش H۱ و H۲ تعریف می شوند. پارامترهای کلی از مجموعه ی زیرمی باشند؛

$PK=(g,g_1,g_2,Z,d,H_1,H_2)$

و کلید رمز اصلی ویژگی احراز هویت X= sk است.

استخراج (sk ،ω): یک کلید خصوصی برای کاربر U با مجموعه ویژگی ω، می باشد، فرآیند زیر با ویژگی احرازهویت اجرا می شود:

.۱-Ad درجه ی پلینومینال q(y) است که به صورت تصادفی و در شرایط X= q(۰) انتخاب شده است.

.سپس، مجموعه ی ω جدیدی تعریف می شود. برای هر iϵω، عددی انتخاب و محاسبه می شود.

.درپایان، خروجی مورد نظر کلید خصوصی برای هر i خواهد بود.

سرور ابری S بدین صورت محاسبه می شود:

$$\sigma_0' = \prod_{i\in\omega'\cup\Omega'} d_{i,0}'^{\Delta_{i,S}(0)} \prod_{i\in\omega^*\cup\Omega'} H_1(i)^{r_i'}$$
$$\sigma_i' = d_{i,1}'^{\Delta_{i,S}(0)} g^{r_i'} \quad \text{for } i \in \omega' \cup \Omega'$$
$$\sigma_i' = g^{r_i'} \quad \text{for } i \in \omega^*/\omega'.$$

بنا بر تئوری لاگرانژ الحاقی، d بدست آمده از نقاط q(d)،...، q(۲)، q(۱) بر پلینومینال ۱-dمی باشد، بدین ترتیب ما می توانیم q(i) رابرای هر i محاسبه کنیم. S نیز از مجموعه عناصر d می باشد. این ضریب به صورت زیرتعریف می شود:

$$\Delta_{i,S}(j) = \prod_{k\in S, k\neq i} \frac{j-k}{i-k}.$$

*نشانه: وقتی که U از یک سرور ابری σ_{lah} دریافت می شود، سرور S و U محاسبه می شوند، و اعداد تصادفی در کلید برون سپاری الگوریتم تولید شده بکار می روند. سپس U برای تولید اثر کاملی از پیام m محاسبه می شود.

$$\because \hat{\sigma} = \sigma_0'/V.$$

.U با مقدار تصادفی S انتخاب شده و محاسبه می شود.

.درپایان، U اثر کاملی از پیام m را با پیش بینی $\Upsilon_{k,\omega*}$ را به صورت خروجی نشان می دهد. برای تایید صحت این اثر، U به صورت معادله ی زیر تصدیق می شود؛

$$\frac{e(g,\sigma_0)}{\prod_{i\in\omega^*\cup\Omega'} e(H_1(i),\sigma_i)e(H_2(m),\sigma_0')} = Z$$

اگر این معادله برقرار باشد، U مقدار σ را برای تایید کننده ارسال می کند، بدین ترتیب سرور ابری ناصحیح باقی می ماند.

*تایید: برای تایید اثر σ از پیام m با پیش بینی $\gamma_{k,\omega*}$، تایید کننده با معادله ی زیر بررسی می شود؛

$$\frac{e(g,\sigma_0)}{\prod_{i\in\omega^*\cup\Omega'} e(H_1(i),\sigma_i)e(H_2(m),\sigma_0')} = Z$$

از ساختار بالا می توانیم بفهمیم که اگر کاربر U اثر سودمندی بر الگوریتم تایید کننده داشته باشد، نیمه اثر به سرور ابری بازمی گردد. U نیز با زمان عمل مصرف محاسبه می شود. هرچندکه، ما می توانیم ساختار بالا را تعدیل بخشیده و فرآیند تایید را موثرتر انجام دهیم. بدین ترتیب سرور ابری S را به صورت زیر محاسبه می کنیم؛

$$\sigma_0' = \prod_{i\in\omega'\cup\Omega'} d_{i,0}'^{\Delta_{i,S}(0)} \prod_{i\in\omega^*\cup\Omega'} H_1(i)^{r_i'}$$
$$\sigma_i' = d_{i,1}'^{\Delta_{i,S}(0)} g^{r_i'} \quad \text{for } i \in \omega'\cup\Omega'$$
$$\sigma_i' = g^{r_i'} \quad \text{for } i \in \omega^*/\omega'.$$

