مقاله پژوهشی

احراز هویت سبک‌وزن متقابل مداوم بر اساس اولویت‌بندی گره‌ها با استفاده از نرخ ترافیک در اینترنت اشیا

رضا سرابی میانجی، سام جبه‌داری و ناصر مدیری

چکیده: امروزه میلیاردها دستگاه از طریق اینترنت اشیا و در اغلب موارد از طریق ارتباطات ناامن به هم متصل شده‌اند، بنابراین مسایل امنیتی و حریم خصوصی این دستگاه‌ها به عنوان یک نگرانی عمده مطرح است. با توجه به محدودیت منابع دستگاه‌های اینترنت اشیا، راه حل‌های امنیتی این محیط از نظر پردازش و حافظه باید امن و سبک‌وزن باشند. با این حال، بسیاری از راه حل‌های امنیتی موجود به طور خاص در زمینه احراز هویت به دلیل محاسبات زیاد برای اینترنت اشیا مناسب نیستند و نیاز به یک پروتکل احراز هویت سبک‌وزن برای دستگاه‌های اینترنت اشیا احساس می‌شود. در این مقاله، یک پروتکل احراز هویت سبک‌وزن متقابل بین گره‌ها با منابع محدود و سرور در اینترنت اشیا معرفی شده است که از اولویت‌بندی گره‌ها بر اساس نرخ ترافیک استفاده می‌کند. این طرح به دلیل استفاده از عملیات XOR و Hash سبک می‌باشد. طرح پیشنهادی در برابر حملات سایبری مانند استراق سمع و حمله تلاش مجدد مقاوم است و همچنین با استفاده از ابزار AVISPA و در مدل تهدید Dolev-Yao امن می‌باشد. ریسک‌های امنیتی این روش در مقایسه با روش‌های سبک‌وزن دیگر کم است. در ضمن طرح پیشنهادی باعث کاهش هزینه محاسباتی، حفظ حریم خصوصی از طریق گمنامی گره‌ها و فراهم‌آوردن رازداری رو به جلو می‌شود. در روش ما، هزینه زمانی احراز هویت نسبت به روش‌های بررسی‌شده ۱۵٪ کاهش یافته است.

کلیدواژه: **احراز هویت سبک‌وزن، احراز هویت مداوم، اینترنت اشیا، حریم خصوصی.**

۱- مقدمه

اینترنت اشیا [۱] شامل تعداد زیادی از اشیای فیزیکی متصل به هم از طریق شبکه است و در بخش‌های مختلف صنعت از جمله مراقبت‌های بهداشتی، حمل و نقل، مدیریت انرژی کارخانه و کنترل لوازم خانگی، توسعه و پیاده‌سازی می‌شود [۲] تا [۴]. کاربردهای اینترنت اشیا بیشتر مربوط به زندگی روزمره انسان‌ها می‌باشد و بنابراین حفظ حریم خصوصی و امنیت باید بیشتر مورد توجه قرار گیرد [۵]. یکی از موضوعات مهم در اینترنت اشیا، امنیت اطلاعاتی گره‌ها در هنگام تبادل اطلاعات می‌باشد و همچنین وجود سخت‌افزارهای ناهمگن با پیچیدگی‌های مختلف، چالش‌های امنیتی بیشتری را برای محیط اینترنت اشیا ایجاد کرده است. از جمله مسایل امنیتی در این محیط می‌توان به حفاظت از داده‌ها، کنترل دسترسی و احراز هویت دستگاه‌ها اشاره کرد [۶] و [۷].

احراز هویت، اصلی‌ترین مکانیزم امنیتی در اینترنت اشیا می‌باشد و هدف آن، مشخص‌کردن درستی هر موجودیت مثل دستگاه یا کاربر است [۸]. بر اساس تحقیقات انجام‌شده، احراز هویت کاربرها به دو دسته احراز هویت استاتیک و احراز هویت مداوم تقسیم می‌شود [۹].

