مجله علمی- پژوهشی "پدافند الکترونیکی و سایبری"
سال پنجم، شماره ۴، زمستان ۱۳۹۶، ص ۶۷-۵۵

یک طرح امضای مبتنی بر شناسه با تأییدکننده مشخص جدید به همراه کاربرد آن در خانه‌های هوشمند

محمد بهشتی آتشگاه[۱]، محمدرضا عارف[۲*]، مرتضی براری[۳]

۱- دانشجوی دکتری، دانشگاه صنعتی مالک اشتر ۲- استاد، دانشگاه صنعتی شریف ۳- استادیار، دانشگاه صنعتی مالک اشتر

(دریافت: ۹۶/۰۲/۰۱، پذیرش: ۹۶/۰۵/۰۱)

چکیده

در یک طرح امضاء با تأییدکننده مشخص قوی، امضاءکننده قادر می‌شود تا امضاء را برای یک گیرنده خاص صادر نماید؛ یعنی تنها گیرنده مشخصی که مدنظر امضاءکننده است می‌تواند اعتبار امضای صادرشده را بررسی نماید. البته باید طرح امضاء به گونه‌ای باشد که هیچ بخش سومی قادر به بررسی اعتبار امضاء نباشد یا به عبارت دیگر، تأییدکننده مشخص نتواند امضاء را به بخش سومی منتقل کند. در این مقاله، یک طرح امضای مبتنی بر شناسه جدید با تأییدکننده مشخص جدید ارائه می‌کنیم که در مدل سروش تصادفی و براساس فرض BDH دارای اثبات امنیتی است. طرح ارائه‌شده تمامی ملزومات امنیتی یک طرح امضای با تأییدکننده مشخص را برآورده می‌نماید. علاوه بر آن، طرح ارائه‌شده ویژگی محافظت از حریم خصوصی کاربر را برآورده ساخته و همچنین به لحاظ کارآمدی از حیث اندازه امضای خروجی و محاسبات لازم برای فازهای صادرشدن امضاء و تأیید آن با دیگر طرح‌های موجود قابل مقایسه بوده و جزو طرح‌های سبک‌وزن محسوب می‌شود. در نهایت نیز برخی از سناریوهای کاربردی طرح ارائه‌شده را در فضای اینترنت اشیاء معرفی می‌کنیم.

واژه‌های کلیدی: طرح امضای مبتنی بر شناسه، امضاء با تأییدکننده مشخص، اینترنت اشیاء، خانه هوشمند، رایانش ابری، اثبات امنیتی، زوج‌سازی دوخطی.

۱- مقدمه

اینترنت اشیاء[۱] (IoT) به صورت شبکه‌ای از دستگاه‌های (یا اشیای) متصل به هم طراحی می‌شود. در قالب تجسم امروزه، IoT شامل انواع مختلفی از دستگاه‌ها همانند حسگرها، فعال‌کننده‌ها، برچسب‌های RFID، تلفن‌های هوشمند یا سرورهای پشت‌خط می‌شود که به طور واضح در مقیاس اندازه[۲]، قابلیت[۳] و عاملیت[۴] باهم متفاوت می‌باشند. چالش اصلی این است که چطور چنین شبکه‌ای را تطبیق دهیم تا در محیط اینترنت مرسوم فعالیت نمایند [۱]. بر طبق پیش‌بینی Gartner [۲]، به استثنای رایانه‌های شخصی، گوشی‌های تلفن هوشمند و تبلت‌ها، فن‌آوری IoT تا سال ۲۰۲۰ بیش از ۲۶ میلیارد دستگاه را در بر خواهد داشت.

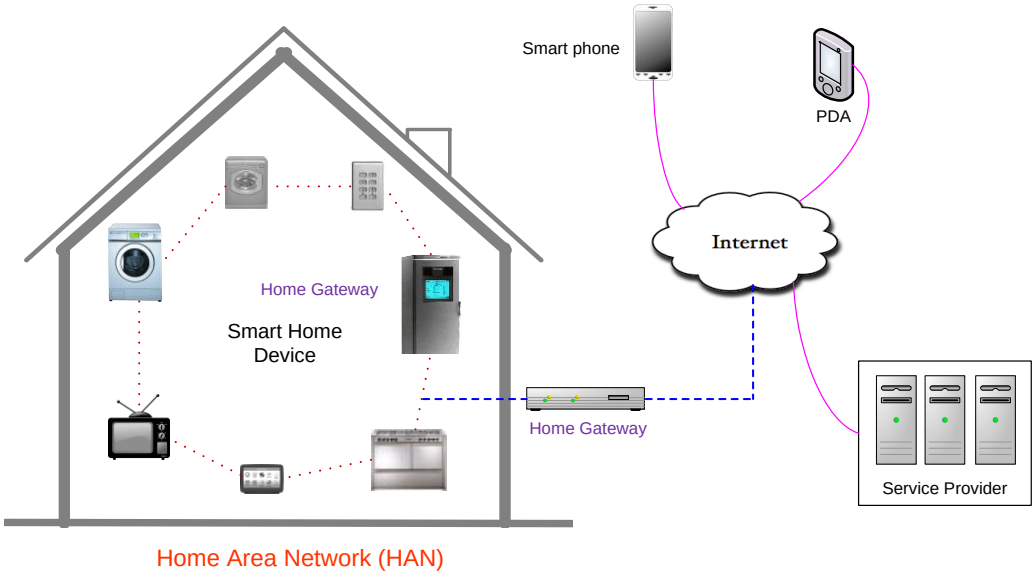
فناوری IoT پتانسیل‌های عظیمی برای توسعه کاربردها و برنامه‌های کاربردی هوشمند جدید را تقریباً در هر حوزه و زمینه‌ای دارا است. این امر به دلیل توانائی دوگانه IoT در اجرای سنجش‌های مناسب (برای مثال، فراهم‌نمودن امکان جمع‌آوری اطلاعات در مورد پدیده‌های طبیعی، پارامترهای پزشکی یا عادت‌های رفتاری کاربر) و همچنین، پیشنهاد و ارائه خدمات مناسب به آن‌ها است. صرف‌نظر از حوزه کاربرد، چنین کاربردهایی به بالابردن کیفیت زندگی روزمره انسان‌ها کمک نموده و تأثیر عمیقی بر روی اقتصاد و جامعه خواهند داشت. همچنین، این‌گونه از کاربردها، جنبه‌ها و منظرهای گوناگونی را پوشش خواهند داد: شخصی، اجتماعی، پزشکی، محیطی، منطقی. کاربردهای مختلف می‌تواند در سه حوزه اصلی تقسیم‌بندی شوند: حوزه صنعتی و تجاری، حوزه شهر هوشمند و حوزه بهداشت و رفاه [۳].

در این مقاله، کاربردی از امضای دیجیتال در خانه‌های هوشمند ارائه می‌شود که خانه‌های هوشمند جزو زیرمجموعه‌های حوزه شهر هوشمند محسوب می‌گردند. خانه‌ها/ ساختمان‌ها در یک شهر هوشمند با تعداد بسیار زیادی از حسگرها و دستگاه‌های هوشمند (نظیر دروازه‌های پهن‌باند، تلفن‌های همراه، لپ‌تاپ‌ها، PCها، تلویزیون، بلندگوها، تجهیزات، دوربین‌های جاسوسی،

* رایانامه نویسنده مسئول: Aref@sharif.edu

1- Internet of Things
2- Size
3- Capability
4 - Functionality

چراغ‌ها، پرده‌ها، ترموستات و غیره) تجهیز خواهند شد که با فن‌اوری‌های ارتباطی آمیخته شده و در ساختمان‌ها و خانه‌های مسکونی به‌کار گرفته می‌شوند و حوزه وسیعی را مورد پوشش قرار خواهند داد. سیستم‌های اتوماسیون خانگی به طور خاصی جذاب و جالب توجه هستند، چراکه امکان کنترل از راه دور هر چیزی را از طریق برنامه‌های کاربردی وب فراهم می‌سازند. برخی برنامه‌های کاربردی ساده‌ترین توانمندی‌های حوزه IoT را مورد استفاده قرار می‌دهند؛ از قبیل کاربردها برای اهداف امنیتی (همانند نظارت تصویری، تشخیص نفوذ، مدیریت دسترسی)، مدیریت و نگهداری واحد صنعتی (مثل تشخیص نقص، مدیریت/ نگهداری دارائی)، برای خودکارسازی (اتوماسیون) سرویس (همانند HVAC، نورپردازی، آبیاری) و برای سیستم‌های سرگرمی. انواع دیگری از کاربردها با شبکه هوشمند مجتمع می‌شوند و مصرف انرژی خانگی را بهینه می‌نمایند [۴]. برای نمونه، شبکه ناحیه خانگی (HAN) به تجهیزات اجازه می‌دهد تا با معیارهای سنجشی هوشمند تعامل نمایند تا در نهایت هزینه‌ها کاهش یابد، درحالی‌که بازدهی موردنیاز تضمین شود و کاهش نیابد. این امر می‌تواند از طریق سرویس‌هایی به‌دست آید که فعالیت‌های مختلف خانگی را (نظیر ماشین لباسشویی یا ظرفشویی) در قالب یک دوره پویا زمان‌بندی نمایند تا از مصرف هزینه مصرف بالا در ساعات اوج مصرف و حداکثر بار جلوگیری گردد.



شکل(۱): شمایی از معماری نمونه خانه هوشمند.

حتی ممکن است برنامه‌های کاربردی پیشرفته‌تر این امکان را فراهم نمایند که از تلفن‌های هوشمند به‌عنوان یک کنترل از راه دور یکپارچه استفاده شود که تمامی دستگاه‌ها/ تجهیزات خانگی را مدیریت می‌نماید (برای مثال برای خاموش‌کردن برخی تجهیزات بلافاصله بعد از اتمام کار) و همچنین رفتار کاربران را از طریق ردیابی تلفن‌های همراهشان کنترل می‌کند تا زندگی خانگی آن‌ها را راحت‌تر نماید [۵]. برای نمونه، از روی تحلیل اطلاعات جریان مصرفی، سامانه[1] می‌تواند زمان رسیدن شخص به خانه را یاد گرفته و بنابراین درب را به رویش بگشاید، چراغ‌ها را روشن کند و غیره. چنین عملیات خودکاری همواره توسط کاربر و در هر زمانی می‌تواند مجدداً زمان‌بندی شده و یا اساساً لغو شود.

