ارائه یک طرح احراز اصالت سبک وزن امن در رایانش مه

آناهید محسنی کبیر [۱]، دکتر مجید بیات [۲] ، دکتر مهدی علاقه بند [۳]، فاطمه حمیدی فشکی [۴]

۱- دانشجوی کارشناسی ارشد مخابرات امن و رمزنگاری، دانشگاه آزاد اسلامی واحد علوم و تحقیقات

۲ - استادیار گروه کامپیوتر، دانشگاه شاهد

۳ - استادیار گروه مخابرات، دانشگاه آزاد اسلامی واحد یادگار امام خمینی

۴- دانشجوی کارشناسی ارشد مخابرات امن و رمزنگاری، دانشگاه آزاد اسلامی واحد علوم و تحقیقات

چکیده

با توجه به افزایش خدمات اینترنتی و حجم بالای اطلاعات تولیدشده و محدودیت‌های موجود در استفاده از پهنای باند انتقال این حجم از اطلاعات به ابر امکان‌پذیر نیست.به همین دلیل نیاز به رایانشی احساس شد تا این محدودیت‌ها را به حداقل برساند.رایانش مه[۳]را می‌توان یک نمونه توزیع‌شده در شبکه نظر گرفت که سرویس‌های محاسباتی را در لبه شبکه در اختیار کاربران(دستگاه‌ها) قرار می‌دهد، رایانش مه به‌عنوان یک مکمل برای رایانش ابری تمامی قابلیت‌های این رایانش ازجمله ذخیره‌سازی و پردازش اطلاعات را دارا می‌باشد و این خدمات را با کیفیت بهتری به کاربران ارائه می‌کند.رایانش مه در مقایسه با رایانش ابری دارای عملکرد بهتر با تاخیر زمانی کم‌تری در پردازش و ذخیره‌سازی اطلاعات است. احراز هویت، حفظ حریم خصوصی و به‌طورکلی حفظ امنیت از مهم‌ترین ویژگی‌هایی است که در رایانش مه موردبررسی قرار می‌گیرند. در این مقاله ابتدا به شرح مختصری از رایانش مه و طرح‌های موجود در رایانش مه می‌پردازیم، سپس با بررسی طرحی که اخیراً درزمینه احراز هویت و مدیریت کلید مطرح شده به نام SAKA-FC [1] ضعف‌های امنیتی موجود در طرح را بیان می‌کنیم ، در ادامه نیز یک طرح احراز اصالت سبک وزن جدید به کمک توابع چکیده ساز[۴] و جمع پیمانه‌ای[۵] در رایانش مه ارائه می‌دهیم که طرح ارائه شده تمامی ویژگی‌های امنیتی مورد نیاز در محاسبات مه را برآورده می‌کند.همچنین طرح پیشنهادی را از لحاظ امنیت،سربار مخابراتی و محاسباتی با طرح‌های مشابه مقایسه می‌کنیم.

کلمات کلیدی: رایانش مه، احراز اصالت، مدیریت کلید،گمنامی،ردیابی

[1] Email: anahid.mohsenikabir@srbiau.ac.ir
[2] *Corresponding author: Email: mbayat@shahed.ac.ir
[3] Fog Computing
[4] Hash function
[5] XOR

۱. مقدمه

با پیشرفت فناوری و استفاده هرچه بیش‌تر از اینترنت و با توجه به این‌که امروزه از اینترنت اشیا[1] به‌طور گسترده استفاده می‌شود، با در نظر گرفتن این موضوع که اینترنت اشیا در هر زمان و هر مکان در دسترس است و دستگاه‌های هوشمند برای برقراری ارتباط از طریق اینترنت اشیا به یکدیگر متصل می‌شوند و در این میان داده‌هایی با حجم بالا تولید و ارسال می‌شوند از این رو نیاز به فضایی برای ذخیره سازی و پردازش داده‌ها مطرح می‌شود که یکی از محدودیت‌های اینترنت اشیا نداشتن فضای کافی برای ذخیره سازی است. [2] به همین دلیل به خدماتی احتیاج داریم که رسیدن به این نیاز را سهولت بخشند که تاکنون رایانش ابری[2] نقش مهمی را در تحقق این موضوع ایفا کرده است، رایانش ابری درواقع شامل خدماتی است که کاربران با استفاده از آن بتوانند عملیات محاسباتی خود را بدون دسترسی به نرم‌افزارها و سخت‌افزارهای محاسباتی که عموماً دارای هزینه بالایی هستند و همچنین بدون در نظر گرفتن محدودیت‌های زیرساختی انجام دهند [3]. ازآنجایی‌که ابرها (رایانش ابری) در دسترس عموم قرار دارند، جذاب هستند اما دارای برخی مشکلات از قبیل عدم وجود قابلیت اطمینان[3]، تاخیر زیاد، عدم پشتیبانی از تحرک و عدم آگاهی از مکان قرارگیری[4] هستند. [4] به همین دلیل برای رسیدن به پردازش سریع‌تر نمونه‌ی دیگری از رایانش توسط شرکت سیسکو[5] در سال ۲۰۱۲ مطرح شد که به آن رایانش مه گفته می‌شود [5]، رایانش مه محاسبات و پردازش‌ها را در نزدیکی و لبه شبکه انجام می‌دهد که باعث افزایش کارایی و بهبود عملکرد سیستم می‌شود.

رایانش مه برای پردازش داده‌های حجیم و سرویس‌هایی که نیاز به سرعت در پاسخ‌گویی و تاخیر کم دارند در محیط مبتنی بر ابر به بهترین نحو ممکن عمل می‌کند که این شامل شهرهای هوشمند، نظارت بر ساختمان‌های هوشمند، نیازهای اورژانسی که به سلامت بیماران مربوط است و یا خدماتی که نیاز به پاسخ گویی سریع دارند، می‌شود. به دلیل این که رایانش مه از رایانش ابری دارای تاخیر کم‌تری است، در شرایطی که نیاز به ارسال و یا پردازش در کوتاه‌ترین زمان را داریم در کنار استفاده از رایانش ابری از رایانش مه نیز کمک می‌گیریم. [6] به طور کلی به دلیل این که پردازش در مه در جایی نزدیک به کاربران و در لبه شبکه اطلاعات را پردازش می‌کند و این موضوع که نیاز نیست داده ها برای پردازش حتما به ابر ارسال شوند زمان لازم برای انجام محاسبات و ارسال داده کاهش می‌یابد. از این رو رایانش مه به پردازش سریع‌تر اطلاعات کمک می‌کند و بار مخابراتی و محاسباتی را نیزکاهش می‌دهد و به کاربران این امکان را می‌دهد تا بتوانند در فضایی مناسب‌تر و امن‌تر، اطلاعات خود را ذخیره و پردازش کنند.

احراز هویت و حریم خصوصی از مهم‌ترین ویژگی هایی هستند که قصد داریم در این مقاله به آن‌ها اشاره کنیم، به همین دلیل ابتدا در بخش دوم به بررسی کارهای پیشین در زمینه احراز هویت، یکپارچگی[6] و حملاتی مانند حمله تکرار[7]،حمله ردیابی[8] و چند حمله دیگر می‌پردازیم، در بخش سه به جزییات طرح SAKA-FC [1] اشاره می‌کنیم و سپس در بخش چهارم طرح بهبود یافته که نقص‌های امنیتی در آن برطرف شده است را ارائه و ایرادات مطرح شده در طرح را نیز بررسی می‌کنیم. در بخش پنجم به تحلیل امنیتی طرح اشاره و بررسی می‌کنیم که طرح احراز اصالت سبک وزن در برابر چه حملاتی مقاوم است،در بخش ششم نیز یک نتیجه‌گیری ارائه می‌شود.

---

[1] Internet of Things(IoT)
[2] Cloud Computing
[3] Reliability
[4] Location Awareness
[5] Cisco
[6] Integrity
[7] Reply attack
[8] Traceability

۲. کارهای پیشین

از آنجایی که یکی از ویژگی های مهم در رایانش مه احراز هویت و امنیت است و با توجه به این موضوع که رایانش مه اساسا در ارتباط با رایانش ابری معنا پیدا می‌کند و این امکان را برای کاربران و دستگاه‌ها فراهم می‌آورد تا از خدمات و ویژگی‌هایی نظیر ذخیره سازی، پردازش و توزیع اطلاعات در لبه شبکه بهره مند شوند. ابتدا به ویژگی‌های امنیتی اشاره شده در رایانش ابری می‌پردازیم و سپس به بررسی مقالات مرتبط با رایانش مه اشاره می کنیم.