در احراز هویت استاتیک، برای ورود کاربر به سرور، در ابتدای ارتباط یک فرایند کامل انجام می‌شود [۹] تا [۱۰]. به طور کلی، مشخصه خاصی مانند کلمه عبور، کد رمزدار، کارت هوشمند، توکن امنیتی، ویژگی‌های چهره و اثر انگشت، به عنوان ورودی در یک درخواست احراز هویت استفاده می‌شود [۱۱] تا [۱۲]. احراز هویت مداوم می‌تواند به طور مداوم، اعتبار یک کاربر را در زمان استفاده از دستگاه، بررسی و تأیید کند. اما باید دقت کرد که احراز هویت مداوم، جایگزینی برای احراز هویت استاتیک نیست [۱۲] و در حقیقت، این روش برای تقویت توان امنیتی احراز هویت استاتیک استفاده می‌شود. رویکردهای موجود برای احراز هویت مداوم کاربر، استفاده از ویژگی‌های رفتاری مانند نحوه حرکت موس و ضربه‌زدن به صفحه کلید برای بررسی مداوم صحت یک کاربر می‌باشد [۱۰]، [۱۳] و [۱۴].

تحقیقات اندکی در مورد احراز هویت مداوم دستگاه به دستگاه در محیط اینترنت اشیا انجام شده و از سوی دیگر، بیشتر گره‌های موجود در این محیط از منابع محاسباتی و ذخیره‌سازی محدودی برخوردار هستند [۱۵]. در ضمن این گره‌ها نمی‌توانند محاسبات پیچیده‌ای مانند عملیات رمزگذاری و رمزگشایی را انجام دهند [۱۶] و [۱۷]. بنابراین در این تحقیق، احراز هویت مداوم دستگاه به دستگاه بین گره‌ها و سرور در محیط اینترنت اشیا ارائه می‌گردد. در ابتدا گره‌ها و سرور، یک احراز هویت استاتیک انجام خواهند داد که شامل یک فرایند کامل است. سپس برای یک بازه زمانی، احراز هویت بین گره‌ها و سرور به صورت احراز هویت مداوم انجام می‌گیرد.

هدف از این مقاله، استفاده از احراز هویت استاتیک و مداوم برای احراز هویت دستگاه به دستگاه بر اساس اولویت‌بندی گره‌ها می‌باشد. بنابراین گره‌ها در سه اولویت قرار می‌گیرند و برای هر اولویت، یک بازه زمانی در نظر گرفته می‌شود. همه اطلاعات در اختیار سرور قرار می‌گیرد. هر یک از اعضای گروه اولویت در ابتدای بازه زمانی، به صورت استاتیک احراز هویت می‌شوند و یک توکن ایجاد می‌گردد و تا اتمام بازه زمانی آن اولویت، با استفاده از توکن، احراز هویت مداوم انجام می‌گردد. احراز هویت مداوم، سریع‌تر از احراز هویت استاتیک می‌باشد و بنابراین احراز هویت کل، سریع‌تر انجام می‌گردد.

این مقاله در تاریخ ۲۱ اردیبهشت ماه ۱۴۰۰ دریافت و در تاریخ ۵ آذر ماه ۱۴۰۰ بازنگری شد.

رضا سرابی میانجی، گروه مهندسی کامپیوتر، واحد تهران شمال، دانشگاه آزاد اسلامی، تهران، ایران، (email: reza.sarabi@gmail.com).

سام جبه‌داری (نویسنده مسئول)، گروه مهندسی کامپیوتر، واحد تهران شمال، دانشگاه آزاد اسلامی، تهران، ایران، (email: s_jabbehdari@iau-tnb.ac.ir).

ناصر مدیری، گروه مهندسی کامپیوتر، دانشکده مهندسی برق و کامپیوتر، واحد زنجان، دانشگاه آزاد اسلامی، زنجان، ایران، (email: nassermodiri@chmail.ir).

این مقاله شامل ویژگی‌های زیر است:

۱) از ترکیب روش‌های احراز هویت استاتیک و مداوم با اولویت‌بندی گره‌ها استفاده شده است. اولویت‌بندی بر اساس ترافیک گره‌ها انجام می‌گیرد، سپس برای هر اولویت یک ضریب اهمیت در نظر گرفته می‌شود که با کمک آن بازه زمانی مورد نیاز برای احراز هویت استاتیک هر اولویت مشخص می‌گردد.