نکته: از لحاظ تئوری، سرور ابری ناصحیح S می تواند به صورت بالا محاسبه شود، بدین ترتیب داریم؛

$$\hat{Z} = \frac{e(g,\sigma_0')}{\Pi_{i\in\omega^*\cup\Omega'} e(H_1(i),\sigma_i')}.$$

سپس، سرور ابری می تواند با نیمه اثرσ_{lah} تعدیل یافته و معادله ی زیر را تولید کند؛

$$\overline{\sigma} = Sig_{sk_s}(\sigma_{hal}', \hat{Z}).$$

Uبرخی از خطاهای نیمه اثر را خواهد پذیرفت. هرچندکه، وقتی U اثر پایانی σ را تولید کند، ویژگی بدست آمده نامعتبر خواهد بود. بنابراین، وقتی تایید کننده اثر نامعتبر σ را به همراه داشته باشد، U می تواند برای حل مشکل بازیابی شود. دراین حالت یک داور می تواند محاسبه را مجددا" انجام دهد و مقدار زیر را به عنوان پروف S بدست آورد؛

$$\hat{Z} \neq \frac{e(g,\sigma_0')}{\Pi_{i\in\omega^*\cup\Omega'} e(H_1(i),\sigma_i')}$$

بنابراین، سرور ابری ناصحیح شارژ خواهد شد. این امر به ندرت رخ می دهد، منابع محاسباتی نیز آمادگی لازم را خواهند داشت،اما نتایج اشتباه درپایان به سرور بازمی گردند. بدین ترتیب، الگوریتم تایید کننده در بالا موثر و کاربردی است.

۵. تحلیل ایمنی و اثرپذیری

۵.۱. تحلیل اثرپذیری

دراین بخش، مقایسه ی میان ساختار ما و طرح اصلی ABS آورده شده است. بنا بر واژه شناسی ما، A و B مجموعه ای از ویژگی های پیش بینی شده و کاربری می باشند. درمنبع (۱)، تولید یک ABS به ۳/۲(|A| +d-k) در مقدار G نیاز دارد، دراین مورد k مقدار آستانه و d مقدار اولیه ای ست که از ۱ تا ۱-d مقدار دهی می شود. درحالی که در طرح ما، محاسبات پیچیده ی تولید مقدار معتبر ABS تنها به ۲ مقدار و ۱ کاربرد معکوس در G نیاز دارد. برای تایید صحت نیمه اثر، Uمقدار ۱ را خواهد داشت.

برای تولید یک کلید برون سپاری، کاربر مقدار |Bl+d-۱ را محاسبه می کند، این مقدار برای مجموعه ویژگی ها مقداری خطی ست. هرچندکه، کاربر تنها کلید برون سپاری را برای همه ی موارد تولید می کند. بنابراین، هزینه ی محاسبات تولیدی بسیار کم خواهد بود. علاوه براین، وقتی به این سناریو توجه نشان می دهیم، قابلیت محاسبه ی کاربر محدود می شود، و کلید برون سپاری می تواند بر صفحه ی کامپیوتر تولید شده و برای سرور ابری ارسال شود. بنابراین، هیچ نیازی به تولید کلید برون سپاری در هنگام استفاده ی کاربر از ابزار سیار نمی باشد. در مجموع، هزینه ی محاسبات آن لاین برای تولید یک ABS در طرح ما، به مقدار (k-d+|Al۳/۲ نیاز دارد. اگر ما محاسبات آن لاین طرح خود را با طرح اصلی ABS مقایسه کنیم، می بینیم که طرح ما می تواند بیش از ۹۰٪ هزینه ی محاسبات را ذخیره سازد. با توجه به پیچیدگی ارتباطات، کاربر به ارسال اثر تولید شده به سرور ابری می پردازد و عناصر |Al+d-k+۲ از G را دریافت می کند.

ارتباط میان سرور ابری و کاربر تنها ده ها کیلوبایت است. در شبکه های پیشرفته، ارتباطات طرح ما قابل توجه نمی باشد.