۱-۱- طرح‌های امضای با تأییدکننده مشخص

در یک طرح امضای دیجیتال مرسوم، هر شخصی می‌تواند با استفاده از کلید عمومی امضاءکننده اعتبار امضای صادرشده را بررسی و در صورت صحت، آن را تأیید نماید. به هر حال، در برخی از کاربردها نیاز می‌شود تا امضای دیجیتال قابلیت این را داشته باشد که تنها توسط فرد خاصی قابل بررسی و تأیید باشد و این امکان خاصی است که توسط امضاءهای با تأییدکننده مشخص برآورده و فراهم می‌گردد. اولین بار Jakobsson و همکارانش در سال ۱۹۹۶ مفهوم طرح‌های امضای دیجیتال با تأییدکننده مشخص[2] را مطرح نمودند [۶]. در یک طرح، امضاء با تأییدکننده مشخص، امضاءکننده به گونه‌ای روی یک پیام امضاء صادر می‌کند که تنها گیرنده خاصی که موردنظر امضاءکننده است می‌تواند اعتبار آن را بررسی کند. در این حالت، تأییدکننده

1- System

2- Designated Verifier Signature Scheme

مشخص قادر نیست تا اعتبار این امضاء را برای یک بخش سومی اثبات نماید. این نوع از امضاءها کاربردهای بسیاری دارند ازجمله آن‌ها می‌توان به استفاده در پروتکل‌های رأی‌گیری الکترونیک، حق مالکیت نرم‌افزاری، تجارت الکترونیک اشاره نمود و حتی در انواع دیگری از طرح‌های امضاء نیز قابلیت استفاده دارند [۲۸].

در سال ۲۰۰۳، Saeednia و همکارانش مفهوم طرح امضاء با تأییدکننده مشخص قوی را مبتنی بر طرح Jakobsson و همکارانش ارائه نمودند که هیچ بخش سومی نمی‌توانست اعتبار یک امضاء با تأییدکننده مشخص را بررسی نماید و دلیل این امر آن بود که کلید خصوصی تأییدکننده مشخص در فاز تأیید امضاء مورد نیاز است [۷]. پس از طرح‌های Jakobsso و Saeednia، طرح‌های امضاء با تأییدکننده مشخص بسیاری ارائه شدند که اغلب این طرح‌ها مبتنی بر شناسه هستند [۱۲-۸]. این‌گونه امضاءها حاصل ترکیب حوزه رمزنگاری مبتنی بر شناسه و طرح‌های امضاء با تأییدکننده مشخص می‌باشند. این امضاءها دارای ملزومات امنیتی زیر هستند [۱۳]:

- غیرقابل جعل‌بودن[1]
- صحت طرح[2]
- قوی‌بودن[3] (به لحاظ تأییدکننده مشخص)
- انتقال ناپذیری[4]
- پنهان‌سازی منبع[5]

اولین طرح امضای با تأییدکننده مشخص مبتنی بر شناسه در سال ۲۰۰۴ توسط Susilo و همکارانش ارائه شد [۱۰]. Lipma و همکارانش نشان دادند که طرح Saeednia و همکارانش در مقابل حمله اعطاپذیری آسیب‌پذیر است [۱۴]. یعنی در طرح Saeednia و همکارانش، امضاءکننده می‌تواند قابلیت امضای خودش را بدون آن که کلید خصوصی‌اش فاش شود به یک بخش سومی اعطا نماید.

اخیراً Zhang و Mao یک طرح امضای مبتنی بر شناسه با تأییدکننده مشخص قوی ارائه نمودند. آن‌ها ادعا کردند که طرح‌شان ویژگی پنهان‌سازی منبع را برآورده می‌کند [۱۵]. در سال ۲۰۰۹، Kang و همکارانش نشان دادند که طرح Zhang و Mao نمی‌تواند ویژگی تأییدکننده مشخص قوی را برآورده سازد؛ چراکه هر کسی که بتواند یک امضاء را شنود کند متعاقباً می‌تواند امضای جدید را بدون دانستن کلید خصوصی تأییدکننده مشخص تأیید نماید. آن‌ها در مقاله خود بهبودی از طرح Zhang و Mao را ارائه نموده و ادعا کردند که طرح‌شان قوی و غیرقابل جعل است [۱۶]. تقریباً همزمان با طرح مذکور، Kang و همکارانش طرح دیگری را ارائه کردند که در آن یک طرح امضای مبتنی بر شناسه با تأییدکننده مشخص جدید به همراه اثبات امنیتی آن مطرح شده است [۱۷]. اما Lee و همکارانش نشان دادند که طرح [۱۵] Kang و همکارانش در برابر حمله جعل[6] آسیب‌پذیر بوده و ویژگی‌های غیرقابل اعطاءپذیر بودن و قوی بودن به لحاظ تأییدکنندگی را برآورده نمی‌سازد [۱۸]. علاوه‌بر آن، Du و Wen نشان دادن که طرح Kang و همکارانش به‌طور فراگیر و جهانی جعل‌پذیر می‌باشد [۱۹]. در سال ۲۰۱۱ میلادی و براساس طرح رمزنگاری سلسله‌مراتبی مبتنی بر شناسه Gentry-Silverberg، Huang و همکارانش یک طرح امضای با تأییدکننده مشخص قوی مبتنی بر شناسه ارائه نمودند. به هر حال، طرح ارائه‌شده توسط آن‌ها از مشکل بزرگی طول امضاء رنج برده و بنابراین برای کاربردهایی که با کمبود پهنای باند مواجه هستند، مناسب نمی‌باشد [۲۰].

در سال۲۰۱۳ میلادی، Duan و همکارانش یک طرح بهبودیافته با اندازه امضای کوتاه ارائه نمودند که براساس فرض CDH دارای اثبات امنیتی بود [۲۱]. طرح آن‌ها مبتنی بر طرح Kang و همکارانش [۱۷] ارائه شده بود. در سال ۲۰۱۴، Wang یک طرح امضای با تأییدکننده مشخص مبتنی بر شناسه با امضاءکننده مجاز ارائه داده که اندازه امضای آن قابل قبول بود [۲۲]. در سال ۲۰۱۵، Islam و Biswas یک طرح مبتنی بر زوج‌سازی با امکان بازیابی پیام به همراه اثبات امنیتی آن ارائه نمودند [۲۳]. به هر حال، Hu و همکارانش نشان دادند که طرح Islam و Biswas از ناحیه دو نوع حمله اعطاءپذیری آسیب‌پذیر می‌باشد. آن‌ها همچنین طرح جدیدی ارائه نمودندکه مشکلات طرح قبلی را برطرف می‌نمود [۲۴]. آن‌ها در سال ۲۰۱۷ همچنین طرح دیگری را با ویژگی انکارناپذیری ارئه نمودند [۲۵].

در این مقاله، می‌خواهیم یک طرح امضای با تأییدکننده مشخص مبتنی بر شناسه جدید ارائه نماییم که از زوج‌سازی‌های دوخطی استفاده نموده و در مدل سروش تصادفی، دارای امنیت قابل اثبات می‌باشد. در ادامه، طرح ارائه‌شده را از دو جنبه کارآمدبودن و امنیت با طرح‌های موجود دیگر مقایسه می‌نماییم. در نهایت نیز سناریوهای کاربردی از طرح ارائه‌شده را در فضای اینترنت اشیاء (IoT) ارائه می‌کنیم.

ادامه این مقاله بدین صورت سازماندهی شده است: در بخش ۲، یکسری از مقدمات لازم را مرور می‌کنیم. در بخش۳، طرح

1- Unforgeability
2- Correctness of scheme
3- Strongness
4- Non-transferability
5- Source hiding
6- Forgery attack

امضای مبتنی بر شناسه جدید با تأییدکننده مشخص را ارائه می‌نماییم. در بخش ۴، به ارزیابی امنیتی و کارآمدی طرح ارائه‌شده می‌پردازیم. در بخش ۵، نمونه‌ای از کاربرد طرح امضای ارائه‌شده در خانه‌های هوشمند ارائه نموده و در نهایت در بخش ۶، نتیجه‌گیری می‌نماییم.

۲- مفاهیم اساسی

در این بخش برخی از مقدمات موردنیاز برای طرح پیشنهادشده همانند زوج‌سازی دوخطی، فرض‌های پیچیدگی، مدل رسمی برای یک طرح امضای مبتنی بر شناسه با تأییدکننده مشخص، انتقال‌پذیری و حریم‌خصوصی امضاکننده را معرفی و مرور می‌نماییم.

۲-۱- زوج‌سازی‌های دوخطی

در نظر بگیرید که G_1 یک گروه جمعی دوری از مرتبه اول q، G_2 یک گروه ضربی دوری از همان مرتبه و همچنین P مولدی از گروه G_1 باشد. فرض کنید $e: G_1 \times G_1 \to G_2$ یک نگاشت دوخطی با ویژگی‌های زیر باشد:

۱- دوخطی بودن[1]: به ازای تمامی $a, b \in \mathbb{Z}_q^*$ و $P, Q \in G_1$ داشته باشیم $e(aP, bQ) = e(abP, Q) = e(P, abQ) = e(P, Q)^{ab}$.

۲- زوال‌ناپذیری[2]: $P \in G_1$ و $Q \in G_1$ به گونه‌ای وجود دارند که $e(P, Q) \neq 1$.

۳- محاسبه‌پذیری[3]: به ازای تمامی $P, Q \in G_1$، الگوریتم کارآمدی برای محاسبه $e(P, Q)$ وجود دارد.

یک مولد پارامتر زوج‌سازی دوخطی به صورت یک الگوریتم زمان چندجمله‌ای احتمالی تعریف می‌شود که ورودی آن پارامتر امنیتی k و خروجی آن چندتایی تصادفی (p, G_1, G_2, e, P) می‌باشد که در آن p عدد اول با اندازه k است.

۲-۲- فرض‌های پیچیدگی

تعریف ۱: (مسأله دیفی- هلمن دوخطی (BDH)). پارامتر تصادفی $P \in G_1$ و همچنین aP, bP و cP (که مقادیر تصادفی a, b و c از گروه $\mathbb{Z}_q$ انتخاب شده‌اند) معلوم بوده و هدف یافتن مقدار $e(P, P)^{abc}$ می‌باشد.