۲) اعضای اولویت در ابتدای بازه زمانی به صورت استاتیک احراز هویت می‌شوند و یک توکن ایجاد می‌گردد و تا اتمام بازه زمانی، از احراز هویت مداوم استفاده می‌شود.

۳) انتخاب بازه زمانی انجام احراز هویت استاتیک به کمک اولویت‌بندی گره‌ها بر اساس اهمیت ترافیک انجام می‌گیرد.

۴) برای کاهش هزینه محاسباتی در طرح پیشنهادی، احراز هویت استاتیک و مداوم فقط از عملگرهای محاسباتی سبک Hash و XOR استفاده می‌کند تا به راحتی از این طرح در گره‌ها با منابع محدود استفاده شود.

۵) احراز هویت متقابل سبک‌وزن و توافق کلید ارائه داده شده و هر زوج دستگاه می‌توانند یکدیگر را احراز هویت نمایند.

۶) از ابزار AVISPA برای تأیید امنیتی طرح و آنالیز امنیتی استفاده کردیم.

۷) طرح ما دارای ریسک امنیتی پایین بوده و هزینه محاسباتی کمی دارد.

۸) بدون استفاده از رمزگذاری آسنکرون، رازداری رو به جلو را تضمین کرده‌ایم.

این مقاله به شرح زیر سازماندهی شده است: تحقیقات مربوط به کارهای مرتبط با احراز هویت اینترنت اشیا در بخش ۲ مورد بررسی قرار می‌گیرد. در بخش ۳، روش پیشنهادی ارائه شده است. در بخش ۴، شبیه‌سازی و تجزیه و تحلیل‌های امنیتی صورت می‌پذیرد و بخش ۵ آنالیز امنیتی را ارائه می‌دهد. در بخش ۶ ارزیابی و مقایسه عملکرد بیان می‌شود و نتیجه‌گیری نهایی در بخش ۷ مورد بررسی قرار گرفته است.

۲- کارهای مرتبط

در این بخش، در مورد کارهای مرتبط با احراز هویت برای اینترنت اشیا بحث شده است. در [۱۸] در سال ۲۰۱۳ یک طرح امنیتی احراز هویت دوطرفه برای اینترنت اشیا بر اساس DTLS [۱۹] پیشنهاد شد که از رمزگذاری نامتقارن مبتنی بر RSA و گواهی X.۵۰۹ استفاده می‌کرد. با توجه به این که این طرح برای ایجاد یک جلسه، به دست‌تکانی نیاز دارد، در نتیجه برای اجرای این روش، نیاز به مصرف انرژی و فضای ذخیره زیادی از حسگرهایی می‌باشد که با محدودیت منابع روبه‌رو هستند. مرجع [۲۰] یک پروتکل احراز هویت غیر قابل ردیابی در اینترنت اشیا ارائه داده که در این روش از توابع Hash و عملیات XOR و از اعداد ترتیبی و تصادفی برای تولید یک هویت مستعار مستقل استفاده شده است. روش پیشنهادی، علاوه بر این که معتبربودن حسگرها را تضمین می‌کند، از ناشناس‌بودن هویت و عدم ردیابی نیز پشتیبانی می‌نماید. مرجع [۲۱] یک طرح احراز هویت مبتنی بر مکان را در محیط‌های اینترنت اشیا ارائه داده که این پروتکل از اطلاعات محیط دستگاه‌ها برای احراز هویت استفاده می‌کند. این طرح به طور مداوم باید اطلاعات محیط را از دستگاه‌های اینترنت اشیا جمع‌آوری نماید. در [۲۲] یک طرح احراز هویت ناشناس اینترنت اشیا بر روی ECC سبک‌وزن ارائه شده که شامل دو مرحله اصلی ثبت نام و احراز هویت می‌باشد. مقایسه‌ای بین ECC و RSA با استفاده از زمان مورد نیاز برای رمزگذاری و رمزگشایی با استفاده از اندازه کلیدهای متفاوت انجام شده است. در [۲۳] یک طرح احراز هویت برای سیستم‌های اینترنت اشیا با استفاده از زنجیره بلوکی به نام Bubbles-of-Trust ارائه شده است. ایده به این صورت می‌باشد که دستگاه‌ها را به ناحیه‌های مجازی به نام حباب تقسیم می‌کنند و اعضا در آن، همدیگر را شناسایی و به هم اعتماد می‌کنند. سپس ارتباطات بین دستگاه‌های مختلف، کنترل می‌گردد و با استفاده از زنجیره بلوکی پیاده‌سازی شده با استفاده از اتریوم اعتبارسنجی می‌شوند. در [۲۴] یک طرح احراز هویت ناشناس برای ایجاد ارتباطات امن برای محیط‌های خانه هوشمند ارائه شده است. طرح پیشنهادی به طور قابل توجهی، امنیت بهتر و ویژگی‌های عملکردی بیشتری دارد. در [۲۵] یک طرح احراز هویت کارامد و ناشناس برای محیط خانه هوشمند با استفاده از ECC پیشنهاد شده که در آن از عدد تصادفی برای مقاومت در برابر حمله مجدد استفاده گردیده است.