با توجه به آن‌چه که در [7] آمده است می‌توان تهدیدات امنیتی مربوط به رایانش ابری را به پنج دسته اصلی تقسیم‌بندی کرد: دسته اول حملات فیزیکی که گم شدن و یا دزیده شدن دستگاه موبایل از جمله این حملات می‌باشد، دسته دوم حملات مبتنی بر برنامه‌ها است مشابه نرم‌افزارهای مخرب،بدافزارها و حریم خصوصی، در دسته سوم به حملات شبکه که حملات جعل[1]،حمله داخلی و انکار سرویس[2] در این دسته قرار می‌گیرند، اشاره شده است.دسته چهارم تهدیدات مبنی بر شبکه نظیر حمله فیشینگ و بازیابی اطلاعات مرورگرها و در دسته پنجم به دیگر حملات فعال نظیر آسیب‌پذیری پروتکل-ها،دسترسی غیرمجاز و دسترسی به اطلاعات مرکز داده اشاره کرده است. برای رایانش ابری راه‌حل‌های امنیتی بسیاری وجود دارد مانند طرح‌های مبتنی بر کلمه عبور [8] که با توجه به برخی ویژگی‌ها برای استفاده در رایانش مه مناسب نیستند. به‌طورکلی با توجه به بررسی‌های انجام شده در [9] محاسبات مه ویژگی‌هایی نظیر انجام محاسبات در مقیاس بالا و به صورت محلی را دارند، همچنین دارای قابلیت انتقال و جابه‌جایی از یک مکان به مکان دیگر نیز هستند و می‌توان در سیستم‌های حساس به زمان و مکان نیز از آن‌ها استفاده کرد.

مقاله [10] نشان داد که با ارائه رایانش مه مقدار قابل توجهی از ذخیره سازی داده‌ها، محاسبات و ارتباطات در رایانش ابری افزایش یافته است که این رایانش باعث افزایش کارایی و بهبود عملکرد سیستم می‌شود، یک تفاوت مهم و اساسی این است که رایانش ابری تلاش می‌کند تا پردازش و محاسبات را به صورت جهانی و گسترده بهینه سازی کند، در حالی که رایانش در مه سازماندهی و مدیریت را به صورت محلی بهینه سازی می‌کند. محاسبات در مه به دلیل نزدیکی به اشیا و دستگاه‌های هوشمند به صورت متمرکز انجام می‌شوند به همین خاطر برای استفاده در شهرهای هوشمند،خانه‌های هوشمند و به طور کلی زندگی هوشمند مناسب هستند، طرح ESHOPE [11] به تمامی ویژگی‌ها و دلایل استفاده از رایانش مه در زندگی هوشمند اشاره کرده است اما در این طرح به ویژگی های امنیتی اشاره‌ای نشده است. در سال ۲۰۱۹ طرحی توسط Alshahrani و Traore [12] ارائه شد که دارای یک معماری جدید مبتنی بر محاسبات مه است، در این مقاله با استفاده از زنجیره توابع چکیده‌ساز یک طرح تبادل کلید سبک وزن با احراز هویت متقابل معرفی شده است که در آن شناسه‌های اصلی کاربران و دستگاه‌ها فقط برای مراجع معتبر قابل تشخیص است و به جای استفاده از شناسه اصلی از شناسه موقت استفاده می‌شود، علاوه بر شناسه‌های موقت از کلیدهای نشست مخفی نیز کمک گرفته می‌شود که این کار امنیت طرح را بهبود می‌بخشد و طرح را در برابر حملاتی مانند حمله داخلی، حمله تکرار، حمله جعل و بسیاری از حملات دیگر مقاوم می‌سازد.

در واقع رایانش مه یک معماری سلسله مراتبی را ایجاد می‌کند که محاسبات محلی در لبه شبکه و محاسبات جهانی در ابر صورت می‌گیرند،این مدل از رایانش شامل سه لایه کاربر،مه و ابر است که استفاده از این سیستم به کمک مدیریت شناسه-های مستعار در رایانش در مه یک ارتباط امن و حریم خصوصی را فراهم می‌کند. [13]

در طرح ارائه شده در [14] برای افزایش اطمینان و گمنامی و محرمانگی کاربران و داده‌ها از شناسه‌های مستعار به کمک رمزنگاری هم ریخت[3] استفاده شده است، یکی از مهم‌ترین مزیت‌های این طرح استفاده از رمزنگاری هوموروفیک است که

[1] Impersonation
[2] Deniable of Service(DoS)
[3] Homomorphic encryption

می تواند برای افزایش حفظ حریم خصوصی با کمک رایانش مه و بدون نیاز به رمزگشایی پردازش های لازم را انجام دهد که این کار سرعت پردازش را تا میزان بسیار زیادی افزایش می‌دهد.

با این که طرح های بسیاری در زمینه توافق کلید مطرح شده است طرح [15] توانسته در یک پروتکل سه طرفه و با کمک تزویج دو خطی[1] تنها در یک مرحله به کلید نشست مشترک دست پیدا کند اما طرح فوق دارای ضعف هایی نظیر نداشتن امنیت پیشرو[2] و مقاوم نبودن در برابر برخی حملات فعال بود. J. Xiaoying و همکارانش [16] توانستند با ارائه یک طرح توافق کلید مبتنی بر رایانش مه و با استفاده از خم بیضوی[3] و تزویج دو خطی به احراز هویت دو طرفه برسند و طرحی را معرفی کردند که در برابر حملاتی چون مردی در میانه و تکرار مقاوم است و همچنین دارای امنیت پیشرو نیز هست.

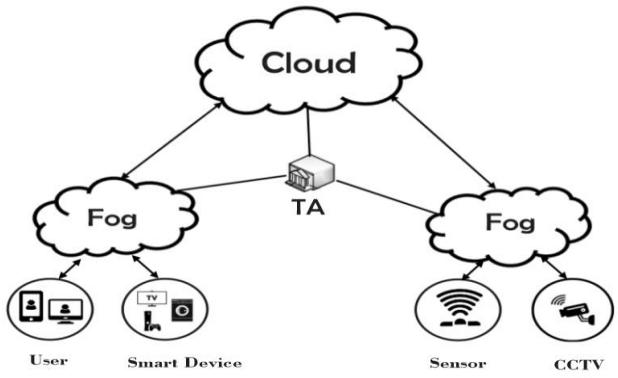
در طرح احراز هویت متقابل [17] و ازآن جایی که احراز هویت متقابل یکی از مهم ترین ویژگی‌ها در رایانش مه است هر کاربر مه و هر گره مه می‌توانند تنها با ذخیره کردن یک کلید مخفی طولانی مدت و بدون نیاز به یک زیر ساخت کلید عمومی[4] (PKI) بتوانند یکدیگر را احراز هویت کنند.در مقاله [18] به طور کامل به تهدیدات و چالش های امنیتی اشاره شده است که شامل هشت چالش امنیتی و راه حل‌هایی برای مقابله با این حملات است و به طور کامل رایانش در مه را در برابر حملاتی همچون حمله مردی در میانه،انکار سرویس و ... بررسی کرده است.

همان طور که Stojmenovic و همکارانش [19] ابتدا تنها به بررسی حمله مردی در میانه به عنوان یکی از ویژگی های امنیتی پرداخته بودند، اما در [20] به جنبه‌های دیگر احراز هویت، احراز اصالت و حریم خصوصی اشاره کردند. ما نیز در این مقاله با بررسی طرح SAKA [1] نقض‌های امنیتی موجود را برطرف و طرحی را ارائه می‌کنیم تا مشکلات امنیتی طرح پیشین که به تازگی ارائه شده است را نداشته باشد.

۳. بررسی طرح SAKA-FC

طرح احراز هویت و مدیریت کلید در رایانش مه شامل سرورهای ابر[5] (CS)، سرورهای مه[6] (FS)، دستگاه‌های هوشمند[7] (SD)، کاربر(دستگاه موبایل) ($U_i(MD_i)$) می‌باشد که دستگاه موبایل اطلاعاتی که برای احراز هویت لازم است را ذخیره می‌کند، در این طرح یک مرجع معتبر مورد اعتماد[8] (TA) نیز وجود دارد که تنها مسئول انجام مرحله ثبت‌نام است و در مراحل بعدی حضور ندارد. طرح در سه مرحله: ۱. ثبت نام، ۲. مدیریت کلید، ۳. احراز هویت و تولید کلید نشست انجام می‌شود. برای ارسال اطلاعات بین نهادها در مرحله ۱ و ۲ از کانال امن استفاده شده است و تنها مرحله احراز هویت در یک کانال ناامن (عمومی) انجام می‌شود، در ادامه به بررسی هریک از مراحل می‌پردازیم، برای درک بهتر ساختار کلی طرح در شکل ۱ نشان داده شده است و همچنین نمادها و توابع مورد استفاده در طرح نیز در جدول ۱ آورده شده است.