تعریف ۲: (فرض BDH). اگر $\mathcal{G}$ یک مولد پارامتر BDH باشد، امتیاز مهاجم $\mathcal{A}$ به صورت $\mathrm{Adv}_{\mathcal{G}}(\mathcal{A})$ تعریف می‌شود که در واقع بیان‌گر احتمالی است که الگوریتم $\mathcal{A}$ در حل مسأله BDH دارد. مهاجم $\mathcal{A}$ مقادیر $G_1, G_2, e, P, aP, bP, cP$ را در اختیار دارد و مقدار $e(P, P)^{abc}$ را به‌عنوان خروجی می‌دهد که G_1, G_2, e خروجی $\mathcal{G}$ برای پارامتر امنیتی k بوده، P مولدی تصادفی از G_1 و c, b, a عناصر تصادفی از گروه $\mathbb{Z}_q$ هستند. فرض BDH این است که $Adv_{\mathcal{G}}(\mathcal{A})$ به ازای تمامی الگوریتم‌های کارآمد $\mathcal{A}$ ناچیز است.

۲-۳- مدل رسمی یک طرح ID-SDVS

در این زیربخش قصد داریم تعریف و همچنین مدل رسمی یک طرح ID-SDVS را توصیف نماییم. همان‌طورکه در [۱۴ و ۱۷-۱۶] بیان شده است، یک طرح ID-SDVS شامل پنج فاز زیر می‌باشد:

- **برپایی:** این فاز در حقیقت یک الگوریتم زمان چندجمله‌ای است که یک پارامتر امنیتی k را به عنوان ورودی خود گرفته و پارامترهای سامانه به اضافه کلید محرمانه اصلی را به عنوان خروجی می‌دهد.

- **تولید کلید:** یک الگوریتم زمان چندجمله‌ای که از پارامترهای سامانه، کلید محرمانه اصلی و یک رشته دلخواه $ID_i \in \{0,1\}^*$ به عنوان ورودی استفاده نموده و یک کلید خصوصی همانند S_{ID_i} را به عنوان خروجی ارایه می‌دهد. در این‌جا، ID_i شناسه شخص مورد نظر می‌باشد و به عنوان کلید عمومی او مورد استفاده قرار می‌گیرد.

- **تولید امضا:** الگوریتم زمان چندجمله‌ای که پارامترهای سامانه، کلید خصوصی شخص امضاءکننده ID_A (یعنی S_{ID_A})، پیام m و کلید عمومی تأییدکننده مشخص ID_B (یعنی Q_{ID_B}) را به عنوان ورودی خود گرفته و یک امضای σ را بر روی پیام m خروجی می‌نماید.

- **تأیید امضاء:** یک الگوریتم زمان چندجمله‌ای معین می‌باشد که پارامترهای سامانه، پیام m، شناسه امضاءکننده (ID_A)، شناسه تأییدکننده مشخص (ID_B) و کلید خصوصی تأییدکننده مشخص (S_{ID_B}) و همچنین امضای σ را به عنوان ورودی گرفته و سپس، اگر امضاء معتبر بود خروجی "پذیرش" می‌دهد و درحالتی که امضاء معتبر نباشد، خروجی "رد" می‌دهد.

- **شبیه‌سازی امضاء:** این فاز، یک الگوریتم زمان چندجمله‌ای احتمالاتی است که به منظور تولید یک رونوشت از امضاء دقیقاً یا همان توزیع یکنواخت و توسط تأییدکننده مشخص اجرا می‌گردد که این رونوشت از امضای واقعی که توسط امضاءکننده صورت گرفته است، تمایزناپذیر می‌باشد.

۲-۴- انتقال‌ناپذیری

این ویژگی بدین معنی است که به ازای یک زوج متن/ امضای

1- Bilinearity
2- Non-degeneracy
3- Computability

(σ, U, m) که توسط تأییدکننده مشخص قابل بررسی و تأیید می‌باشد، هیچ تمایزگر احتمالاتی زمان چندجمله‌ای نمی‌تواند بگوید که این امضاء توسط امضاءکننده صورت گرفته است و یا این‌که تأییدکننده مشخص آن را تولید نموده است. تعریف رسمی این ویژگی به صورت زیر می‌باشد:

تعریف ۳ (انتقال‌ناپذیری): یک طرح امضای با تأییدکننده مشخص انتقال‌ناپذیر نامیده می‌شود اگر امضای تولیدشده توسط امضاءکننده به لحاظ محاسباتی از امضای تولیدشده توسط تأییدکننده مشخص تمایزناپذیر باشد.

۲-۵- حریم خصوصی امضاءکننده

حریم خصوصی امضاءکننده بیان می‌کند که بدو در اختیارداشتن کلید خصوصی تأییدکننده مشخص، هیچ کس نمی‌تواند امضاءهای تولیدشده توسط دو امضاءکننده S_0 و S_1 را که برای تأییدکننده مشخص DV صورت داده شده‌اند، تمایز دهد. تعریف رسمی حریم خصوصی امضاءکننده توسط یک بازی مدل می‌شود که این بازی میان چالش‌گر $\mathcal{C}$ و یک تمایزگر $\mathcal{D}$ به صورت زیر اجرا می‌شود [۲۶]:

۱- $\mathcal{C}$ زوج‌کلید امضاءکنندگان S_0 و S_1 و تأییدکننده مشخص DV یعنی (Q_{S_0}, S_{S_0})، (Q_{S_1}, S_{S_1}) و (Q_{DV}, S_{DV}) را تولید نموده و تمایزگر $\mathcal{D}$ را بر روی ورودی $(Q_{S_0}, Q_{S_1}, Q_{DV})$ فراخوانی می‌کند.

۲- $\mathcal{D}$ همانند آن‌چه که در اثبات امنیتی غیرقابل جعل‌بودن داریم، پرسش‌های زمان چندجمله‌ای انجام می‌دهد البته با یک تفاوت؛ و آن‌که سروش‌ها یک ورودی اضافی $d \in \{0,1\}$ را می‌گیرند که این ورودی اضافی نشان‌دهنده شناسه امضاءکننده می‌باشد. بنابراین، سروش‌ها عمل تولید/ تأیید امضاءها را با توجه به کلیدهای Q_{DV}, Q_{S_d} انجام می‌دهند.

۳- $\mathcal{D}$ یک پیام M^* خروجی می‌دهد. سپس، $\mathcal{C}$ یک سکه $b \in \{0,1\}$ پرتاب کرده، امضای چالشی (σ^*, U^*, M^*) را محاسبه نموده و این امضاء را به تمایزگر $\mathcal{D}$ باز می‌گرداند.

همانند گام ۲، تمایزگر $\mathcal{D}$ به انجام پرسش‌های خود ادامه می‌دهد؛ با این تفاوت که به ازای $d \in \{0,1\}$ و ورودی (d, σ^*, U^*, M^*) از سروش $\mathcal{O}_{Ver}$ پرسش نمی‌کند. در نهایت، تمایزگر یک بیت b' را به عنوان خروجی ارائه می‌دهد و درصورتی که $b' = b$ باشد بازی را می‌برد.

تعریف ۴ (حریم خصوصی امضاءکننده): یک طرح امضای با تأییدکننده مشخص ویژگی حفظ حریم‌خصوصی امضاءکننده را دارد اگر هیچ مهاجمی همانند $\mathcal{D}$ که در زمان حداکثر t اجرا می‌شود و می‌تواند حداکثر تعداد q_{Sign} پرسش از سروش $\mathcal{O}_{Sign}$، حداکثر تعداد q_{Sim} پرسش از سروش $\mathcal{O}_{Sim}$ و حداکثر تعداد q_{Ver} پرسش از سروش $\mathcal{O}_{Ver}$ انجام دهد و بتواند بازی فوق را با احتمالی بیشتر از $1/2$ (یعنی با مزیت ε) ببرد وجود نداشته باشد.

۳- طرح امضای مبتنی بر شناسه با تأییدکننده مشخص جدید

در این بخش، طرح امضای با تأییدکننده مشخص مبتنی بر شناسه و همچنین طرح امضای وکالتی با تأییدکننده مشخص مبتنی بر شناسه جدید خود را ارایه می‌نماییم.

۳-۱- طرح امضای با تأییدکننده مشخص جدید

طرح ارائه‌شده دارای پنج فاز است که به صورت زیر می‌باشند:

- **برپایی:** در این فاز، PKG یک گروه دیفی- هلمن گپ[1] G_1 از مرتبه‌ی q و یک گروه ضربی G_2 از همان مرتبه و یک زوج‌سازی دوخطی $e: G_1 \times G_1 \to G_2$ را به همراه یک مولد دلخواه $P \in G_1$ انتخاب می‌کند. سپس، PKG یک مقدار تصادفی $s \in \mathbb{Z}_q^*$ را به عنوان کلید اصلی انتخاب و کلید عمومی متناظر $P_{\text{pub}} = sP$ را محاسبه می‌نماید. $H_1(\cdot)$ و $H_2(\cdot)$ دو تابع چکیده‌ساز رمزنگاری هستند که $H_1: \{0,1\}^* \to G_1$ و $H_2: \{0,1\}^* \to \mathbb{Z}_q^*$. پارامترهای سامانه $(G_1, G_2, P, P_{\text{pub}}, H_1, H_2, e, q)$ بوده و کلید محرمانه اصلی نیز s می‌باشد.

- **تولید کلید:** به ازای یک شناسه معلوم ID، PKG عبارت $S_{ID} = sH_1(ID)$ را محاسبه نموده و به کاربری با شناسه ID ارسال می‌کند. یادآور می‌شویم که کلید عمومی کاربر با شناسه ID برابر $Q_{ID} = H_1(ID)$ می‌باشد. در این سناریو، آلیس با شناسه ID_A کلید عمومی $Q_{ID_A} = H_1(ID_A)$ و کلید خصوصی $S_{ID_A} = sQ_{ID_A}$ را دارد. باب نیز با شناسه ID_B کلید عمومی $Q_{ID_B} = H_1(ID_B)$ و کلید خصوصی $S_{ID_B} = sQ_{ID_B}$ را دارد.

- **امضاء:** برای امضای پیام m برای باب، آلیس عدد تصادفی $r \in \mathbb{Z}_q^*$ را انتخاب کرده و عبارت زیر را محاسبه می‌کند:

$$U = rQ_{ID_A}, h = H_2(m, U) \quad (۱)$$

$$T = H_2(m, U)S_{ID_A} = hS_{ID_A} \quad (۲)$$

$$\sigma = e(T, Q_{ID_B}) \quad (۳)$$

آلیس دوتائی (σ, U) را به عنوان امضای روی پیام m به گیرنده مشخص یعنی باب می‌فرستد.