در سال‌های اخیر برای احراز هویت مداوم، روش‌هایی ارائه شده است. هدف از این روش‌ها، کمک به دستگاه‌ها برای احراز هویت مداوم کاربر به منظور جلوگیری از جعل هویت آنها یا استفاده غیر قانونی از دستگاه‌ها است. این روش‌ها برای احراز هویت ارتباط کاربر به دستگاه پیشنهاد شده‌اند و در بیشتر آنها از ویژگی‌های رفتاری برای فرایند احراز هویت مداوم استفاده گردیده است. در [۲۶] یک مکانیسم احراز هویت مداوم ارائه شده که هویت کاربر، بارها بر اساس سبک ضربات صفحه کلید توسط کاربر بررسی می‌شود. روش پیشنهادی [۲۶]، چندین رفتار را از کاربر اصلی جمع‌آوری می‌کند تا مجموعه ویژگی‌های مربوط به کاربر اصلی را ایجاد نماید و از این مجموعه‌ها به عنوان مرجع استفاده کند. در [۲۷] یک مکانیزم احراز هویت مبتنی بر گواهی‌نامه ارائه شده که از احراز هویت دو فاز بین سنسور و کاربران نهایی استفاده می‌کند تا یکدیگر را تأیید نمایند و با هم ارتباط برقرار کنند. پروتکل پیشنهادی از گره‌های با منابع محدود و ناهمگون پشتیبانی می‌کند و دارای قابلیت مقیاس‌پذیری است.

تحقیقات دیگری وجود دارد که ویژگی‌های مختلف کاربر را برای ساخت احراز هویت مداوم استفاده می‌کنند. یک روش احراز هویت کاربر مداوم با استفاده از الگوهای ترکیبی همچون صفحه کلید، ماوس و واسط گرافیکی کاربر برای دستیابی به دقت احراز هویت بالاتر در [۲۸] پیشنهاد شده است. مرجع [۱۰] یک طرح احراز هویت مداوم کاربر را بر اساس الگوهای حرکات انگشت کاربر در صفحه لمسی دستگاه معرفی کرده است. همچنین مدل‌های اصلاح‌شده تصمیم‌گیری مارکوف برای استفاده متفاوت از محتوا در [۱۰] ارائه شده است. ویژگی‌های بیومتریک نرم از جمله پوست صورت و رنگ لباس برای ایجاد مکانیسم احراز هویت مداوم کاربر در [۲۹] به کار گرفته شده است. همچنین در [۳۰] یک فریم‌ورک برای احراز هویت کاربران بر اساس ویژگی‌های بیومتریک ارائه گردیده که تحمل‌پذیری بالایی در برابر تغییر وضعیت کاربر در جلوی کامپیوتر دارد. در سال ۲۰۱۲، در [۳۱] روش احراز هویت مداوم کاربران بر اساس مکانیسم تشخیص عنبیه کاربر توسط محققان پیشنهاد شد. در این طرح توانایی اضافه‌نمودن رمز عبور به این روش وجود دارد تا یک راه حل احراز هویت چندعاملی ایجاد شود. مرجع [۲۹] یک مکانیسم احراز هویت مداوم را برای محافظت از حریم شخصی کاربرانی که عینک‌های هوشمند دارند، معرفی می‌کند. مهم‌ترین ویژگی‌هایی که این روش از آنها استفاده می‌کند، حرکت‌های لمسی انگشت و دستورات صوتی کاربران است. یکی از ویژگی‌های منحصربه‌فرد کاربران که [۳۲] در مکانیزم پیشنهادی خود برای احراز هویت استفاده نموده است، به کارگیری الگوهای امواج مغزی کاربر در فرایند احراز هویت می‌باشد.