[1] Bilinear Pairing
[2] Forward Secrecy
[3] Elliptic Curve
[4] Public key infrastructure
[5] Cloud Servers
[6] Fog Servers
[7] Smart Devices
[8] Trusted Authority



شکل ۱- ساختار کلی طرح احراز هویت و مدیریت کلید امن [1]

جدول ۱- نمادها و اختصارات به کار رفته در طرح [1]

نمادها و اختصارات	
TA	مرجع معتبر مورد اعتماد
D_k,ID_k	دستگاه هوشمند، شناسه دستگاه هوشمند
FS_j,ID_j	سرور مه، شناسه سرور مه
ID_l,CS_l	سرور ابر،شناسه سرور ابر
MD_i ,U_i	کاربر،دستگاه موبایل کاربر
ID_i,PW_i,BIO_i	شناسه،کلمه عبور، بیومتریک کاربر
RID_i,RID_k,RID_l,RID_j	شناسه‌های مستعار سرور مه،سرور ابر،دستگاه هوشمند و کاربر
TID_i,TID_k,TID_l,TID_j	شناسه‌های موقت سرور مه،سرور ابر،دستگاه هوشمند و کاربر
k	کلید مخفی TA
s	عدد تصادفی مخفی کاربر
r_u,r_f,r_k	عدد تصادفی دستگاه هوشمند،سرور مه و کاربر
RTS_i,RTS_k,RTS_l,RTS_j	مهر زمانی زمان ثبت‌نام سرور مه،سرور ابر،دستگاه هوشمند و کاربر
TS_u,TS_k,TS_f	مهر زمانی فعلی سرور مه،دستگاه هوشمند و کاربر
n_u,n_f,n_k	تعداد دستگاه‌های هوشمند،سرورهای مه و کاربران
ΔT	حداکثر میزان تاخیر در ارسال پیام
σ_i	کلید خصوصی پارامتر بیومتریک کاربر
τ_i	مقدار عمومی پارامتر بیومتریک کاربر
||,⊕, h()	تابع چکیده ساز،جمع پیمانه‌ای،الحاق کردن
$SK_{ki}=SK_{ik}$	کلید نشست بین دستگاه هوشمند و کاربر
P	مولد گروه G

۳-۱. ثبت نام

TAمسئول ثبت نام تمامی اجزای شبکه است و تمامی اجزا برای این که بتوانند وارد پروتکل شوند باید توسط TA که یک مرجع قابل اعتماد است بررسی شوند، ثبت نام در چهار مرحله و به ترتیب زیر انجام می‌شود.

مرحله۱ :در ابتدا TA یک شناسه ID_k و یک شناسه موقت TID_k برای هر دستگاه هوشمند D_k انتخاب می‌کند و سپس RID_k و TC_k را تولید می‌کند. در نهایت $\{RID_k , TID_k , TC_k , F(TID_k,y)\}$در حافظه D_k ذخیره می‌شود.

مرحله ۲ :در این مرحله TA برای سرور مه FS_j یک شناسه ID_j و یک شناسه موقت TID_j در نظر گرفته و سپس RID_j و TC_j به ازای هر سرور تولید می‌کند و مقادیر $\{\{RID_i|i=1,2,\ldots,n_u\}, RID_j,TID_j ,TC_j ,F(TID_j,y)\}$ نیز در حافظه FS_j برای انجام مراحل بعدی ذخیره می‌شود.

مرحله ۳: برای هر سرور ابر CS_l یک شناسه ID_l و یک شناسه موقت TID_l توسط TA انتخاب می‌شود و RID_l وTC_l نیز به ازای هر سرور تولید می‌شود، مقادیر $\{\{(RID_l,TID_l,TC_l),\{G_{j,l}(TID_l,y)|j=1,2,\ldots,n_f\}\}$ را در حافظه CS_l و$\{G_{j,l}(Tal_ly)|j=1,2,\ldots,n_c\}$ در حافظه FS_j برای این‌که بتواند هر سرور ابر را شناسایی کند،ذخیره می‌شود.

مرحله ۴: در مرحله آخر ثبت‌نام کاربر(دستگاه موبایل) $U_i(MD_i)$ یک پارامتر بیومتریک $Gen(BIO_i)=(\sigma_i,\tau_i)$، یک عدد تصادفی مخفی s و یک شناسه ID_i را انتخاب و RID_i را برای خودش تولید می‌کند. سپس کاربر یک کلید خصوصی $d_i \in Z_p^*$ انتخاب و کلید عمومی $P_i= d_i.G$ که مرتبط با آن است را نیز تولید می‌کند و $\{RID_i , P_i\}$ را برای TA ارسال می‌کند، سپس به ازای هر کاربر TC_i را تولید و $\{TC_i,\{(TID_j,\{(TID_j,h(TC_j)|j=1,2,\ldots,n_f\}\}$ را برای U_i ارسال می‌کند.

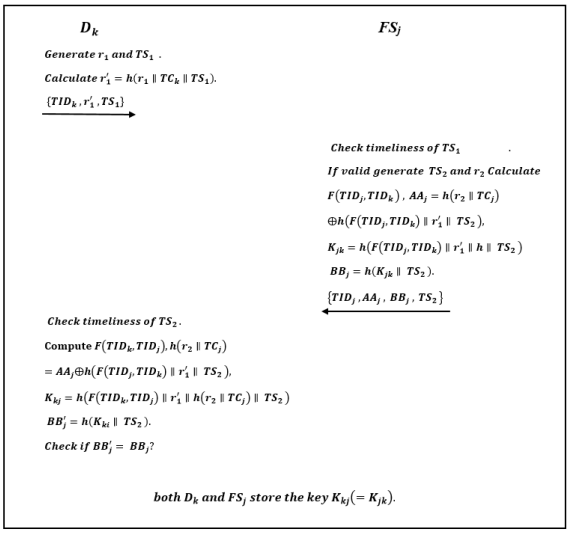
کاربر در نهایت مقادیر $\{RID_i^*,d_i^*,TC_i^*,RPB_i,\{(TID_j,\{(TID_j, TC_j^*|j=1,2,\ldots,n_f\},P_i,\tau_i,Gen(.),Rep(.),h(.),et\}$ حساب و ذخیره می‌کند و تمامی مقادیر قبلی $\{s,RID_i,d_i,TC_i,\{h(TC_j)|j=1,2,\ldots,n_f\}\}$ را از حافظه خودش پاک می‌کند.

۳-۲. مدیریت کلید

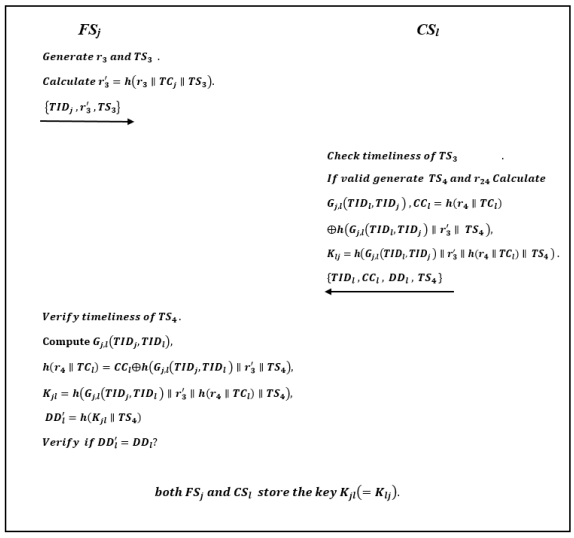
مدیریت کلید در یک کانال امن و در دو مرحله انجام می‌شود، مرحله اول بین D_kو FS_j و مرحله دوم بین FS_j و CS_l ، در هر دو مرحله نهادها بعد از تایید هویت یکدیگر یک کلید مشترک مخفی برای انجام محاسبات بعدی می‌سازند.

مرحله اول: ابتدا دستگاه هوشمند یک عدد تصادفی r_1 و یک مهر زمانی TS_1 انتخاب می‌کند و برای سرور مه $\{TS_1, r'_1, TID_k\}$را ارسال می‌کند سپس سرور مه بعد از بررسی TS_1 و تایید آن یک r_2 و یک مهر زمانی TS_2 را انتخاب سپس AAj، BB_j، $F(TID_j, TID_k)$ و کلید K_{jk} را محاسبه می‌کند و $\{TS_2, BB_j, AA_j, TID_j\}$ را برای D_k می‌فرستد.دستگاه هوشمند پس از دریافت پیام مهر زمانی TS_2 را بررسی و در صورت تایید کلید K_{kj} را تولید می‌کند، در نهایت یک کلید مشترک مخفی بین دستگاه هوشمند D_k و سرور مه FS_j ساخته می‌شود. در شکل ۲ مرحله اول مدیریت کلید نشان داده شده است.

مرحله دوم:در این مرحله سرور مه یک عدد تصادفی r_3 و یک مهر زمانی TS_3 انتخاب می‌کند و برای سرور ابر $\{TS_3, r_3, TID_j\}$ را ارسال می‌کند پس از آن سرور ابر بعد از بررسی TS_3 و تایید آن یک r_4 و یک مهر زمانی TS_4 را انتخاب می‌کند و CC_l، $G_{j,l}(TID_j, TID_l)$ و کلید K_{lj} را محاسبه می‌کند و $\{TS_4, DD_l, CC_l, TID_l\}$ را برای FS_j می‌فرستد.دستگاه هوشمند پس از دریافت پیام مهر زمانی TS_4 را بررسی و در صورت تایید کلید K_{jl} را تولید می‌کند.در این مرحله نیز یک کلید مشترک مخفی بین سرور مه FS_j و سرور ابر CS_l تشکیل می‌شود، مرحله دوم نیز در شکل ۳ به نمایش درآمده است.