- **تأیید امضاء:** با دریافت امضای (σ, U)، باب ابتدا با استفاده از

1- Gap Diffie-Hellman

U عبارت $h = H_2(m, U)$ را محاسبه کرده و سپس امضاء را می‌پذیرد اگر و تنها اگر شرط زیر برقرار باشد:

$$\sigma = e(T, Q_{ID_B}) = e(hQ_{ID_A}, S_{ID_B}) \quad (۴)$$

- **شبیه‌سازی رونوشت:** باب عدد تصادفی $r' \in \mathbb{Z}_q^*$ را انتخاب کرده و به صورت زیر محاسبه می‌کند:

$$U' = r'Q_{ID_A}, h' = H_2(m, U') \quad (۵)$$

$$T' = H_2(m, U')S_{ID_A} = h'S_{ID_A} \quad (۶)$$

$$\sigma' = e(T', Q_{ID_B}) \quad (۷)$$

واضح است که امضای (σ', U') رابطه تأیید را برآورده می‌نماید.

۴- ارزیابی امنیتی و کارآیی طرح ارائه‌شده

در این بخش قصد داریم تا در قالب چهار زیربخش، چهار نوع ارزیابی و تحلیل از طرح نشان داده‌شده ارایه نماییم. در ابتدا، نشان می‌دهیم که طرح ارایه‌شده غیرقابل جعل می‌باشد. بعد از آن، بر روی سایر ملزومات امنیتی مربوطه خواهیم پرداخت. سپس، تحلیل حریم‌خصوصی طرح ارایه‌شده را مورد بررسی قرار داده و در نهایت، ارزیابی کارآیی طرح مورد نظر را بررسی می‌نماییم.

۴-۱- غیرقابل جعل‌بودن

اکنون در قالب قضیه زیر نشان می‌دهیم که طرح ارائه‌شده در مقابل حمله جعل فراگیر مقاوم است.

قضیه ۱. اگر مهاجمی همانند $\mathcal{A}$ وجود داشته باشد که بتواند طرح ارائه‌شده را بشکند، آن‌گاه شبیه‌سازی همانند $\mathcal{B}$ وجود خواهد داشت که قادر است تا یک نمونه از مسأله BDH را با یک احتمال غیرقابل صرف‌نظر حل نماید. در حقیقت، هدف الگوریتم $\mathcal{B}$ این است که با دانستن (P, aP, bP, cP) به ازای $a, b, c \in \mathbb{Z}_q$ نامعلوم، مقدار $e(P, P)^{abc}$ را محاسبه کند.

اثبات: می‌توانیم کلید عمومی P_{pub} را به صورت $P_{pub} = bP$ بنویسیم. علاوه‌بر آن، به‌دلیل آن‌که تابع چکیده‌ساز H_1 به‌صورت یک سروش تصادفی درنظر گرفته می‌شود، می‌توان نوشت:

$$Q_{ID_A} = aP\ , Q_{ID_B} = cP \quad (۸)$$

- **برپایی:** در این مرحله، شبیه‌ساز $\mathcal{B}$ پارامترهای سیستم یعنی $(G_1, G_2, e, P_{\text{pub}}, H_1, H_2)$ را به مهاجم $\mathcal{A}$ می‌فرستد که در آن توابع چکیده‌ساز H_1 و H_2 به صورت سروش‌های تصادفی در نظر گرفته می‌شوند.
- **پرسمان‌ها از H_1:** زمانی‌که مهاجم $\mathcal{A}$ یک پرسمان ID_i را از سروش تصادفی H_1 انجام می‌دهد، شبیه‌ساز $\mathcal{B}$ عبارت (ID_i, R_i) را در فهرست-H_1 جستجو می‌کند. توجه کنید که این لیست و فهرست از ابتدا خالی است. سپس به نحوه زیر به مهاجم $\mathcal{A}$ پاسخ می‌دهد:

$$R_i = H_1(ID_i) = \begin{cases} aP, & \text{if } ID_i = ID_A \\ cP, & \text{if } ID_i = ID_B \\ r_iP, & \text{if } ID_i \neq ID_A, ID_B, r_i \in_R \mathbb{Z}_q \end{cases} \quad (۹)$$

- **پرسمان‌های استخراج:** زمانی‌که $\mathcal{A}$ بر روی شناسه ID_i پرسمان می‌کند، $\mathcal{B}$ شناسه ID_i را در فهرست-استخراج جستجو می‌کند. اگر $i \neq A$ یا $i \neq B$ باشد، شبیه‌ساز $\mathcal{B}$ عبارت $S_i = r_i(bP)$ را به عنوان کلید خصوصی متناظر با $H_1(ID_i)$ و برای شناسه ID_i محاسبه و به مهاجم باز می‌گرداند. در غیراین صورت، شبیه‌ساز $\mathcal{B}$ بازی و روند را متوقف می‌کند.
- **پرسمان‌ها از H_2:** زمانی‌که مهاجم $\mathcal{A}$ از سروش تصادفی H_2 با عبارت (m_i, t_i) پرسمان می‌کند، $\mathcal{B}$ عبارت متناظر را در فهرست-H_2 جستجو می‌کند و اگر چنین مقداری در آن تعریف شده باشد، مقدار c_i را به عنوان خروجی می‌دهد. در غیر این صورت، $\mathcal{B}$ یک عدد تصادفی $c_i \in \mathbb{Z}_q^*$ را انتخاب کرده و به عنوان خروجی چکیده‌سازی (m_i, t_i) به مهاجم $\mathcal{A}$ می‌دهد. همچنین، عبارت (m_i, t_i, c_i) را در فهرست-H_2 اضافه می‌کند.
- **پرسمان‌های تولید امضاء:** فرض کنید که $\mathcal{A}$ پرسمان‌های q_s را از الگوریتم امضاء انجام می‌دهد. اگر مهاجم $\mathcal{A}$ امضاء روی پیام m_i و با شناسه امضاءکننده ID_i و شناسه تأییدکننده ID_j پرسمان نماید، شبیه‌ساز $\mathcal{B}$ به صورت زیر پاسخ می‌دهد:

اگر $i \neq A$ و $i \neq B$ باشد، آن‌گاه $\mathcal{B}$ کلید خصوصی امضاءکننده با شناسه ID_i را به صورت:

$$r_i \in \mathbb{Z}_q^* \rightarrow U_i = r_iQ_{ID_i}, h = H_2(m_i, U_i) \quad (۱۰)$$

$$T_i = H_2(m_i, U_i)S_{ID_i} = hS_{ID_i} \quad (۱۱)$$

$$\sigma_i = e\left(T_i, Q_{ID_j}\right) \quad (۱۲)$$

اگر $j \neq B$ و $j \neq A$ باشد، آن‌گاه $\mathcal{B}$ کلید خصوصی تأییدکننده مشخص موردنظر با شناسه ID_i را به صورت $S_{ID_j} = r_jP_{\text{pub}} =$

$r_j bP$ محاسبه می‌کند. علاوه بر آن، عبارات زیر را محاسبه می‌کند:

$$r_i \in \mathbb{Z}_q^* \rightarrow U_i = r_i Q_{ID_i}, h = H_2(m_i, U_i) \quad (۱۳)$$

$$T_i = H_2(m_i, U_i) S_{ID_j} = h S_{ID_j} \quad (۱۴)$$

$$\sigma_i = e\left(T_i, Q_{ID_i}\right) \quad (۱۵)$$

در غیر این صورت، روند بازی متوقف می‌شود.

سرانجام $\mathcal{B}$ عبارت (σ_i, U_i) را به‌عنوان امضای روی پیام m_i با شناسه امضاءکننده ID_i و شناسه تأییدکننده ID_j به مهاجم $\mathcal{A}$ باز می‌گرداند.

پرسمان‌های تأیید امضاء: زمانی‌که مهاجم $\mathcal{A}$ یک پرسمان تأیید بر روی امضای (σ_i, U_i) انجام می‌دهد که ID_i شناسه امضاءکننده و ID_j شناسه تأییدکننده می‌باشد، $\mathcal{B}$ بررسی می‌کند که آیا $(i,j) = (A,B)$ یا $(i,j) = (B,A)$ است یا نه. اگر برقرار بود آن‌گاه $\mathcal{B}$ بازی را خاتمه می‌دهد. در غیر این صورت، $\mathcal{B}$ کلید خصوصی تأییدکننده مشخص را به صورت $S_{ID_j} = r_j P_{\text{pub}}$ را برای تأیید اعتبار امضاء با استفاده از الگوریتم تأیید محاسبه می‌کند. سرانجام مهاجم $\mathcal{A}$ یک امضای معتبر (σ^*, U^*) را بر روی پیام m^* با شناسه امضاءکننده ID_i و تأییدکننده مشخص $\mathrm{ID_j}$ با احتمال غیرقابل صرف‌نظر ε به عنوان خروجی می‌دهد. اگر آیا $(i,j) = (A,B)$ یا $(i,j) = (B,A)$ باشد، آن‌گاه $\mathcal{B}$ عبارت (σ^*, U^*) را به عنوان خروجی می‌دهد. به دلیل این‌که حداکثر به تعداد q_s پرسمان می‌تواند صورت پذیرد و در این تعداد پرسمان، تعداد $q_s(q_s - 1)$ زوج از امضاءکننده و تأییدکننده مشخص وجود دارد، احتمال آن‌که $\mathcal{A}$ یک امضای معتبر (σ^*, U^*) با $(i,j) = (A,B)$ یا $(i,j) = (B,A)$ تولید کند برابر است با $\frac{\varepsilon}{q_s(q_s-1)}$.

و این امضای خروجی باید شرط زیر را برآورده کند:

$$\sigma^* = e\left(h^* Q_{ID_A}, S_{ID_B}\right) \quad (۱۶)$$

بنابراین، می‌توانیم مقدار $e(P,P)^{abc}$ را از روی امضای جعل‌شده با احتمال $\frac{2\varepsilon}{q_s(q_s-1)}$ به دست آوریم:

$$\begin{aligned} e(P,P)^{abc} &= e(aP, bcP) \\ &= e\left(aP, bQ_{ID_B}\right) \\ &= e\left(Q_{ID_A}, S_{ID_B}\right) \\ &= (\sigma^*)^{h^{*-1}} \end{aligned} \quad (۱۷)$$

که این با فرض BDH در تناقض است. بنابراین طرح ارائه‌شده غیرقابل جعل است.