جدول ۱: روش‌های پیشین احراز هویت در اینترنت اشیا.

ردیف	روش احراز هویت	معایب
۱	مبتنی بر صدور گواهی‌نامه	نیاز به دست‌تکانی دارد، هزینه مصرف بالا و فضای ذخیره‌سازی زیاد دارد، محدودیت منابع وجود دارد.
۲	مبتنی بر رمزنگاری	نیاز به گواهی احراز هویت دارند. زمان زیاد برای ایجاد نشست نیاز است. احراز هویت اضافی برای کاربران ایجاد می‌شود. دستگاه‌ها نیاز به توانایی انجام رمزنگاری دارند. فرایند پیچیده محاسباتی نیاز است.
۳	مبتنی بر روش‌های غیر رمزنگاری	جمع‌آوری مداوم اطلاعات از محیط انجام می‌گیرد.

بر اساس مطالعات صورت‌گرفته، تحقیقات بسیار اندکی در مورد احراز هویت مداوم بین دستگاه‌ها در محیط اینترنت اشیا وجود دارد. مرجع [۳۳] یک روش احراز هویت سبک‌وزن را برای محیط WIFI ارائه داده و باعث کاهش مصرف باتری می‌گردد. در [۱۵] یک روش احراز هویت مداوم سبک‌وزن برای محیط اینترنت اشیا پیشنهاد گردیده است. از آنجا که حسگرها در سناریوهای خاص مانند نظارت بر سلامت شخصی و سیستم‌های کنترل صنعتی [۳۴]، می‌بایست داده‌های دریافتی را در مدت زمان کوتاه به دروازه‌ها منتقل کنند، یک مکانیسم احراز هویت مداوم سریع نیاز می‌باشد. در روش پیشنهادی [۱۵]، از یک رمز مشترک برای ساخت توکن احراز هویت استفاده می‌شود و توکن‌ها و پیام‌های مربوط در یک بازه زمانی برای احراز هویت، از کاربر به سرور منتقل می‌گردند. سرور وظیفه بررسی و تأیید پیام‌های دریافت‌شده از کاربر بر اساس توکن‌های مرتبط را دارد.

روش‌های سبک‌وزن دیگری برای اینترنت اشیا ارائه شده‌اند. مرجع [۳۵] امکان احراز هویت بین سنسورها و ایستگاه‌های پایه را فراهم می‌آورد تا اطلاعات مربوط به سلامت جمع‌آوری شوند. این طرح از nonces و HMAC استفاده می‌کند. مرجع [۳۶] یک روش سبک‌وزن با استفاده از nonces، عملیات XOR و پیام Keyed-Hash برای احراز هویت و مبادله داده و همچنین توافق بر روی کلید بین گره‌های سنسور و کاربران نهایی ارائه داده است. در [۳۷] یک روش احراز هویت متقابل سبک‌وزن و توافق کلید برای شبکه بی‌سیم بدن پیشنهاد شده که هزینه ارتباطی کمی دارد. در [۳۸] روشی با هزینه محاسباتی کمتر نسبت به [۳۷] ارائه شده است اما این روش، هزینه‌های ارتباطی زیادی دارد. در [۳۹] با استفاده از عملگرهای XOR و Hash یک احراز هویت متقابل ارائه گردیده است.

در جدول ۱، یک دسته‌بندی از روش‌های احراز هویت در اینترنت اشیا ارائه گردیده و معایب هر گروه بیان شده است.