شکل۲- مدیریت کلید بین D_k و FS_j [1]



شکل۳- مدیریت کلید بین FS_j و CS_l [1]

۳-۳. احراز هویت و توافق کلید

مرحله احراز هویت بین کاربر(دستگاه موبایل) $U_i(MD_i)$، سرور مه FS_j و دستگاه هوشمند D_k در یک کانال عمومی انجام می‌شود، کاربر و یا دستگاه موبایل برای شروع احراز هویت ID_i،PW_i و BIO'_i را وارد می‌کند، مقادیر $\{TC_i, d_i, RID_i, RPB_i, Б_i\}$ محاسبه می‌کند سپس TID_j ،RID_k و TC^*_j را به عنوان ورودی می‌گیرد، یک عدد تصادفی r_u و یک مهرزمانی TS_u انتخاب می‌کند و مقادیر $\{a_u, R_u, RID'_i, E_u, F_u\}$ همانند آن‌چه که در شکل۴ نشان داده شده است، محاسبه می‌کند. سپس پیام ۱ که شامل $\{TS_u, RID'_i, E_u, F_u, a_u, R_u\}$ است را برای سرور مه می‌فرستد.

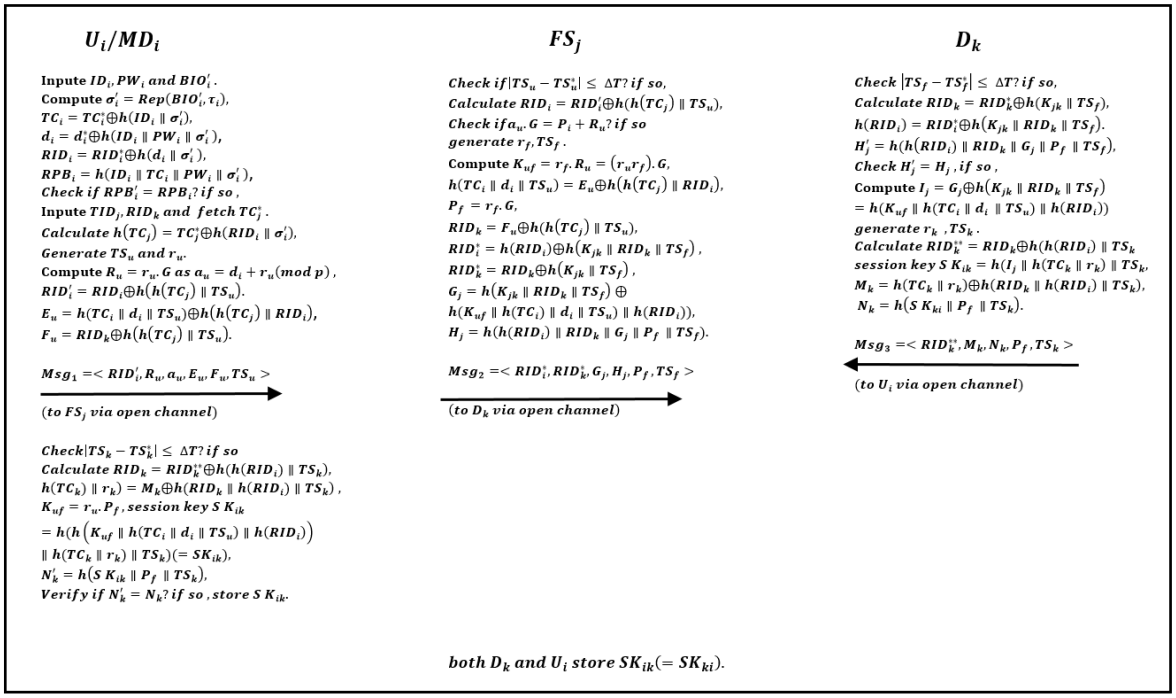
پس از دریافت پیام۱ ابتدا FS_j مهرزمانی $|TS_u - TS^*_u| \leq \Delta T$ را بررسی می‌کند، بعد از مقایسه a_u وتولید RID_i سرور مه یک عدد تصادفی r_f و یک مهرزمانی TS_f را انتخاب می‌کند و مقادیر $\{K_{uf}, RID_k, RID^*_k, RID^*_i, P_f, G_j, H_j\}$ را با توجه به روابطی که در در شکل۴ آمده، محاسبه می‌کند. سپس پیام۲ $\{TS_f, RID^*_i, RID^*_k, P_f, G_j, H_j\}$ را از طریق کانال عمومی برای دستگاه هوشمند ارسال می‌کند.

بعد از این‌که D_k پیام۲ را دریافت کرد مهرزمانی $|TS_f - TS^*_f| \leq \Delta T$ را بررسی می‌کند، سپس RID_k،$h(RID_i)$ و H'_j را محاسبه و H'_j را با H_j که دریافت کرده است مقایسه می‌کند ،در صورت تایید I_j را محاسبه و یک عدد تصادفی r_k و یک مهرزمانی TS_k را نیز انتخاب می‌کند.در نهایت نیز $\{RID^{**}_k, M_k, N_k\}$و یک کلید نشست[1] Sk_{ki} را طبق روابط موجود در شکل ۴ محاسبه می‌کند و سپس مقادیر $\{TS_k, RID^{**}_k, M_k, N_k, P_f\}$ را در قالب پیام۳ برای U_i ارسال می‌کند.

در آخرین مرحله نیز با دریافت پیام۳ ابتدا صحت مهرزمانی $|TS_k - TS^*_k| \leq \Delta T$ را بررسی می‌کند،سپس RID_k، $h(TC_k||r_k)$، k_{uf} و یک کلید نشست Ski_k را مطابق روابط ذکر شده در شکل ۴ محاسبه می‌کند و در نهایت نیز N'_k را محاسبه و N'_k را با N_k دریافت شده، مقایسه می‌کند.

پس از این که تمامی مراحل به درستی انجام شد U_i و D_k یک کلید نشست می‌سازند تا بتوانند در مراحل بعدی به جای استفاده از کلید اصلی خود از کلید نشست Sk_{ik} کمک بگیرند.

[1] Session key



شکل۴- مراحل احراز هویت و توافق کلید نشست بین D_k و U_i [1]

۳-۴. بررسی امنیتی

در طرح SAKA-FC [1] از کلید نشست استفاده شده است که سبب می‌شود طرح دارای امنیت پیشرو باشد و در صورت لو رفتن یک کلید نشست امنیت طرح همچنان حفظ شود. در هر مرحله از یک مهر زمانی جدید استفاده می‌شود و اعتبار مهر زمانی نیز قبل شروع محاسبات بررسی می‌شود که این کار سیستم را در برابر حمله تکرار و حمله انکار سرویس مقاوم می‌کند.طرح دارای گمنامی نیز هست زیرا در هیچ مرحله‌ای از شناسه کاربر و یا دیگر نهادها استفاده نشده و در پیام‌های ارسالی از شناسه مستعار و شناسه موقت استفاده شده است.

اما برخلاف آن‌چه که ادعا شده طرح در برابر حمله ردیابی مقاوم نیست زیرا برای تولید کلید نشست از مقدار di.G استفاده می‌شود که یک مقدار ثابت است و به راحتی از روی پیام۱ قابل محاسبه است. همچنین طرح در مقابل حمله دستگاه هوشمند سرقت شده نیز آسیب‌پذیر است زیرا $\{RID_k, TID_k, TC_k, F(TID_k,y), K_{jk}\}$ در داخل دستگاه ذخیره می‌شود و اگر مهاجم پیام۲ را دریافت کند به راحتی می‌تواند با اطلاعاتی که در اختیار دارد یک کلید نشست معتبر تولید کند به همین دلیل طرح در برابر حمله دستگاه هوشمند سرقت شده نیز آسیب‌پذیر است.

۴. طرح بهبود یافته

با بررسی‌های انجام شده بر روی طرح SAKA-FC [1] به این نتیجه رسیدیم که برخلاف آن چه که در آنالیزهای امنیتی به آن اشاره شده است طرح دارای ضعف های امنیتی است و در برابر ۱.حمله ردیابی[1] و ۲. حمله دستگاه هوشمند سرقت شده[2] آسیب پذیر است از این رو با اعمال تغییراتی در طرح توانستیم علاوه بر رفع این آسیب پذیری‌ها، سربار مخابراتی و

[1] Traceability Attack
[2] Stolen Smartcard Attack

محاسباتی را نیز تا حد امکان افزایش ندهیم.طرح بهبود یافته همانند طرح اصلی شامل سه مرحله است که در ادامه به بررسی اجمالی هر مرحله می‌پردازیم و سپس به طور مختصر اشاره می‌کنیم که اعمال هر تغییر چه نقشی در بهبود امنیت داشته است.