قضیه ۲ (انتقال‌ناپذیری IDDVSS): طرح IDVS ارایه‌شده در مقابل حمله متن انتخابی تطبیقی و تمایزگر PPT انتقال‌ناپذیر است.

اثبات: باید نشان دهیم که طرح فوق انتقال‌ناپذیر است.

برپایی: چالش‌گر $\mathcal{C}$ پارامترهای سامانه و کلیدهای عمومی را به صورت زیر تنظیم می‌نماید:

۱- $\mathcal{C}$ زوج کلیدهای عمومی/خصوصی امضاءکننده را به ترتیب $Q_{ID_A} = H_1(ID_A)$ و $S_{ID_A} = sQ_{ID_A}$ قرار می‌دهد.

۲- چالش‌گر $\mathcal{C}$ لیست-L را تشکیل داده و تمامی کلیدهای عمومی و خصوصی تأییدکنندگان را در آن ذخیره می‌نماید. به منظور تولید کلیدهای عمومی/ خصوصی تأییدکننده i-اُم، $\mathcal{C}$ مقادیر $Q_{ID_i} = H_1(ID_i)$ و $S_{ID_i} = sQ_{ID_i}$ را محاسبه می‌کند. سپس $\mathcal{C}$ مقادیر $\left(Q_{ID_i}, S_{ID_i}\right)$ را به لیست-L اضافه می‌نماید.

۳- $\mathcal{C}$ کلید عمومی امضاءکننده (Q_{ID_A})، کلیدهای عمومی تأییدکنندگان (Q_{ID_i}) و پارامترهای مشترک را به تمایزگر $\mathcal{D}$ باز می‌گرداند. توجه داشته باشید که از منظر تمایزگر، تمامی پارامترها توزیع یکسان داشته و دقیقاً همانند ساختار واقعی می‌باشند.

مرحله اول: $\mathcal{D}$ می‌تواند از سروش‌های امضاء، تأییـد، اسـتخراج و شبیه‌سازی پرسش نماید.

۱- به‌دلیل آن‌که $\mathcal{C}$ کلیدهای خصوصی امضاءکنندگان و تأییدکننده‌ها را می‌داند، پس می‌تواند الگوریتم امضاء، الگوریتم تأیید، الگوریتم استخراج و الگوریتم شبیه‌سازی را اجرا نماید تا بتواند به پرسش‌های متناظر تمایزگر $\mathcal{D}$ پاسخ دهد.

۲- پرسش استخراج: فرض کنید که $\mathcal{D}$ کلید خصوصی متناظر با کلید عمومی Q_{ID_i} را درخواست نماید. چالش‌گر $\mathcal{C}$ لیست-L را بررسی نموده و کلید خصوصی متناظر S_{ID_i} را به تمایزگر $\mathcal{D}$ باز می‌گرداند.

چالش: در مرحله اول، تمایزگر $\mathcal{D}$ بر روی یک چالش و یک کلید عمومی تأییدکننده تصمیم می‌گیرد؛ به گونه‌ای که تاکنون چنین درخواستی را ثبت نکرده است. در پاسخ به این درخواست، چالش‌گر $\mathcal{C}$ یک سکه تصادفی $c \in \{0,1\}$ پرتاب می‌کند. اگر $c = 1$ بود، $\mathcal{C}$ الگوریتم امضاء را اجرا نموده و امضای σ را به تمایزگر $\mathcal{D}$ باز می‌گرداند. در غیر این‌صورت، $\mathcal{C}$ الگوریتم شبیه‌سازی

را اجرا نموده و امضای $\sigma\hat{\sigma}$ را به تمایزگر $\mathcal{D}$ باز می‌گرداند.

حدس: در نهایت، تمایزگر $\mathcal{D}$ حدس خود یعنی $c' \in \{0,1\}$ را خروجی می‌کند. اکنون، نشان می‌دهیم امضای $\hat{\sigma}$ که توسط الگوریتم شبیه‌سازی شده است از امضای $\hat{\sigma}$ که توسط الگوریتم امضاء تولید شده است، تمایزناپذیر می‌باشد.

$$\begin{aligned}\hat{\sigma} &= e(S_{ID_A}, \hat{h}Q_{ID_B}) = e(sQ_{ID_A}, \hat{h}Q_{ID_B}) \\ &= e(\hat{h}sQ_{ID_A}, Q_{ID_B}) = e(\hat{h}Q_{ID_A}, sQ_{ID_B}) \\ &= e(\hat{h}sQ_{ID_A}, S_{ID_B})\end{aligned} \quad (۱۸)$$

و

$$\sigma = e(hQ_{ID_A}, S_{ID_B}) \quad (۱۹)$$

سپس داریم $\Pr[\sigma = \hat{\sigma}] = 1$ که به این معنی است خروجی الگوریتم شبیه‌سازی کاملاً از امضای σ که توسط الگوریتم امضاء تولید شده است تمایزناپذیر است.

۴-۲- ملزومات امنیتی دیگر

در این قسمت نشان می‌دهیم که طرح امضای مبتنی بر شناسه با تأییدکننده مشخص ارائه‌شده تمامی ملزومات امنیتی یک طرح DVS مبتنی بر شناسه را برآورده می‌کند:

۱- صحت طرح: روابط زیر صحت رابطه تأیید امضای ارائه‌شده را اثبات می‌کند.

$$\begin{aligned}\sigma &= e(T, Q_{ID_B}) = e(H_2(m,U)S_{ID_A}, Q_{ID_B}) \\ &= e(hsQ_{ID_A}, Q_{ID_B}) = e(hQ_{ID_A}, sQ_{ID_B}) \\ &= e(hQ_{ID_A}, S_{ID_B})\end{aligned} \quad (۲۰)$$

۲- تأییدکنندگی قوی: در طرح ارائه‌شده، تأییدکننده (B) مجبور است تا در فاز تأیید از کلید خصوصیش یعنی $S_{ID_B} = sQ_{ID_B}$ استفاده نماید. همچنین در صورت شنود یک امضاء، هیچ اطلاعات مفیدی جهت تأیید امضای بعد در اختیار شنودگر قرار نمی‌گیرد. بنابراین، طرح ارائه‌شده یک طرح با تأییدکننده قوی است.

۳- انتقال‌ناپذیری: منظور از انتقال‌ناپذیری این است که تأییدکننده مشخص نتواند اعتبار امضایی را که توسط امضاءکننده صادرشده برای یک بخش سومی اثبات نماید. فرض کنید که $(\hat{\sigma}, \hat{U})$ امضایی باشد که از بین تمامی امضاءهای معتبری که A برای B صادر کرده، انتخاب شده است.

به‌دلیل آن‌که امضای (σ, U) با استفاده از یک مقدار تصادفی انتخاب‌شده r از $\mathbb{Z}_q^*$ تولیدشده، پس احتمال آن‌که این دو امضاء یکی باشند یعنی $\Pr[(\sigma, U) = (\hat{\sigma}, \hat{U})]$ برابر است با $\frac{1}{q-1}$. همچنین، به علت آن‌که امضای (σ', U') با استفاده از یک مقدار تصادفی انتخاب‌شده r' از $\mathbb{Z}_q^*$ تولیدشده، پس احتمال $\Pr[(\sigma', U') = (\hat{\sigma}, \hat{U})]$ دقیقاً برابر با $\frac{1}{q-1}$ است. این بدان معنی است که رونوشت‌های شبیه‌سازی‌شده توسط باب از امضاهایی که توسط A صادر شده، تمایزناپذیر می‌باشد و بنابراین، طرح ارائه‌شده خاصیت انتقال‌ناپذیری را برآورده می‌نماید.

۴- پنهان‌سازی منبع: در طرح ارائه‌شده، یک بخش سوم حتی با داشتن کلیدهای خصوصی امضاءکننده و تأییدکننده مشخص باز هم نمی‌تواند تعیین کند که یک امضاء توسط امضاءکننده یا توسط تأییدکننده تولید شده است. رابطه زیر به وضوح این ادعا را نشان می‌دهد:

$$\sigma = e(hS_{ID_A}, Q_{ID_B}) = e(hQ_{ID_B}, Q_{ID_A}) \quad (۲۱)$$

که در رابطه فوق، $h = H_2(m,U)$. بنابراین، طرح ارائه‌شده خاصیت پنهان‌کردن منبع را برآورده می‌کند.

۴-۳- تحلیل حریم خصوصی

در این زیربخش، قصد داریم نشان دهیم که طرح ID-DVS ارائه‌شده از حریم خصوصی شناسه امضاءکننده محافظت می‌نماید یا به عبارت دیگر، ویژگی حفاظت از حریم خصوصی شناسه امضاءکننده را برآورده می‌نماید. این ویژگی در قالب قضیه زیر اثبات می‌گردد.

قضیه ۳- اگر مهاجمی همانند $\mathcal{A}$ وجود داشته باشد که بتواند طرح ارایه‌شده را بشکند آن‌گاه شبیه‌سازی همانند $\mathcal{B}$ وجود دارد که می‌تواند مسأله BDH را با احتمالی غیرقابل صرف‌نظر حل نماید.

اثبات: فرض کنید که مهاجم $\mathcal{D}$ یک مهاجم علیه حریم خصوصی شناسه امضاءکننده است. این مهاجم در زمان چندجمله‌ای t اجرا می‌شود و می‌تواند حداکثر تعداد q_{Sign} پرسش از سروش $\mathcal{O}_{Sign}$، حداکثر تعداد q_{Sim} پرسش از سروش $\mathcal{O}_{Sim}$ و حداکثر تعداد q_{Ver} پرسش از سروش $\mathcal{O}_{Ver}$ انجام دهد و مزیت ε را دارد. همچنین، یک شبیه‌ساز همانند $\mathcal{C}$ برای حل مسأله دیفی- هلمن گَپ (GDH) می‌سازیم. یک نمونه مسأله GDH معلوم است یعنی مقادیر (P, aP, bP, cP) و یک سروش DDH مشخص هستند؛ شبیه‌ساز $\mathcal{C}$ یک عدد تصادفی $a' \in \mathbb{Z}_q$ را انتخاب نموده و $Q_{ID_{S_0}} = aP$، $Q_{ID_{S_1}} = (aa')P$، $Q_{ID_V} = cP$ و $P_{pub} = bP$ را قرار می‌دهد. شبیه‌ساز $\mathcal{C}$، تمایزگر $\mathcal{D}$ را بر روی ورودی

$\left(Q_{ID_{S_0}}, Q_{ID_{S_1}}, Q_{ID_V}\right)$ فراخوانی نموده و سپس پرسش‌های مربوطه را برای مهاجم شبیه‌سازی می‌کند. شبیه‌ساز $\mathcal{C}$ همچنین چهار جدول H_1، استخراج، H_2 و S را تشکیل می‌دهد؛ این جدول‌ها در ابتدا خالی هستند. به جهت سادگی، فرض می‌کنیم که $\mathcal{D}$ نمی‌تواند پرسش‌های تکراری دوباره بپرسد و همچنین پرسش‌های صورت‌گرفته از سروش O_{sim} را در روند اثبات قضیه درنظر نمی‌گیریم.