احراز هویت استاتیک در مقابل حملات دزدی نشست دارای نقاط ضعفی است که موجب نیاز به روش احراز هویت مداوم در کنار آن می‌باشد. از سوی دیگر، تکرار زیاد احراز هویت استاتیک در تمامی تحقیقات پیشین موجب افزایش هزینه‌های محاسباتی، هزینه زمان اجرا و ارتباطات در سطح شبکه می‌شود که این امر موجب افزایش مصرف انرژی گره‌ها می‌گردد. یکی دیگر از مشکلات احراز هویت استاتیک، زمان اجرای ثابت و تکرار زیاد آن می‌باشد. به منظور حل مشکل ثابت‌بودن دوره زمانی که باعث افزایش هزینه‌های احراز هویت می‌شود، می‌توان از احراز هویت مداوم استفاده نمود و بازه بین هر احراز هویت استاتیک را به نوع و اهمیت ترافیک گره‌ها وابسته کرد. در روش پیشنهادی از یک پروتکل مکمل در کنار احراز هویت استاتیک و مداوم استفاده می‌کنیم که این امر باعث می‌شود برای گره‌های مختلف حسگر بر اساس اهمیت ترافیک، بازه زمانی کم یا زیاد گردد که همین موضوع موجب کاهش هزینه‌های مختلف در احراز هویت می‌شود.

۳- روش پیشنهادی

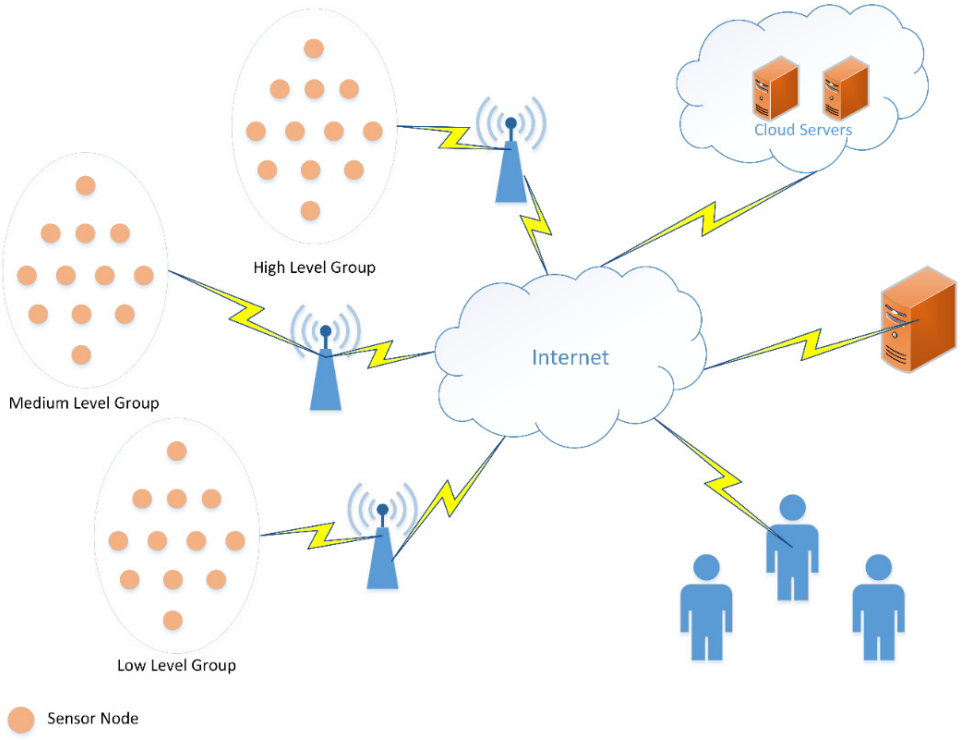
در این مقاله برای احراز هویت دستگاه به دستگاه، احراز هویت استاتیک و مداوم بر اساس اولویت‌بندی گره‌ها با استفاده از نرخ ترافیک ارائه شده است. ابتدا گره‌ها بر اساس [۴۰] اولویت‌بندی می‌شوند. برای این منظور، ابتدا اولویت ترافیک گره‌ها مشخص می‌گردد و سپس گره‌ها بر اساس اولویت، گروه‌بندی می‌شوند. برای هر اولویت، یک ضریب اهمیت در نظر گرفته می‌شود که با کمک آن، بازه زمانی مورد نیاز برای احراز هویت استاتیک هر گروه مشخص می‌گردد. در واقع گره‌ها بر اساس میزان اهمیت ترافیک‌هایشان، اولویت‌بندی و گروه‌بندی می‌شوند. تعداد اولویت در این مقاله ۳ در نظر گرفته شده است (شکل ۱). اعضای هر اواویت در ابتدای بازه زمانی به صورت استاتیک احراز هویت می‌شوند و یک توکن ایجاد می‌گردد و تا اتمام بازه زمانی، از این توکن ایجادشده جهت احراز هویت مداوم استفاده می‌گردد. همچنین روش احراز هویت سبک‌وزن بر اساس عملگرهای XOR و Hash برای احراز هویت استاتیک و مداوم ارائه شده است.