۴-۱. مرحله ثبت نام

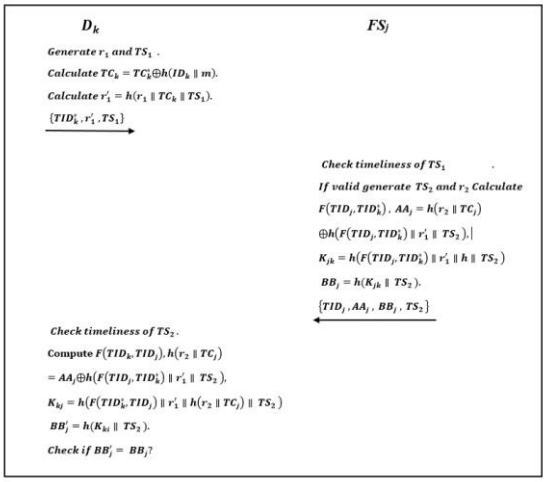
در طرح بهبود یافته ابتدا برای اطمینان از این که تمامی اجزا معتبر هستند مرحله ثبت نام انجام می‌شود و تفاوت آن این است که در مرحله ۱ دستگاه هوشمند D_k به جای ذخیره کردن مقادیر $\{RID_k , TID_k , TC_k ,F(TID_k,y)\}$ یک عدد تصادفی مخفی m به عنوان کلید خصوصی انتخاب می‌کند که در دستگاه ذخیره نمی‌شود. با کمک روابط ۲،۱و۳ که در زیر آمده است مقادیر $\{RID_k^*,TID_k^*, TC_k^*,F(TID_k,y)\}$ را جایگزین و مقادیر قبلی را از حافظه دستگاه پاک می‌کنیم.

$$TC_k^* = TC_k \oplus h(ID_k \| m) \quad (۱)$$
$$RID_k^* = RID_k \oplus h(ID_k \| m) \quad (۲)$$
$$TID_k^* = TID_k \oplus h(ID_k \| m) \quad (۳)$$

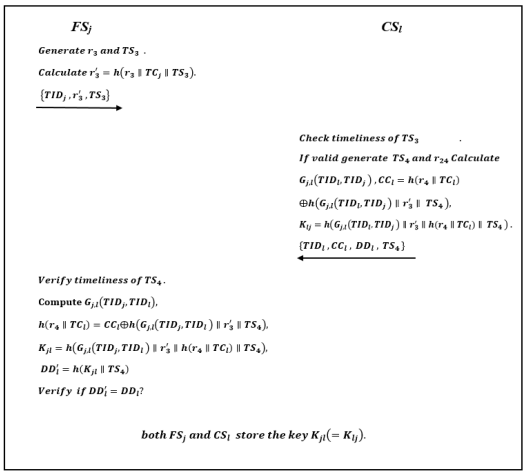
مرحله ثبت نام کاربر U_i،سرور مه FS_j و سرور ابر CS_l مانند طرح قبلی انجام می‌شود و بدون تغییر باقی می‌مانند زیرا ثبت نام توسط TA انجام می‌شود و به مقادیر ذخیره شده در دستگاه هوشمند ارتباطی ندارند.

۴-۲. مرحله مدیریت کلید

مدیریت کلید همانطور که گفته شد در یک کانال امن و در دو مرحله انجام می‌شود، مرحله اول بین D_k و FS_j ، مرحله دوم بین FS_j و CS_l انجام می‌شود. مرحله اول و دوم با اعمال تغییرات در شکل ۵ و ۶ نشان داده شده است.



شکل۵- مدیریت کلید اصلاح شده بین D_k و FS_j



شکل۶- مدیریت کلید بین FS_j و CS_l

از آن جایی که تغییرات اعمالی تنها بر روی دستگاه هوشمند انجام شده است، در شکل ۵ می‌بینیم که ابتدا به کمک رابطه ۱ مقدار TC_k را به دست می‌آوریم و به جای ارسال TID_k نیز TID_k^*در کانال ارسال می‌شود و در انتها نیز پس از این که یک کلید مشترک بین D_kو FS_j ساخته شد D_k به جای ذخیره کلید K_{kj} با کمک رابطه ۴ کلید K_{kj}^* را ذخیره می‌کند و FS_j همان کلید K_{jk} را در حافظه خود ذخیره می‌کند، اما مدیریت کلید بین FS_jو CS_l تغییری

نکرده است و مشابه طرح پیشین می‌باشد.

$$K_{kj}^{*} = K_{kj} \oplus h\,(ID_k \| m) \qquad (۴)$$

۴-۳. احراز هویت و توافق کلید

همانند آن چه که گفته شد این مرحله از طرح در یک کانال عمومی و بین کاربر(دستگاه موبایل) $U_i(MD_i)$،سرور مه FS_j و دستگاه هوشمند D_k انجام می‌شود ولی سرور مه در تولید کلید نشست نقشی ندارد و تنها مسئول برقراری ارتباط و ارسال پیام‌ها بین کاربر و دستگاه هوشمند است از این رو نیاز به محاسبات سنگینی همچون تزویج دو خطی نداریم. تفاوت ایجاد شده در این قسمت همان‌طور که در شکل ۷ مشاهده می‌شود استفاده از یک عدد تصادفی است که کاربر یک عدد تصادفی x_i انتخاب می‌کند و برای محاسبه a_u از آن استفاده می‌کند سپس در پیام ۱ علاوه بر ارسال $\{RID_i', R_u, a_u, E_u, F_u, TS_u\}$، y_i و x_i را نیز برای FS_j ارسال می‌کند تا بتواند محاسبات لازم را انجام دهد. در رابطه ۵ و ۶ نحوه انجام محاسبات آورده شده است.

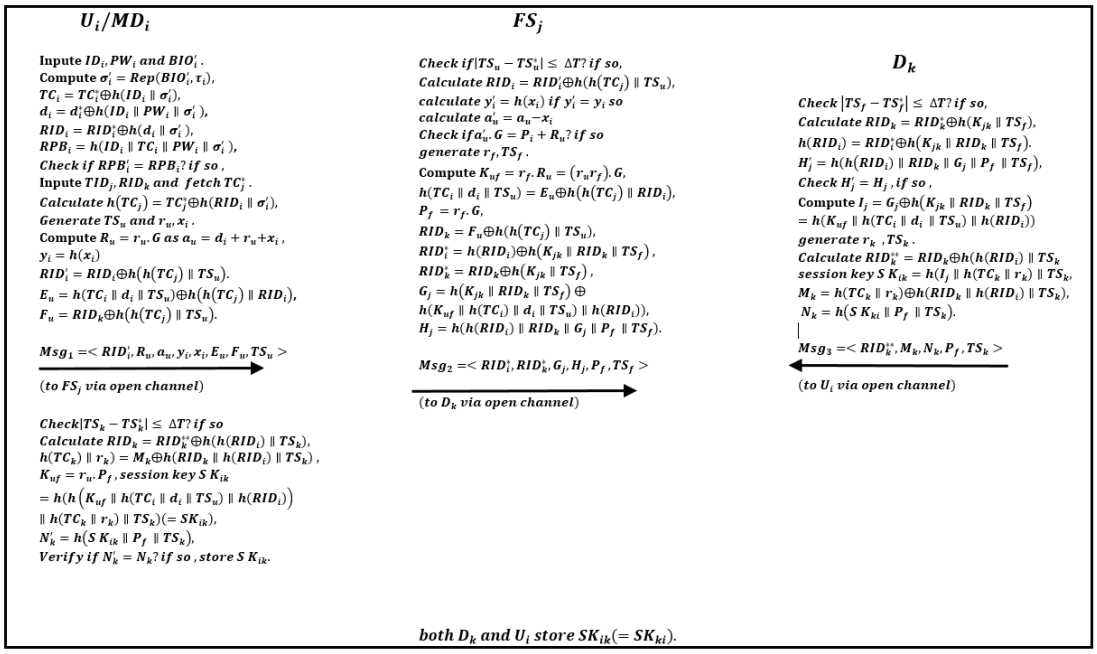
$$a_u = d_i + x_i \qquad (۵)$$

$$y_i = h\,(x_i) \qquad (۶)$$

پس از دریافت پیام ۱ ابتدا FS_j، $|TS_u - TS_u^{*}| \le \Delta T$ را بررسی می‌کند،سپس با استفاده از رابطه ۷ y_i' را محاسبه و با y_i که از U_i دریافت کرده است مقایسه می‌کند و در صورتی که مقدار یکسانی داشته باشند، مقادیر بعدی را محاسبه می‌کند.

$$y_i' = h\,(x_i) \qquad (۷)$$

بقیه مراحل تغییری نکرده و مانند آن‌چه که بررسی شد انجام می‌شود.



شکل۷- مراحل احرازهویت و توافق کلید نشست اصلاح شده بین D_k و U_i

۴-۴. بررسی حملات پیشنهادی

یکی از حملاتی که طرح در برابر آن مقاوم نبود حمله ردیابی است، به دلیل این که مقدار $d_i.G$ همواره یک مقدار ثابت است مهاجم می‌تواند به راحتی با استفاده از a_u که بر روی کانال ارسال می‌شود و G که یک پارامتر عمومی است $d_i.G$ را حساب کند و با قرار دادن آن در روابط بین تراکنش‌ها یک ارتباط ایجاد کند. از این رو برای این که طرح در برابر حمله ردیابی مقاوم شود، کاربر در هر مرحله برای تولید یک کلید نشست جدید یک عدد تصادفی انتخاب و طبق آن‌چه که در روابط ۵ و ۶ گفته شده این عدد تصادفی را با d_i جمع کرده و چکیده آن را نیز با کمک تابع چکیده‌ساز یک طرفه محاسبه می‌کند. این کار سبب می‌شود که برای هر کلید نشست جدید یک عدد تصادفی دیگر انتخاب شود و مهاجم نتواند ارتباطی بین کلیدهای نشست برقرار کند و طرح نیز در برابر حمله ردیابی آسیب‌پذیر نباشد.