پرسش‌های چکیده‌ساز-۱: زمانی‌که مهاجم $\mathcal{D}$ از سروش H_1 پرسش می‌کند، شبیه‌ساز $\mathcal{C}$ دوتایی (ID_i, R_i) را در لیست-H_1 جستجو نموده و سپس به صورت زیر پاسخ می‌دهد:

$$R_i = H_1(ID_i) = \begin{cases} aP, & \text{if } ID_i = S_0 \\ (aa')P, & \text{if } ID_i = S_1 \\ cP, & \text{if } ID_i = V \\ r_iP, & \text{if } ID_i \neq S_0, S_1, V, r_i \in_R \mathbb{Z}_q \end{cases} \quad (۲۲)$$

پرسش‌های استخراج: زمانی‌که مهاجم $\mathcal{D}$ بر روی ID_i پرسش می‌کند، شبیه‌ساز $\mathcal{C}$ پارامتر ID_i را در لیست-استخراج جستجو می‌کند. اگر $i = V$ یا $i \neq S_0, S_1$ باشد، $\mathcal{C}$ مقدار $S_i = r_i(bP)$ را به عنوان کلید عمومی متناظر ID_i خروجی می‌دهد. در غیر این صورت، شبیه‌ساز $\mathcal{C}$ بازی را متوقف کرده و گزارش شکست می‌دهد.

پرسش‌های چکیده‌ساز-۲: زمانی‌که یک پرسش (m_i, t_i) توسط مهاجم $\mathcal{D}$ از سروش H_2 صورت می‌پذیرد، شبیه‌ساز $\mathcal{C}$ در لیست-H_2 جستجو نموده و درصورتی‌که یک چنین مقداری از پیش تعریف شده باشد، خروجی c_i می‌دهد.

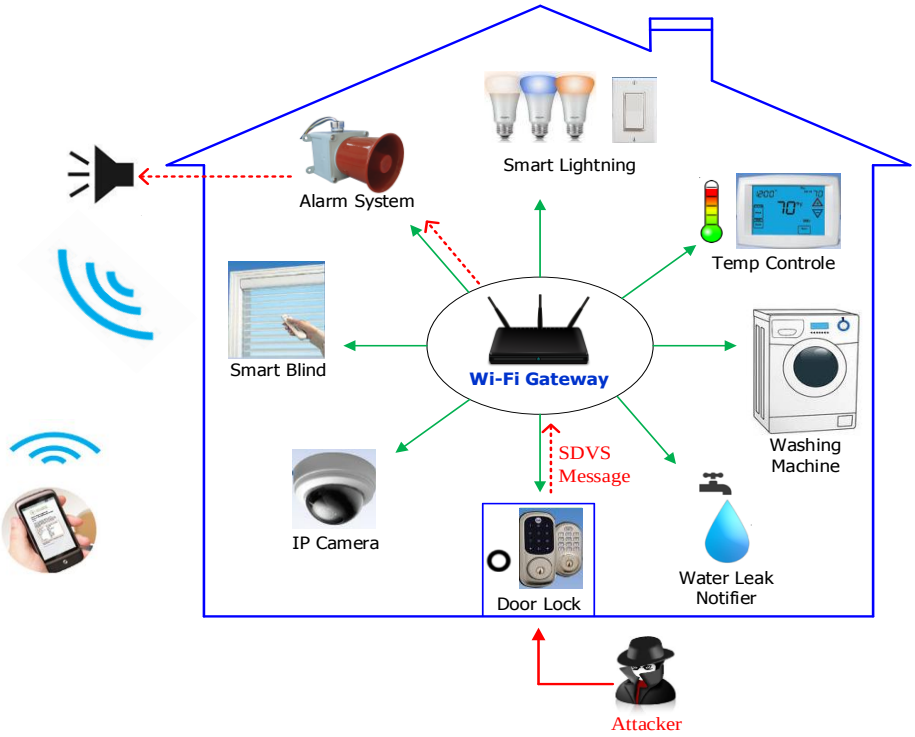
در غیر این‌صورت، $\mathcal{C}$ عدد تصادفی $c_i \in \mathbb{Z}_q$ را انتخاب نموده و $\left(m_i, pk_{S_0}, t_{i_0} \leftarrow U_{S_0} = rQ_{S_0}, c_i\right)$ و $\left(m_i, pk_{S_1}, t_{i_1} \leftarrow U_{S_1} = rQ_{S_1}, c_i\right)$ را به سروش DDH داده و دو بیت b_0 و b_1 به دست می‌آورد. اگر $b_0 = 1$ بود، $\mathcal{C}$ خروجی t_{i_0} را داده و متوقف می‌شود؛ اگر $b_1 = 1$ بود، $\mathcal{C}$ خروجی t_{i_0}/ra' را داده و متوقف می‌شود. در غیر این صورت، $b_0 = b_1 = 0$، $\mathcal{C}$ عدد تصادفی تازه $c_i \in \mathbb{Z}_q$ را انتخاب نموده و در لیست-H_2 ذخیره نموده و c_i را به $\mathcal{D}$ می‌دهد.

پرسش‌های امضاء: به ازای یک پیام معلوم M و یک بیت d، $\mathcal{C}$ عدد تصادفی $r \in \mathbb{Z}_q$ را انتخاب نموده و $U = rQ_{S_d}$ و همچنین $h = H_2(M, U)$ را محاسبه می‌کند. اگر h قبلاً استفاده شده بود، $\mathcal{C}$ فرآیند فوق را مجدداً تکرار می‌کند. $\mathcal{C}$ مقادیر $(M, \perp, t_i, c_i)$ را در لیست-H_2 ذخیره نموده و امضای $\sigma = (T, Q_{ID_V})$ را برمی‌گرداند. علاوه بر آن، $\mathcal{C}$ مقادیر (d, M, σ) را در لیست-S ذخیره می‌کند.

پرسش‌های تأیید: به ازای یک پیام معلوم M، یک بیت d و یک امضای $\sigma = (T, Q_{ID_V})$، اگر چندتایی (d, M, σ) در لیست-S وجود داشت، $\mathcal{C}$ مقدار 1 را باز می‌گرداند و در غیر این‌صورت، $\mathcal{C}$ مقدار 0 را بازمی‌گرداند.

شهود پشت شبیه‌سازی سروش تصادفی تأیید این است که اگر یک امضاء قبلاً توسط شبیه‌ساز $\mathcal{C}$ خروجی شده باشد، پس باید معتبر هم بوده باشد. در غیر این‌صورت یعنی اگر امضا، نامعتبر باشد آن‌گاه سروش تصادفی قطعاً باید مقدار 0 را بدهد. اگر امضاء معتبر باشد، مهاجم $\mathcal{D}$ با موفقیت توانسته تا یک امضای معتبر را جعل کند. این بدان معنی است که مهاجم $\mathcal{D}$ یک پرسش (M, U, t_i, c_i) پرسیده که در آن، $U = rQ_{S_d}$ می‌باشد. متعاقباً، مهاجم می‌تواند امضای $\sigma = (T, Q_{ID_V})$ و در حقیقت $e(P,P)^{abc}$ را محاسبه نموده و متوقف گردد. در این حالت، تمایزگر $\mathcal{D}$ یک پیام چالشی جدید M^* را ثبت می‌کند. سپس، شبیه‌ساز $\mathcal{C}$ یک سکه b را پرتاب می‌کند، عدد تصادفی $r^* \in \mathbb{Z}_q$ را انتخاب نموده و $U^* = r^*Q_{ID_{S_b}}$ را محاسبه می‌نماید. همچنین مقدار $h^* = H_2(M^*, U^*)$ را محاسبه می‌کند. مجدداً، اگر h^* قبلاً استفاده شده باشد، $\mathcal{C}$ فرآیند فوق را دوباره تکرار می‌کند. سپس، $\mathcal{C}$ مقدار $\sigma^* = (T^*, Q_{ID_V})$ را باز گردانده و (M^*, U^*, t_i^*, c_i^*) در لیست-H_2 ذخیره کرده و (b, M^*, σ^*) را در لیست-S ذخیره می‌نماید. $\mathcal{C}$ به شبیه‌سازی سروش‌ها برای تمایزگر $\mathcal{D}$ و مانیتورکردن پرسش‌های چکیده‌سازی ادامه می‌دهد. در نهایت، $\mathcal{D}$ یک بیت b' را خروجی می‌دهد. اگر $\mathcal{C}$ تا بدین جا متوقف نشده باشد، از بازی انصراف داده و گزارش شکست خود در یافتن $e(P,P)^{abc}$ را اعلام می‌کند.

$\mathcal{C}$ زوج‌کلید امضاءکنندگان S_0 و S_1 و تأییدکننده مشخص DV یعنی (Q_{S_0}, S_{S_0})، (Q_{S_1}, S_{S_1}) و (Q_{DV}, S_{DV}) را تولید نموده و تمایزگر $\mathcal{D}$ را بر روی ورودی $(Q_{S_0}, Q_{S_1}, Q_{DV})$ فراخوانی می‌کند.



شکل (۲): کاربرد SDVSS در خانه هوشمند.

۴-۴- ارزیابی کارآیی

در این زیربخش، مقایسه‌ای از کارآیی طرح ارائه‌شده را برحسب پارامترهایی چون طول امضاء و هزینه‌های محاسباتی مورد نیاز ارائه می‌نماییم. این مقایسه میان طرح ارائه‌شده و طرح‌های مطرح موجود دیگر انجام گفته است.