۵.۲. تحلیل ایمنی

تئوری ۱: بافرض محاسبات موجود در G، و انتخاب تصادفی H۱ و H۲، استخراج کلید خصوصی، تولید کلید برون سپاری و تولید اثر را خواهیم داشت. پس، طرح ABS پیشنهادی ما به صورت زیر خواهد شد،

$$(t, q_{H_1}, q_{H_2}, q_K, q_O, q_S, \epsilon)\text{-EUF-sP-CMA},$$

که درآن

$$(q_{H_1}+q_{H_2}+2q_K+q_O+3q_S d)t_{exp}, t_{exp}$$

زمانی برای تقریب زیر خواهد بود؛

$$\epsilon' \approx \epsilon/(q_{H_2} \binom{d-k}{d-1}).$$

پروف: فرض می کنیم که F می تواند ایمنی طرح پیشنهادی ما را با احتمال عدم چشم پوشی بشکند، بدین ترتیب ما الگوریتم A را طراحی کرده و فرضیه ی CDH را نشان می دهیم. بنابراین ما به نشان دادن کاربردهای حقیقی و پروف نیاز داریم.

A دو مقدار L۱ و L۲ را برای واکنش با H۱ و H۲ ثبت می شود. فرض می کنیم که F بیشتر زمان خود را برای H ۱ و H۲ سپری می کند. بدین ترتیب خواهیم داشت:

*اگر نمایش تصادفی i برای مقدار تصادفی H۱(i) باشد، A درابتدا فهرست L۱ را بررسی می کند. پس این مقادیر تصادفی با مقادیر موجود به صورت زیر محاسبه خواهند شد؛

$$H_1(i) = \begin{cases} g^{\beta_i} & \text{for } i \in \omega^* \cup \Omega^*,\ \beta_i \in_R \mathbb{Z}_p \\ g_1^{-\beta_i} g^{\gamma_i} & \text{for } i \notin \omega^* \cup \Omega^*,\ \beta_i, \gamma_i \in_R \mathbb{Z}_p. \end{cases}$$

*اگر m به H۲(m) ارسال شود، A اولین مقدار تصادفی منتخب خواهد بود و فهرست L۲ را بررسی خواهد کرد. مقادیر بدست آمده برای Fبازگردانده می شوند.

$$H_2(m) = \begin{cases} g_1^{\alpha_i} g^{\beta_i'} & \text{for } i \neq \delta,\ \alpha_i, \beta_i' \in_R \mathbb{Z}_p \\ g^{\beta_i'} & \text{for } i = \delta,\ \beta_i' \in_R \mathbb{Z}_p. \end{cases}$$

*برای هر i که به طور تصادفی از مجموعه ی zp انتخاب می شود، داریم؛

$$g_2^{q(i)} H_1(i)^{r_i} = g_2^{\frac{\Delta_{0,S}(i)\gamma_i}{\beta_i} + \Sigma_{j \in I'} \Delta_{j,S}(i) q(j)} H_1(i)^{r_i'},$$
$$g^{r_i} = g_2^{\frac{\Delta_{0,S}(i)}{\beta_i}} g^{r_i'}.$$

که دراین معادله، F درخواستی برای نمایش اثر بر پیام m با پیش بینی مقدار Υ است. اگر مقدار مورد نظر تولید شود، کلید خصوصی به عنوان یک کلید استخراج همسان سازی می شود و اثر نرمالی تولید می کند. اگر مقدار مورد نظر به صورت تصادفی انتخاب شود، می تواند به صورت زیر همسان سازی شود؛

در همسان سازی مقدارزیر

$$(g_2^x \Pi_{i \in \omega' \cup \Omega'} H_1(i)^{r_i} H_2(m)^s, \quad \{g^{r_i}\}_{i \in \omega' \cup \Omega'}, g^s),$$

A انتخاب شده و خواهیم داشت،

$$s = -\frac{1}{\alpha_{i_d}} y + s'.$$

پس داریم،

$$g_2^x \Pi_{i \in \omega' \cup \Omega'} H_1(i)^{r_i} H_2(m)^s = (g_1^{\alpha_{i_d}} g^{\beta_i})^{s'} \Pi_{i \in \omega' \cup \Omega'} H_1(i)^{r_i} g_2^{\frac{-\beta_i}{\alpha_{i_d}}},$$
$$g^s = g_2^{\frac{-1}{\alpha_{i_d}}} g^{s'}$$