پس از بررسی حمله ردیابی، حمله دستگاه هوشمند سرقت شده را بررسی می‌کنیم.

عامل اصلی در حمله سرقت D_k این موضوع بود که تمامی مقادیری که بعد از ثبت نام با TA و پس از ایجاد یک کلید مشترک با FS_j انجام شده است $\{RID_k, TID_k, TC_k, K_{jk}\}$را در حافظه خودش ذخیره می‌کند.طبق روابط انجام شده در ۳،۲،۱و ۴ دستگاه هوشمند با انتخاب یک کلید خصوصی m و به کمک تابع چکیده ساز یک طرفه برای انجام مراحل بعدی مقادیر $\{RID_k^*, TID_k^*, K_{kj}, TC_k^*\}$را به جای مقادیر اصلی ذخیره می‌کند اما برای انجام پروتکل ابتدا در هر مرحله ابتدا مقدار اصلی را حساب می‌کند و در پروتکل قرار می‌دهد.از این رو با سرقت دستگاه هوشمند و از آن‌جایی که مهاجم به کلید m و به مقادیر $\{RID_k, TID_k, TC_k, K_{jk}\}$ دسترسی ندارد، طرح ارائه شده در برابر حمله سرقت دستگاه هوشمند مقاوم است.

۴-۵. مقایسه سربار مخابراتی، محاسباتی و امنیتی طرح

در این مرحله طرح احراز اصالت امن سبک وزن با طرح های دیگری مانند طرح Li و همکارانش [21]، طرح Sun و همکارانش [22]، طرح Li و همکارانش [23]، طرح Hu و همکارانش [24] و طرح SAKA-FC [1] را از لحاظ ویژگی‌های امنیتی، مخابراتی و محاسباتی بررسی می‌کنیم.در جدول۲ ویژگی‌های امنیتی طرح احراز اصالت امن و پنج طرح دیگر مقایسه شده است.

جدول۲- مقایسه ویژگی امنیتی

ویژگی های امنیتی	۱	۲	۳	۴	۵	۶	۷	۸
طرح Li و همکارانش [21]	—	√	—	—	—	√	√	—
طرح Sun و همکارانش[22]	√	√	—	—	—	√	√	—
طرح Li و همکارانش [23]	√	—	—	—	—	—	√	—
طرح Hu و همکارانش [24]	√	√	—	√	—	—	—	—
طرح SAKA-FC [1]	√	√	—	√	—	√	√	√
طرح احراز اصالت امن	√	√	√	√	√	√	√	√

پی‌نوشت: ۱.احراز هویت متقابل، ۲.گمنامی، ۳. حمله کارت هوشمند سرقت شده، ۴. حمله تکرار، ۵.حمله ردیابی، ۶. حمله داخلی، ۷.توافق کلید، ۸.اثبات امنیتی به کمک AVISPA

برای بررسی سربار محاسباتی میزان سربار محاسباتی توابعی مانند جمع پیمانه‌ای را ناچیز در نظر می‌گیریم و سربار توابعی مانند تابع چکیده‌ساز،خم بیضوی،توابع تولید کننده پارامترهای بیومتریک،توابع رمزنگاری و رمزگشایی را محاسبه می‌کنیم. در جدول۳ طرح از نظر سربار محاسباتی بررسی شده و زمانی که برای انجام این محاسبات نیاز داریم نیز آورده شده است. با توجه به مقادیر آورده شده در جدول طرح احراز اصالت امن سبک وزن در مقایسه با طرح تغییر چندانی نداشته اما ویژگی‌های امنیتی بیش‌تری را مطابق آن‌چه در جدول۲ آمده دارا می‌باشد.

جدول۳- مقایسه سربار محاسباتی

نهادها	طرح Li و همکارانش [21]	طرح Sun و همکارانش[22]	طرح Li و همکارانش [23]	طرح Hu و همکارانش [24]	طرح SAKA-FC [1]	طرح احراز اصالت امن
کاربر(دستگاه‌موبایل) $U_i(MD_i)$	$1T_{enc-ibe}+1T_{sig-ibe}$	$3T_h+4T_{ecm}$	$6T_h+2T_{ecm}$	-	$16T_h+2T_{ecm}+1T_{ef}$	$17T_h+2T_{ecm}+1T_{ef}$
سرور مه(گره مه) $FS_j(FN)$	-	-	-	$2T_{exp}+2T_{pke}+1T_{pkd}$	$10T_h+3T_{ecm}$	$10T_h+3T_{ecm}$
سرور ابر $CS_l(SP/MS)$	$1T_{dec-ibe}+1T_{ver-ibe}$	$4T_h+6T_{ecm}+4T_{eca}$	$7T_h+3T_{ecm}+4T_{sed}$	$2T_{exp}+1T_{pke}+2T_{pkd}$	-	-
دستگاه هوشمند D_k	-	-	-	-	$9T_h$	$9T_h$
هزینه کلی	$1T_{enc-ibe}+1T_{sig-ibe}+1T_{dec-ibe}+1T_{ver-ibe}$	$7T_h+10T_{ecm}+4T_{eca}$	$13T_h+5T_{ecm}+4T_{sed}$	$4T_{exp}+3T_{pke}+3T_{pkd}$	$35T_h+5T_{ecm}+1T_{ef}$	$36T_h+5T_{ecm}+1T_{ef}$
تخمین زمانی	243 ms	677.75 ms	356.68 ms	7308 ms	395.95 ms	396.45 ms

برای محاسبه سربار مخابراتی تعداد پیام‌هایی که در مرحله احراز هویت و تولید کلید نشست در طرح طرح احراز اصالت امن سبک وزن ارسال شده است را محاسبه می‌کنیم، در این مرحله پیام ۲،۱و۳ ارسال می‌شود که با تعداد پیام ارسالی در طرح SAKA-FC [1] تفاوتی ندارد اما در پیام۱ مقدار x_i وy_i نیز ارسال می‌شود به همین دلیل تعداد بیت‌های ارسال شده افزایش یافته است. در جدول۴ طرح از نظر سربار مخابراتی بررسی شده و تعداد بیت‌های ارسالی نیز آورده شده است.

جدول۴- مقایسه سربار مخابراتی

طرح‌های مورد بررسی	مجموع تمامی پیام‌ها	مجموع تمامی بیت‌ها
طرح Li et al	۲	۱۴۲۸۰
طرح Sun et al	۲	۲۶۸۸
طرح Li et al	۴	۴۱۶۰
طرح Hu et al	۳	۷۱۶۰
طرح SAKA-FC	۳	۲۸۱۶
طرح احراز اصالت امن	۳	۳۱۰۴

۵. تحلیل امنیتی طرح

- مقاوم در برابر حمله تکرار: همان‌طور که اشاره شد تمامی نهادها در هر مرحله از یک مهر زمانی استفاده می‌کنند و بعد از ارسال هر پیام نیز اعتبار مهر زمانی با کمک ΔT بررسی می‌شود به همین دلیل مهاجم نمی‌تواند از مهرزمانی مجددا استفاده کند و طرح ارائه شده در برابر حمله تکرار مقاوم است.

- مقاوم در برابر حمله دزدیده شدن دستگاه هوشمند:طبق بررسی انجام شده در بخش ۴-۴ اگر دستگاه هوشمند مقادیر اصلی شناسه موقت،شناسه مستعار و کلید را ذخیره نکند و به جای آن $\{RID_k^*, TID_k^*, K_{kj}, TC_k^*\}$ را ذخیره کند،طرح در برابر حمله دزدیده شدن دستگاه هوشمند مقاوم می‌شود.

- مقاوم در برابر حمله داخلی:با فرض این‌که مهاجم RID_i و P_i را در اختیار داشته باشد و بخواهد به جای کاربر وارد سیستم شده و به یک کلید مشترک برسد،برای پیدا کردن ID_i به کلید مخفی کاربر یعنی S نیاز دارد که ذخیره نشده است، با این که کاربر P_i و G را در اختیار دارد نمی‌تواند d_i را محاسبه کند زیرا در تولید آن از خم بیضوی استفاده شده است $P_i=d_i.G$ ، نتیجه می‌گیریم طرح در مقابل حمله داخلی مقاوم است.

- گمنامی:از آن‌جایی که در تمامی مراحل به جای استفاده از شناسه اصلی نهادها از شناسه‌های مستعار $\{RID_i,RID_k,RID_l,RID_j\}$ و شناسه‌های موقت $\{TID_i,TID_k,TID_l,TID_j\}$ استفاده می‌شود و در تولید این شناسه‌ها از توابع چکیده ساز یک طرفه استفاده شده است نمی‌توان به شناسه اصلی دست پیدا کرد و طرح گمنامی را حفظ می‌کند.