در نظر بگیرید که C_p نشان‌دهنده زمان موردنیاز برای عملیات زوج‌سازی، C_* نمایانگر ضرب در گروه G_1 و C_e نمایانگر عملیات نمایی در گروه G_2 باشد. C_h نمایانگر عملیات چکیده‌سازی و C_i نمایانگر عملیات یافتن معکوس باشد. عملیات جمعی در گروه G_1 صرف‌نظر می‌شوند. توجه شود که عملیات محاسبه C_e (توان‌رسانی) و C_p (زوج‌سازی) جزو محسبات هزینه‌بر محسوب می‌شوند؛ و پس از آن‌ها عملیات محاسبه ضرب یعنی C_* قرار دارد. محاسبه یافتن معکوس C_i و محاسبه یافتن مقدار چکیده‌سازی C_h به لحاظ پیچیدگی به ترتیب بعد از توان‌رسانی، زوج‌سازی و ضرب قرار می‌گیرند.

فرض می‌کنیم که طول بیتی یک عضو موجود در G_1 به صورت $|G_1|$ باشد (فرض می‌شود که $|G_1| = |G_2|$ است)؛ همچنین، فرض می‌کنیم که $|\mathbb{Z}_q| = 160$ بیتی باشد. جدول (۱) نشان می‌دهد که در کل طرح ارائه‌شده نسبت طرح‌های دیگر از حیث کارآمدی در هر سه قسمت: اندازه امضای خروجی، محاسبات لازم در فاز صادرشدن امضاء و محاسبات موردنیاز در فاز تأیید امضاء در وضعیتی مناسب و قابل رقابت با طرح‌های دیگر قرار دارد و این درحالی است که طرح ارائه‌شده دارای امنیت لازم بوده و ویژگی حریم‌خصوصی کاربر را نیز برآورده می‌سازد.

۵- کاربرد طرح ارائه‌شده در مفهوم اینترنت اشیاء

در این بخش کاربردهایی از طرح ارائه‌شده در فضای اینترنت اشیاء ارائه می‌نماییم.

۵-۱- کاربرد DVSS در خانه هوشمند

همان‌طورکه قبلاً توضیح داده شد، در یک خانه هوشمند صاحب منزل این امکان را دارد که برای مثال از تلفن هوشمند خود به عنوان یک کنترل از راه دور یکپارچه استفاده نموده و تمامی دستگاه‌ها و لوازم خانگی IoT را مدیریت نماید (شکل ۲).

برای نمونه، صاحب منزل می‌خواهد برخی از تجهیزات منزلش را بلافاصله بعد از اتمام کار خاموش نماید. کاربر می‌تواند پیغام خود را از طریق یک امضای با تأییدکننده مشخص ارسال نماید. مزیت این کار در این است که اگر صاحب منزل برای مثال پیغامی را که اختصاص به عملکرد ماشین لباس‌شویی دارد، اشتباهاً به دستگاه دیگری با عملکرد مشابه (برای نمونه ماشین ظرف‌شویی) ارسال نماید، در عمل مشکلی ایجاد نخواهد شد؛ چرا

که در طرح ID-DVSS ارائه‌شده گیرنده از قبل مشخص است و گیرنده دیگری نمی‌تواند از پیغامی که برای یک گیرنده خاص صادر شده است، استفاده نماید.

همچنین، فرض کنید در فصل سرما، سیستم گرمایشی منزل در زمان مشخصی از روز روشن می‌گردد تا خانه در زمان رسیدن افراد، گرم شده باشد و به این طریق در مصرف انرژی نیز صرفه‌جویی گردد. حال فرض کنید که صاحب منزل در یک روز خاص متوجه می‌شود که مهمانی ویژه دارد و نیاز است تا سیستم گرمایشی منزل در ساعتی غیر از روال برنامه روزانه‌اش روشن گردد. در این حالت، صاحب منزل می‌تواند با استفاده از DVSS ارائه‌شده این پیام را به سیستم گرمایشی منزل ارائه نموده و آن را فعال نماید.

جدول (۱): مقایسه میان طرح ارائه‌شده با طرح‌های معروف دیگر.

طرح	طول امضاء	محاسبات فاز صدور امضاء	محاسبات فاز تأیید امضاء
طرح Susilo [۱۰]	$2\|G_1\|+\|H\|$	$1C_p+2C_*+1C_e+1C_h+1C_i$	$2C_p+1C_*+2C_e+1C_h$
طرح Kumar [۹]	$4\|G_1\|$	$1C_p+5C_*+1C_h+1C_i$	$4C_p+1C_h$
طرح Zhang [۱۵]	$3\|G_1\|$	$4C_*+1C_h+1C_i$	$3C_p+1C_h$
طرح Kang [۱۶]	$2\|G_1\|$	$1C_p+1C_*+1C_h$	$1C_p+1C_h$
طرح Kang [۱۷]	$2\|G_1\|$	$2C_p+2C_*+1C_e+1C_h$	$1C_p+1C_*+1C_e+1C_h$
طرح Duan [۲۱]	$2\|G_1\|+\|\mathbb{Z}_q\|$	$1C_p+1C_*+1C_h$	$2C_p+1C_*+1C_h$
طرح Chen [۱۲]	$\|G_1\|+\|\mathbb{Z}_q\|$	$1C_p+1C_*+2C_h$	$2C_p+1C_*+2C_h$
طرح Wang [۲۲]	$2\|\mathbb{Z}_q\|$	$1C_p+1C_e+1C_*+1C_i+1C_h$	$1C_p+3C_e+1C_*+1C_h$
طرح Huang [۲۰]	$4\|G_1\|+3\|\mathbb{Z}_q\|$	$4C_p+3C_e+2C_*+5C_h$	$1C_p+1C_i+2C_h$
طرح Islam [۲۳]	$2\|G_1\|+\|\mathbb{Z}_q\|$	$2C_p+1C_e+1C_*+2C_h$	$2C_p+2C_h$
طرح Hu [۲۴]	$5\|\mathbb{Z}_q\|$	$3C_e+2C_*+2C_h$	$4C_e+2C_*+2C_h$
طرح Hu [۲۵]	$\|G_1\|+4\|\mathbb{Z}_q\|$	$6C_e+2C_*+3C_h$	$9C_e+3C_*+1C_h$
طرح ارائه‌شده	$2\|G_1\|$	$1C_p+2C_*+1C_h$	$1C_p+1C_*+1C_h$

جدول (۲): مقایسه امنیتی میان طرح ارایه‌شده و برخی از طرح‌های مطرح دیگر.

طرح	حمله اعمال شده	اثبات امنیتی
طرح Susilo [۱۰]	-	دارد
طرح Kumar [۹]	-	دارد
طرح Zhang [۱۵]	حمله به ویژگی تأییدکننده مشخص	دارد
طرح Kang [۱۶]	حمله جعل فراگیر	ندارد
طرح Kang [۱۷]	حمله به ویژگی تأییدکننده مشخص	دارد
طرح Duan [۲۱]	حمله به ویژگی اعطاپذیربودن	دارد
طرح Chen [۱۲]	-	ندارد
طرح Wang [۲۲]	-	دارد
طرح Huang [۲۰]	-	دارد
طرح Islam [۲۳]	حمله به ویژگی اعطاپذیربودن	دارد
طرح Hu [۲۴]	-	دارد
طرح Hu [۲۵]	-	دارد
طرح ارائه‌شده	-	دارد

۵-۲- کاربرد در کنترل داده ابری

همان‌طورکه در [۲۷] اشاره شده، طرح DVS ارایه‌شده می‌تواند به منظور کنترل و بررسی داده ابری مورد استفاده قرار گیرد. همانند طرح Worku و همکارانش، طرح ارایه‌شده نیز برای استفاده در سامانه کنترل داده ابری سه موجودیت دارد: کاربر (U) کـه حـجـم زیادی داده برای ذخـیره‌سـازی دارد؛ ارایه‌دهنده سرویس ابری[1] (CSP) که سرویس‌های مربوطه را به کاربرانش ارایه می‌دهد؛ و تأییدکننده مشخص (DV) که اعتبار و صحت داده کاربر را بررسی و کنترل می‌کند. کاربر و ارایه‌دهنده سرویس ابری به صورت بالقوه جزء امضاکنندگان می‌باشند و قادرند تا یک تأییدکننده مشخص را برای سامانه کنترل اختصاص دهند.

علاوه‌بر آن، هر زمان که کاربر یا ارایه‌دهنده سرویس ابری

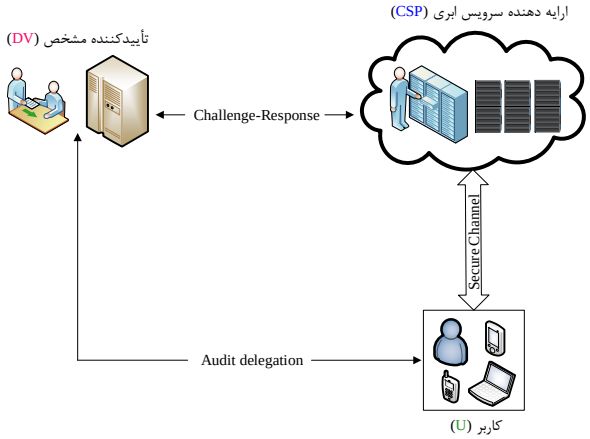
1- Cloud Service Provider

بخواهند تا تأییدکننده دیگری را جایگزین نمایند، فرض می‌کنیم که بر روی تأییدکننده دیگری مذاکره نموده و نوافق نمایند. همان‌گونه که در شکل (۳) نشان داده شده، سناریوی ارایه‌شده دارای سه فاز می‌باشد: شروع (برپایی) سامانه[1]، اعطای (قابلیت) بررسی و کنترل[2] و پروتکل چالش- پاسخ برای بررسی و کنترل[3].

فاز شروع سامانه: این فاز شامل سه الگوریتم برپایی، تولید کلید و تولید امضا می‌شود. برپایی یک الگوریتم زمان چندجمله‌ای احتمالاتی است که پارامتر امنیتی مربوطه را به صورت ورودی گرفته و پارامترهای عمومی سامانه را خروجی می‌دهد. تولید کلید پارامترهای عمومی را به عنوان ورودی گرفته و کلیدهای عمومی/ خصوصی (sk_S, pk_S) و (sk_V, pk_V) را برای کاربر و تأییدکننده مشخص خروجی می‌دهد. تولید امضاء نیز یک الگوریتم زمان چندجمله‌ای معین است که یک امضای σ را تولید می‌نماید.

فاز اعطای (قابلیت) کنترل: یک الگوریتم تخصیص تأییدکننده است که امضای σ و کلید عمومی تأییدکننده مشخص pk_V را به عنوان ورودی گرفته و یک امضای با تأییدکننده مشخص $\sigma\hat{o}$ را به عنوان خروجی می‌دهد.