درپایان، جعل F یک اثر جعلی σ^* بر پیام m^* با پیش بینی Υ نتیجه می دهد. اگر داشته باشیم،

$$H_2(m) \neq g^{\beta_\delta} \text{ or } \hat{\Omega}^* \neq \Omega^*$$

مقدار Aنقض می شود. به عبارت دیگر، این مقدار معادله ی ثابت شده را تایید می کند، بدین معنا که،

$$\sigma^* = (\sigma_0^*, \{\sigma_i^*\}_{i \in \omega^* \cup \Omega^*}, \sigma_0^{*'})$$
$$= (g_2^x \Pi_{i \in \omega^* \cup \Omega^*} H_1(i)^{r_i} H_2(m^*)^s, \{g^{r_i}\}_{i \in \omega^* \cup \Omega^*}, g^s).$$

که درآن مقدار A بدین صورت محاسبه می شود،

$$g^{xy} = \sigma_0^* / \Pi_{i \in \omega^* \cup \Omega^*} \sigma_i^{*\beta_i} \sigma_0^{*'\beta_\delta},$$

زیرا داریم،

$$H_1(i) = g^{\beta_i} \text{ and } H_2(m^*) = g^{\beta_\delta}.$$

تئوری ۲: طرح ABSO ما تجهیزات ایمن اثبات پذیری را تایید می کند.

طرح پروف: ما نشان دادیم که برای هر سرور ابری S، احتمال موفقیت در λ وجود دارد، که درآن λ یک پارامتر امنیتی ست. ما می توانیم پروف اثبات پذیری را در دو جنبه تفکیک کنیم. در جنبه ی اول، اگر مقدار σ_{lah} به صورت صحیح تولید شود، داریم؛

$$\begin{aligned}\hat{Z} &= \frac{e(g,\sigma_0')}{\Pi_{i\in\omega^*\cup\Omega'}e(H_1(i),\sigma_i')}\\ &= \frac{e(g,\Pi_{i\in\omega'\cup\Omega'}(g_2^{t_i+q(i)}H_1(i)^{r_i})^{\Delta_{i,S}(0)}\Pi_{i\in\omega^*\cup\Omega'}H_1(i)^{r_i'})}{\Pi_{i\in\omega'\cup\Omega'}e(H_1(i),g^{r_i\Delta_{i,S}(0)}g^{r_i'})\Pi_{i\in\omega^*/\omega'}e(H_1(i),g^{r_i'})}\\ &= e(g,g_2^{x+\sum_{i\in\omega'\cup\Omega'}t_i\Delta_{i,S}(0)}) = Ze(g,V)\\ &= \hat{Z}'\end{aligned}$$

بدین ترتیب اگر σ_{lah} یک نیمه اثر اشتباه باشد، مقدار Z با احتمالات بیان شده حفظ نخواهد شد. دوم اینکه، اگر سرور ابری از اثبات پذیری اولیه با نیمه اثر اشتباه عبورکند، وقتیUنیمه اثر اشتباه σ_{lah} را برای تولید اثر ویژه ی پایانی σ بکار برد، اثبات پذیری σ به صورت معادله ی زیر خواهد بود؛

$$\frac{e(g,\sigma_0)}{\prod_{i\in\omega^*\cup\Omega'}e(H_1(i),\sigma_i)e(H_2(m),\sigma_0')} = Z.$$

بنابراین، صحت این نیمه اثر بازگشته از سرور ابری می تواند براساس اثبات پذیری ABS تایید شود.

۶. جمع بندی

دراین تحقیق، ما اثر ایمنی موثری را در طرح تولیدی برون سپاری ABS پیشنهاد داده ایم. در طرح ما، سرور ابری می تواند نیمه اثری برای کاربر در استفاده از کلید برون سپاری تولید کند. سپس، کاربر می تواند تولید اثر را هزینه ی اندک محاسبات، کامل کند. با کاربرد طرح ما، نشان کننده می تواند در مقایسه با طرح های سنتی رایج، بیش از ۹۰٪ از هزینه ی محاسبات را ذخیره کند. علاوه براین، ما تعریف طرح ABSO را به صورت معمولی ارائه کرده ایم و امنیت این طرح را تحت این مدل به ثبت رسانده ایم.