- مقاوم در برابر حمله جعل کاربر:فرض می‌کنیم مهاجم بخواهد پیام۱ را به جای کاربر تولید کند ابتدا یک مهر زمانی و عدد تصادفی جدید انتخاب می‌کند، می‌تواند R_u را تولید کند اما برای تولید $\{E_u,F_u,a_u\}$ نیاز به مقادیر $\{RID_i,d_i,TC_i\}$ دارد و از آن‌جایی که کاربر این مقادیر را ذخیره نمی‌کند، مهاجم قادر به تولید پیام۱ نیست و ثابت می‌شود طرح در برابر حمله جعل کاربر مقاوم است.

- مقاوم در برابر حمله ردیابی:همان‌طور که در بخش۴-۴ اشاره شد با کمک عدد تصادفی x_i می‌توان آسیب‌پذیری طرح در برابر حمله ردیابی را رفع کرد.

- اثبات امنیتی به کمک AVISPA: برای تحلیل و ارزیابی مرحله احراز هویت و تولید کلید و بررسی امنیتی طرح به کمک نرم‌افزار AVISPA نیاز داریم. برای این‌که بتوان از این نرم‌افزار استفاده کرد ابتدا تعاریف و مقدمات لازم برای هر نهاد را به صورت جداگانه تعریف می‌کنیم. در شکل ۸ مشخصات کاربر U_i آورده شده است، در شکل ۹ و ۱۰ نیز به ترتیب مشخصات سرور مه FS_j و دستگاه هوشمند D_k به تفکیک تعریف شده است.از آن‌جایی که در نرم‌افزار AVISPA نمی‌توانیم تمامی حملات را بررسی کنیم تنها به بررسی حمله تکرار و حمله مردی در میانه اشاره می‌کنیم، نتیجه طرح که به کمک CL-AtSe[1] و OFMC[2] بررسی شده است در شکل ۱۱ نشان داده شده و می‌توان دید که طرح در برابر هردو این حملات مقاوم است.

[1] CL-based Model-Checker
[2] On-the-Fly Model-Checker

```
role fogsever (Ui, TA, FSj, Dk : agent, H: hash_func,
        Sad, Rcv: channel(dy))
played_by FSj
def=
local State: nat,
    PWi, BI0i, S, Di,1Di, G, TSu, Ru, Rul: text,
    F: hash_func, K, RTSi, RTSj, P, IDj, TSf, Mk, Rf, Pf: text,
    RIDi2, Kjk, RIDkI, Kuf, Gj, Hj: text
coast spl, sp2, sp3, sp4, sp5, ui_fsj_ru, ui_fsj_tsu,
    fsj_dk rf, fsj_dk tsf: protocol_id
init State := 0
transition
% Login and authentication phases
% Recive message Msgl from Ui via public channel
1. State = 0 A Rcv(xor(H(S' .1Di), H(H(H(K.RTSj'.IDj)).TSu')).F(Ru'.G).
        F(Di' .Ru').Di'.TSu'), H(Xi), Au'1 := F(Au'.Xi)
        H(H(H(K.RTV.IDj)).H(S'.1Di))).
        xor(H(K.IDk),H(H(H(K.RTSj'.1Dj)).TSu')).TSul =1>
State' := 3 A secret({IDi,S',Di',PWi,BIOi }, spl, {UM
A secret({K,RTSi',RTSj' }, sp2, {TA}) A secret((IDj}, sp3, (TA,FSj})
A secret({Kjk), sp4, (FSj,D1{}) A secret({IDk), sp5, (TA,Dk))
A TSf := new() A RI' := new() A Pf := F(Rf'.G)
A RIDi2' := xor(H(H(S'.1Di)),H(Kjk.H(K.Mok).TSf ))
A RlDkl ' := xor(H(K.IDk),H(Kjk.TSf )) A Kuf := F(Rf .F(Ru'.0))
A Gj' := xor(H(Kjk.H(K.IDk).TSf ),H(Kuf .H(H(H(S' .1Di).
        K.RTSi').Di'.TSu').H(H(S'.IDi))))
A Hj' := H(H(H(S'IDi)).H(K.IDk).Gy .Pf .TSf)
% Send message Msg2 to Dk via public channel
        A Snd(RIDi2'.R1Dkl '.Gj'.Hj'.Pf .TSfl )
% FSj has freshly generated the values rf and TSf for Dk
        A witness(FSj, Dk, fsj_dk_rf, Rf )
        A witness(FSj, Dk, fsj_dk_tsf, TSf )
% FSj's acceptance of the values ru and TSu generated for PSj by Ui
        A request(Ui, FSj, ui_fsj_ru, Ru')
        A request(Ui, FSj, ui_fsj_tsu, TSu')
end role
```

شکل۹-مشخصات سرور مه FS_j

```
role smartdevice (Ui, TA, FSj, Dk : agent, H. Imli_fune,
        Sad, Rcv: channel(dy))
played_by Dk
def=
local State: nat,
    S, Di, 111/i, G, IDk, K, TSk, Kjk: text, F: hash_func,
    RTSi, TSu, Rf, Ru, TSf, Kuf, Rk, Ij, RIDk2: text,
    Mk, RTSk, Nk, SICki, PWi, BIOi, RTSj, IDj: text
coast spl, sp2, sp3, sp4, sp5, fsj_dk rf, fsj_dk tsf,
    dk_ui_rk, dk_ui_tsk: protocoLid
Mit State := 0
transition
% Login and authentication phases
% Recive message Msg2 from FSj via public channel
1. State = 0 A Rcv(xor(H(H(S'.100),H(Kjk.H(IC.1Dk).TSF)).
   xor(H(K.1Dk),H(Kjk.TSP)).
   xor(H(Kjk.H(K.IDk).TSP),H(ICuf .H(H(H(S'.1Di).
   K.RTSi').DP.TSu').H(H(S'.11Di)))).H(H(H(S'.1Di)).
   H(K.IDk).xor(H(Kjk.H(K.IDk).TSP),H( F(Rf.F(Ru'.0)).
   H(H(H(S'.1D0.K.RTSi').DP.TSul.H(H(S'IDB))).
   F(Rf.G).TSO.F(RP.G).TSP) =1>
State' := 4 A secretalDi,S',DP,PWLBIOi}, spi, {Ui})
A secret({1C,RTSP,RTSy 1, sp2, {TA}) A secret({IDj}, sp3, (TA,FSj})
A secret({Kjk), sp4, {FSj,Dk}) A secret((lDk), sp5, (TA,Dk})
        A Rk' := new() A TSk' := new()
        A Ij' := H(F(Rf .F(Ru'.G)).H(H(H(S'.100.K.RTSi').
            Di'.TSu').H(H(S'.IDi)))
        A RIDk2' := xor(H(K.IDk),H(H(H(S'.1Di)).TS10)
        A SKki' := H(Ir.H(H(K.RTSk'.1Dk).Rk').TSk')
        A Mk' := xor(H(H(K.RTSk'.1Dk).Rkl,
            H(H(K.Mk).H(H(S'.1D0).TSkl)
        A Nk' := H(SKki'.F(Rf'.G).TSk')
% Send message Msg3 to Ui via open channel
        A Snd(RIDk2'.MIC.NIc'.F(Rf .G).TSk')
% Dk has freshly generated the values rk and TSk for Ui
        A witness(Dk, Ui, dk_ui_rk, Rk')
        A witness(Dk, Ui, dk_ui_tsk,TSk')
% Dk's acceptance of the values rf and TSf generated for Dk by FSj
        A request(FSj,Dk,fsj_dk_rf, Rf)
        A request(FSiDk,fsj_dk_tsf,TSP)
end role
```

شکل۱۰-مشخصات دستگاه هوشمند D_k

```
role user (Ui, TA, FSj, Dk : agent, H: hash_func,
        SKuita : symmetricikey, Snd, Rev: channel(dy))
played by Ui
def=
 local State: nat,
    S, Di, IDi, RIDi, Pi, PWi, BIOi,Xi,Yi P, G, K, IDj, IDk: text,
    RTSi, TIDj, TSu, Ru, Rul, Au, RIDil, RTSj, Eu, Fu: text, RTSk,
    TSk, Kjk, Rk, Fk, Rf: text, F: hash_func
coast spl, sp2, sp3, sp4, sp5, ui_fsj_ru, ui_fsj_tsu,
    dlc ui_rk, dk ui_tsk : protocol_id
init State := 0
transition
 1. State = 0 A Rcv(start) =1>
% User registration phase
State' := I A S' := newO A Di' := newO
        A RIDi1 := H(S.EDi) A Pi' := F(Di'.G)
% Send registration request to the TA securely
        A Snd({RIDV.Pi' }_SKuita)
        A secret((IDi,S',Di',PWLBIOi), spl, {Ui))
% Receive registration reply from the TA securely
 2. State = I A Rcv({1.1(H(S'IDi).K.RTSi').TIDj.
            H(H(K.RTVIDj)))_SKuita) =1>
State' := 2 A secret({K,RTSi',RTSj' ), sp2, {TA))
        A secretaIDD, sp3, (TA,FSD)
A secret({Kjk}, sp4, (FSj,Dk)) A secret({1Dk}, sp5, (TA,Dk)) %
Login & authentication phases
        A TSu' := new() A Ru' := new()
        A Rul' := F(Ru'.G) A Au' := F(Di'.Xi.Ru'.P)
        A Xi:=new() A Yi:=H(Xi)
        A Ft1Di1' := xor(H(S' .IDi), H(H(H(K.RTSj'.IDj)).TSu')) A
            Eu' := xor(H(H(H(S%1Di).K.RTSil.Di' .TSu'),
            H(H(H(K.RTSr .IDj)).H(S' .IDi)))
        A Fu' := xor(H(K.IDk),H(H(H(K.RTSjUDj)).TSu')) %
Send message Msg 1 to FSj publicly
        A Snd(RIDi 1 ' .Ru I ' .Au' .Eu ' .Fu' .Xi.Yi.TSul
% Ui has freshly generated the values ru and TSu for FSj A
         witness(Ui, FSj, uLfsj_ru, Ru')
         A witness(Ui, FSj, ui_fsj_tsu, TSu')
% Receive message Msg3 from Dk publicly
 3. State = 2 A Rcv(xor(H(K.IDk),H(H(H(S'IDB).TSV)).
    xor(H(H(K.RTSk'.1Dk).Rle), H(H(K.IDk).H(H(S'IDi)).TSkl) .
    H(H(H(F(RP.F(Ru'O)).H(H(S .1Di).K.RTSi').Dr.TSu').
    H(S'.IDi)) .H(H(K.RTSk'.IDk). Rk').TSk').
  F(Rf .G).TSIC).F(Rf .G).TSk') =1>
% Ui's acceptance of the values rk and TSk generated for Ui by Dk State'
:= 3 A request(Dk, Ui, dk ui_rk, Rk')
        A request(Dk, Ui, dk ui_tsk,TSk')
end role
```