فاز چالش-پاسخ: این فاز شامل سه الگوریتم می‌گردد: تولید چالش، تولید اثبات و اثبات تأیید. تولید چالش یک الگوریتم تصادفی است که پارامترهای عمومی را به عنوان ورودی گرفته و یک چالش مناسب را تولید می‌کند. الگوریتم تولید اثبات، پارامترهای عمومی، داده کاربر، امضای σ و چالش مربوطه را به عنوان ورودی دریافت نموده اثبات P را به منظور تأیید داده کاربر به تأییدکننده مشخص ارسال می‌کند. اثبات تأیید نیز یک الگوریتم تصادفی‌شده معین است که توسط تأییدکننده مشخص در زمان چندجمله‌ای اجرا می‌شود تا اعتبار اثبات P که توسط ارایه‌دهنده سرویس تولید شده است را بررسی و کنترل نماید. این الگوریتم کلید خصوصی تأییدکننده مشخص، چالش مربوطه و اثبات P را به عنوان ورودی گرفته و نتیجه تأیید را به صورت 0 یا 1 خروجی می‌دهد که در آن، 0 به معنی شکست بوده و 1 نیز به این معنی است که تأیید اثبات می‌شود و در نهایت فایل به صورت درست بر روی سرور ابری ذخیره می‌گردد.



شکل (۳): معماری ذخیره‌سازی و بررسی داده ابری.

۶- نتیجه‌گیری

طرح‌های امضای با تأییدکننده مشخص نوع خاصی از امضاءهای دیجیتال می‌باشند که در آن‌ها، شخص تأییدکننده مشخص بوده و تنها او می‌تواند اعتبار امضاء را بررسی کند. در این مقاله، یک طرح امضای مبتنی بر شناسه با تأییدکننده مشخص جدید را به همراه اثبات امنیتی آن ارائه نمودیم. همان‌طور که در جدول (۱) مشخص است، طرح ارائه‌شده از نقطه نظر کارآیی در هر سه بخش: اندازه امضای خروجی، محاسبات فاز صدور امضاء و محاسبات موردنیاز برای فاز تأیید امضاء مناسب بوده و با طرح‌های موجود قبلی قابل مقایسه می‌باشد. از طرفی، طرح ارائه‌شده در مدل سروش تصادفی دارای امنیت قابل اثبات بوده و همچنین تمامی ملزومات امنیتی دیگری را که یک طرح امضای مبتنی بر شناسه با تأییدکننده مشخص باید داشته باشد را برآورده می‌سازد. از طرف دیگر، طرح ارائه‌شده ویژگی محافظت از حریم خصوصی کاربر را برآورده می‌سازد که از این جهت می‌تواند بسیار مورد توجه قرار گیرد. همچنین، سناریوهای کاربردی از طرح ID-DVSS ارائه‌شده را در خانه‌های هوشمند ارائه نمودیم که از حیث کاربردهای آتی اولیه‌های رمزنگاشتی (همانند طرح‌های امضای دیجیتال) در حوزه اینترنت اشیاء حائز اهمیت است.

سپاس‌گذاری

این مقاله توسط بنیاد ملی علمی ایران (INSF) به شماره قرارداد ۹۲/۳۲۵۷۵ حمایت شده است.

1- System initialization
2- Auditing delegation
3- Challenge-Response protocol for auditing

۷- مراجع

[1] K.-T. Nguyen, M. Laurent, N. Oualha, "Survey on secure communication protocols for the Internet of Things," Ad-hoc Networks, pp.1-15, 2015.

[2] Gartner Inc., "Forecast: The Internet of Things," Worldwide, 2013.

[3] E. Borgia, "The Internet of Things vision: Key features, Applications and open issues," Computer Communications, pp. 1–31, 2014.

[4] J. Lu, T. Sookoor, V. Srinivasan, G. Gao, B. Holben, J. Stankovic, and E. Field, "Whitehouse, The smart thermostat: using occupancy sensors to save energy in homes," in: Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems (SenSys'10), pp. 211–224, 2010.

[5] C. Chen, D. J. Cook, and A. S. Crandall, "The user side of sustainability: modeling behavior and energy usage in the home," Pervas. Mob. Comput., vol. 9, no. 1, pp. 161–175, 2013.

[6] S.-K. Jakobsson and R. Impagliazzo, "Designated verifier proofs and their applications," In: Advances in Eurocrypt'96. LNCS, 1070. Springer-Verlag, pp. 143–54, 1996.

[7] S. Saeednia, S. Kramer, and O. Markovitch, "An efficient strong designated verifier signature scheme," In: ICISC 2003, Berlin: Springer-Verlag, pp. 40–54, 2003.

[8] X. Huang, W. Susilo, Y. Mu, and F. Zhang, "Short designated verifier signature scheme and its identity-based variant," International Journal of Network Security, vol. 6, no. 1, pp. 82–93, 2008.

[9] K. Kumar, G. Shailaja, and A. Saxena, "Identity based strong designated verifier signature scheme," Cryptography eprint Archive Report 2006/134. Available at http://eprint.iacr.org/complete/2006/134.pdf.

[10] W. Susilo, F. Zhang, and Y. Mu, "Identity-based strong designated verifier signature schemes," In: ACISP 2004, LNCS 3108, pp. 313–324, 2004.

[11] S. Lal and V. Verma, "Identity base strong designated verifier proxy signature schemes," Cryptography eprint Archive Report, 2006. Available at: http://eprint.iacr.org/complete/2006/394.pdf.

[12] G. Chen and Sh. Wan, "Analysis and improvement of identity-based designated verifier signature scheme," 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet), pp. 2388–2391, 2012.

[13] J.-S. Lee, J.-H. Chang, and D.-H. Lee, "Forgery attacks on Kang et al.'s identity base strong designated verifier signature scheme and its improvement with security proof," Computer & Electrical Engineering, 36 (2010), pp. 948-954, 2010.

[14] H. Lipma, G. Wang, and F. Bao, "Designated verifier signature schemes: attacks, new security notions and new construction," In: ICALP 2005, LNCS 3580, Springer-Verlag, pp. 459-471, 2005.

[15] J. Zhang and J. Mao, "A novel ID-based designated verifier signature scheme," Information Sciences, vol. 178, pp. 733–66, 2008.

[16] B. Kang, C. Boyd, and E. Dawson, "Identity-based strong designated verifier signature schemes: attacks and new construction," Computer & Electrical Engineering, vol. 35, pp. 49-53, 2009.

[17] B. Kang, C. Boyd, and E. Dawson, "A novel identity-based strong designated verifier signature scheme," The Journal of Systems & Software, vol. 82, pp. 270-273, 2009.

[18] J. Lee, J. Chang, and D. Lee, "Forgery atacks on Kang et al.s identity-based strong designated verifier signature scheme and its improvement with security proofs," Computers and Electrical Engineering, vol. 36, pp. 948-954, 2010.

[19] H. Du and Q. Wen, "Attack on Kang et al.s Identity-Based Strong Designated Verifier Signature Scheme," Cryptography eprint report 2006/134. International Association for CryptologicResearch, http://eprint.iacr.org/complete/2006/134.

[20] Q. Huang, G. Yang, D.-S.Wang, and W. Susilo, "Identity based strong designated verifier signature revisited," The Journal of Systems and Software, vol. 84, pp. 120-129, 2011.

[21] M. Duan, J. Xu, and D. Feng, "Efficient identity-based strong designated verifier signature schemes," Security and Communication Networks (6), Wiley, pp. 902-911, 2013.

[22] H. Wang, "Signer-admissible strong designated verifier signature from bilinear pairings," Security and Communication Networks (7), Wiley, pp. 422-428, 2014.

[23] S.-H. Islam and G.-P. Biswas, "Provably secure and pairing-based strong designated verifier signature scheme with message recovery," Arab Journal of Science Engineering Springer, vol. 40, pp. 1069-1080, 2015.

[24] X. Hu, H. Xu, Y. Liu, J. Wang, W. Tan, and X. Zhang, "An Efficient Designated Verifier Signature Scheme with Pairing-Free and Low Cost," Security and Communication Networks, vol. 9, no. 18, pp. 5724-5732, 2017.

[25] X. Hu, W. Tan, H. Xu, J. Wang, and Ch. Ma, "Strong Designated Verifier Signature Scheme with Undeniable Property and Their Application," Security and Communication Networks, pp. 1-9, 2017.

[26] Q. Huang, G. Yang, D. S. Wong, and W. Susilo, "Efficient Strong Designated Verifier Signature Scheme without Random Oracles or Delegatability," Cryptography eprint Archive Report 2009/518, http://eprint.iacr.org/2009/518.pdf.

[27] S.-G. Worku, Ch. Xu, and J. Zhao, "Cloud data auditing with designated verifier", Frontiers of Computer science, vol. 8, no. 3, pp. 503-512, 2014.

[28] M. Beheshti-Atashgah, M. Gardeshi, and M.-R. Aref, "A Designated Verifier Threshold Proxy Signature Scheme," Journal of Electronic and cyber defense, vol. 1, no. 5, pp. 25-36, 2014. (In Persian)

Journal of Electronical & Cyber Defence
Vol. 5, No. 4, 2017, Serial No. 20

A Novel Identity-based Strong Designated Verifier Signature Scheme with its Application in Internet of Things Era

M. Beheshti-Atashgah, M. R. Aref*, M. Barari
Malek-Ashtar University of Technology
(Received: 21/04/2017, Accepted: 23/07/2017)

ABSTRACT

In a strong designated verifier signature scheme, a signer can issue a signature for a special receiver; i.e. only the designated verifier can verify the validity of the issued signature. Of course, the signature scheme should be such that no third party will be able to validate the signature. In other words, the designated verifier cannot transfer the issued signature to a third party. In this article, we propose a new ID -based designated verifier signature scheme that has provable security in the random oracle model and BDH assumption. The proposed scheme satisfies all security requirements of an IDVSS. In addition, the proposed scheme protects from user's privacy and from the efficiency point of view, and more precisely, in terms of parameters such as the size of output signature and computations required for signing and verification phases. As a result,our proposed scheme is comparable with other existing schemes; in other words, the proposed scheme is a light-weight construction. Finally, we introduce some practical scenarios of the proposed scheme in the Internet of Things concept.

Keywords: ID-Based Signature Scheme, Designated Verifier Signature, Internet Of Things, Smart Home, Cloud, Provable Security, Bilinear Pairing.

* Corresponding Author Email: Aref@sharif.edu