[1] J. Li, M.H. Au, W. Susilo, D. Xie, K. Ren, Attribute-based signature and its applications, in: Proceeding of the 5th ACM Symposium on Information, Computer and Communications Security, ASIACCS'10, ACM, 2010, pp. 60–69.
[2] J. Li, K. Kim, Hidden attribute-based signatures without anonymity revocation, Inform. Sci. 180 (9) (2010) 1681–1689. Elsevier.
[3] H. Maji, M. Prabhakaran, M. Rosulek, Attribute based signatures: achieving attribute privacy and collusion-resistance, 2008. Available at http://eprint.iacr.org/2008/328.
[4] A. Shamir, Identity-based cryptosystems and signature schemes, in: CRYPTO 1984, in: LNCS, vol. 196, 1984, pp. 47–53.
[5] J. Li, Q. Wang, C. Wang, K. Ren, Enhancing attribute-based encryption with attribute hierarchy, Mob. Netw. Appl. (MONET) 16 (5) (2011) 553–561. Springer-Verlag.
[6] H. Maji, M. Prabhakaran, M. Rosulek, Attribute based signatures, in: CT-RSA 2011, in: LNCS, vol. 6558, 2011, pp. 376–392.
[7] M. Armbrust, A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, M. Zaharia, Above the cloud: a Berkeley view of cloud computing, Berkeley University, 2009.
[8] M. Miller, Cloud Computing:WebBased Applications that Change theWayYou Work and Collaborate Online, Safari Books Online, LLC, 2008.
[9] X. Chen, J. Li, W. Susilo, Efficient fair conditional payments for outsourcing computations, IEEE Trans. Inf. Forensics Secur. (TIFS) 7 (6) (2012) 1687–1694.
[10] K. Bicakci, N. Baykal, Server assisted signature revisited, in: CT-RSA 2004, in: LNCS, vol. 2964, 2004, pp. 143–156.
[11] M. Jakobsson, S. Wetzel, Secure server-aided signature generation, in: PKC 2001, in: LNCS, vol. 1992, 2001, pp. 383–401.
[12] K.M. Chung, Y.T. Kailai, F.H. Liu, R. Rza, Memory delegation, in: CRYPTO 2011, in: LNCS, vol. 6841, 2011, pp. 151–168.
[13] C. Gentry, Fully homomorphic encryption using ideal lattices, in: STOC 2009, 2009, pp. 169–178.
[14] C. Genrty, S. Halevi, Implementing Gentry's fully-homomorphic encryption scheme, in: EUROCRYPTO 2011, in: LNCS, vol. 6632, 2011, pp. 129–148.
[15] J. Li, X. Chen, M. Li, J. Li, P.C. Lee, W. Lou, Secure deduplication with efficient and reliable convergent key management, IEEE Trans. Parallel Distrib. Syst. 25 (6) (2014) 1615–1625.
[16] R. Gennaro, C. Gentr, B. Parno, Non-interactive verifiable computing: outsourcing computation to untrusted worker, in: CRYPTO 2010, in: LNCS, vol. 6223, 2010, pp. 465–482.
[17] J. Li, X. Huang, J. Li, X. Chen, Y. Xiang, Securely outsourcing attribute-based encryption with checkability, IEEE Trans. Parallel Distrib. Syst. 25 (8) (2014) 2201–2210.
[18] J. Li, J. Li, X. Chen, C. Jia, W. Lou, Identity-based encryption with outsourced revocation in cloud computing, IEEE Trans. Comput. (2013) http://doi.ieeecomputersociety.org/10.1109/TC.2013.208.
[19] S. Goldwasser, Y.T. Kalai, G.N. Rothblum, Delegating computation: interactive proofs for muggles, in: STOC 2008, 2008, pp. 113–133.
[20] X. Chen, J. Li, X. Huang, J. Li, Y. Xiang, Secure outsourced attribute-based signatures, IEEE Trans. Parallel Distrib. Syst. (2013) http://doi.ieeecomputersociety.org/10.1109/TPDS.2013.2295809.
[21] X. Chen, J. Li, J. Ma, Q. Tang, W. Lou, New algorithms of outsourcing modular exponentiations, IEEE Trans. Parallel Distrib. Syst. (2013).
[22] X. Ma, J. Li, F. Zhang, Outsourcing computation of modular exponentiations in cloud computing, Cluster Comput. 16 (4) (2013) 787–796.
[23] D. Boneh, M. Franklin, Identity-based encryption from the weil pairing, in: CRYPTO 2001, in: LNCS, vol. 2139, 2001, pp. 213–229.
[24] A. Beimel, Secure schemes for secret sharing and key distribution,Technion, Haifa, Israel, 1996.