شکل۸-مشخصات کاربر U_i

```
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  C:\progra-1 \SPANN\testsuite
     \results \auth.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
 ParseTime: 0.00s
 SearchTime: 13.53s
 VisitedNodes: 1432 nodes
 Depth: 8 plies
```

```
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL
PROTOCOL
C:\progra-1 \SPANN\testsuite
    \results \auth.if
GOAL
  as_specified
BACKEND
  CL—AtSe
STATISTICS
 Analysed: 8 states
 Reachable: 0 states
 Translation: 0.60 seconds
 Computation: 0.00 seconds
```

شکل۱۱-تحلیل نتایج شبیه‌سازی

۶. نتیجه‌گیری

رایانش مه به عنوان یک رایانش سریع‌تر در کنار رایانش ابری از نیازهای ضروری در استفاده از اینترنت اشیا می‌باشد. در این مقاله یک طرح احراز اصالت سبک وزن امن مبتنی بر توابعی نظیر تابع چکیده ساز و جمع پیمانه‌ای پیشنهاد گردید. طرح از نظر ویژگی‌های امنیتی، سربار محاسباتی و مخابراتی در بخش ۴-۵ بررسی شده و مشاهده می‌شود در مقایسه با طرح‌های دیگر دارای سربار محاسباتی و مخابراتی بالایی نمی‌باشد. همان‌طور که گفته شد طرح ارائه شده در برابر حملات شناخته شده‌ای همچون حمله ردیابی، حمله کارت هوشمند سرقت شده، حمله تکرار و بسیاری از حملات دیگر مقاوم است و دارای احراز هویت و گمنامی نیز می‌باشد. برای اثبات امنیتی طرح ابتدا در بخش ۵ طرح را در برابر حملات مختلفی که بیان شد، بررسی کردیم و سپس به کمک نرم‌افزار AVISPA نیز امنیت رسمی[1] طرح بررسی شده است. به طور کلی طرح احراز اصالت سبک وزن امن نسبت به سایر طرح‌ها دارای ویژگی‌های امنیتی بیش‌تری می‌باشد و سربار محاسباتی و مخابراتی آن نیز نسبت به طرح‌های مشابه کم‌تر است.

۷. مراجع

[1] M. Wazid, A. Kumar Das, N. Kumar and A. V. Vasilakos, "Design of secure key management and user authentication scheme for fog computing services," *Future Generation Computer Systems*, vol. 91, pp. 475-492, 2019.

[2] Y. Yang, L. Wu, G. Yin, L. Li and H. Zhao, "A Survey on Security and Privacy Issues in Internet-of-Things," *IEEE INTERNET OF THINGS JOURNAL*, vol. 4, no. 5, pp. 1250 - 1258, 2017.

[3] Z. Fu, K. Ren, J. Shu, X. Sun and F. Huang, "Enabling Personalized Search over Encrypted Outsourced Data with Efficiency Improvement," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 9, pp. 2546 - 2559, 2015.

[4] S. Basudan, X. Lin and K. Sankaranarayanan, "A Privacy-preserving Vehicular Crowdsensing based Road surface Condition Monitoring System Using Fog Computing," *IEEE Internet of Things Journal*, 2017.

[5] R. Mahmud and R. Buyya, "Fog Computing: A Taxonomy, Survey and Future Directions," *Technology, Communications and Computing*, pp. 103-130, 2017.

[6] A. V. Dastjerdi and R. Buyya, "Fog Computing: Helping the Internet of Things Realize Its Potential," *Computer*, vol. 49, no. 8, pp. 112-116, 2016.

[7] S. S. Vikas, K. Pawan , A. K. Gurudatt and G. Shyam, "Mobile cloud computing: Security threats," in *International Conference on Electronics and Communication Systems (ICECS)*, Coimbatore, India, 2014.

[8] M. Kumar, "A New Secure Remote User Authentication Scheme with Smart Cards," *International Journal of Network Security*, vol. 11, no. 2, pp. 88-93, 2010.

[1] Formal Security

[9] S. Khan, S. Parkinson and Y. Qin, "Fog computing security: a review of current applications and security solutions," *Journal of Cloud Computing:Advances, Systems and Applications,* vol. 6, no. 19, 2017.

[10] M. Mukherjee, R. Matam, L. Shu, L. Maglaras, M. A. Ferrag, N. Choudhury and V. Kumar, "Security and Privacy in Fog Computing: Challenges," *IEEE Access,* vol. 5, pp. 19293 - 19304, 2017.

[11] J. Li, J. Jin, D. Yuan, M. Palaniswami and K. Moessner, "EHOPES: Data-centered Fog platform for Smart Living," in *International Telecommunication Networks and Applications Conference (ITNAC),* Sydney, NSW, Australia, 2015.

[12] M. Alshahrani and I. Traore, "Secure mutual authentication and automated access control for IoT smart home using cumulative Keyed-hash chain," *Journal of Information Security and Applications,* vol. 45, pp. 156-175, 2019.

[13] J. Kang, R. YU, X. Huang and Y. Zhang, "Privacy-Preserved Pseudonym Scheme for Fog Computing Supported Internet of Vehicles," *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS,* 2017.

[14] H. Wang, Z. Wang and J. Domingo-Ferrer, "Anonymous and secure aggregation scheme in fog-based public cloud computing," *Future Generation Computer Systems,* vol. 78, pp. 712-747, 2017.

[15] H. A. Al Hamid, S. M. Mizanur Rahman, M. S. Hossain, A. Almogren and A. Alamri, "A Security Model for Preserving the Privacy of Medical Big Data in a Healthcare Cloud Using a Fog Computing Facility With Pairing-Based Cryptography," *IEEE Access,* vol. 5, pp. 22313-22328, 2017.

[16] J. Xiaoying , H. Debiao, N. Kumar and C. Kim-Kwang Raymond , "Authenticated key agreement scheme for fog driven IoT healthcare system," *Wireless Networks,* pp. 1-14, 2018.

[17] . I. Maged Hama, "Octopus: An Edge-Fog Mutual Authentication Scheme," *International Journal of Network Security,* vol. 18, pp. 1089-1101, 2016.

[18] R. Roman, J. Lopez and M. Mambo, "Mobile edge computing, Fog et al.: A survey and analysis of security threats and challenges," *Future Generation Computer Systems,* vol. 78, pp. 680-698, 2018.

[19] I. Stojmenovic and S. Wen, "The Fog computing paradigm: Scenarios and security issues," in *2014 Federated Conference on Computer Science and Information Systems,* Warsaw, Poland, 2014.

[20] I. Stojmenovic, S. Wen, X. Huang and H. Luan, "An overview of Fog computing and its security issues," *Concurrency and Computation Practice and Experience,* vol. 28, no. 10, pp. 2991-3005, 2016.

[21] H. Li, Y. Dai, L. Tian and H. Yang, "Identity-Based Authentication for Cloud Computing," *Lecture Notes in Computer Science book series,* vol. 5931, pp. 157-166, 2009.

[22] H. Sun, Q. Wen, H. Zhang and Z. Jin, "A novel remote user authentication and key agreement scheme for mobile client-server environment," *Appl. Math. ,* vol. 7, no. 4, pp. 1365-1374, 2013.

[23] H. Li, F. Li, C. Song and Y. Yan, "Towards Smart Card Based Mutual Authentication Schemes in Cloud Computing," *KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS*, vol. 9, no. 7, pp. 2719-2735, 2015.

[24] P. Hu, H. Ning, T. Qiu, H. Song, Y. Wang and X. Yao, "Security and Privacy Preservation Scheme of Face Identification and Resolution Framework Using Fog Computing in Internet of Things," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1143 - 1155, 2017.