در این مقاله، مدل تهدید Dolev-Yao استفاده می‌شود که فرض می‌نماید گره‌های حسگر و سرورها در یک کانال ناامن، ارتباط برقرار می‌کنند. سرور به عنوان گره قابل اعتماد است، اما مهاجم می‌تواند به پایگاه داده سرور نفوذ کند و تمام داده‌های موجود در آن را به جز کلید اصلی سرور به دست آورد.

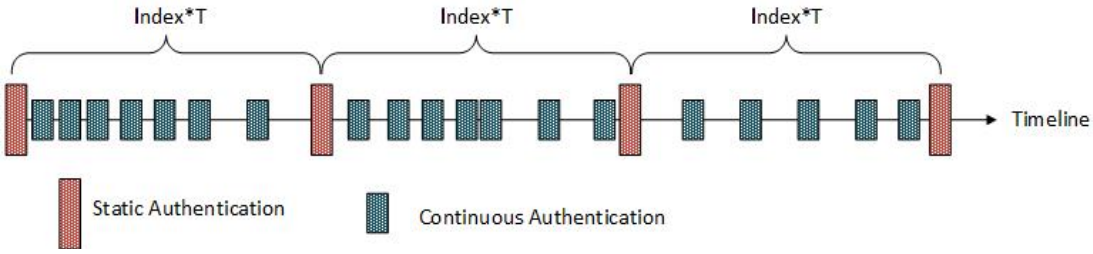
همچنین مهاجم، توانایی به دست آوردن تمام داده‌های مبادله‌شده در کانال ارتباطی را دارد و می‌تواند داده‌های جدیدی در کانال تزریق کند. در ضمن، داده‌های ارسالی موجود در کانال را می‌تواند با مقادیر جدید جایگزین کرده و از نو در کانال ارتباطی ارسال نماید. حسگرها به دلیل محدودیت هزینه مالی از نظر فیزیکی محافظت نمی‌شوند و در نتیجه مهاجم می‌تواند داده‌های ذخیره‌شده در حافظه گره حسگر را استخراج کند و از آنها برای انجام کارهای مخرب استفاده نماید.

۳-۱ اولویت‌بندی بر اساس نرخ ترافیک

در [۴۰] یک روش جهت اولویت‌بندی گره‌ها در یک شبکه بر اساس ترافیک پیشنهاد شده است. این روش مطابق جدول ۲ از چهار کلاس ترافیک بلادرنگ (کلاس RT)، کلاس ترافیک غیر بلادرنگ اولویت بالا (NRT۱)، کلاس ترافیک غیر بلادرنگ اولویت متوسط (کلاس NRT۲) و کلاس ترافیک غیر بلادرنگ اولویت پایین (کلاس NRT۳) پشتیبانی می‌کند و برای هر کلاس، ترافیک وزنی اختصاص داده شده است. اولویت ترافیک هر گره برابر با مجموع وزن کلاس ترافیک‌های هر گره می‌باشد.



شکل ۱: گروه‌بندی و اولویت‌بندی گره‌های حسگر.



شکل ۲: نحوه احراز هویت استاتیک و مداوم و بازه زمانی در روش پیشنهادی.

جدول ۲: کلاس‌بندی ترافیک.

نوع ترافیک	کلاس ترافیک	وزن (W)
بلادرنگ	ترافیک بلادرنگ (RT)	۱۰
غیر بلادرنگ	اولویت بالا، کلاس ترافیک غیر بلادرنگ (NRT۱)	۶
	اولویت متوسط، کلاس ترافیک غیر بلادرنگ (NRT۲)	۳
	اولویت پایین، کلاس ترافیک غیر بلادرنگ (NRT۳)	۱

جدول ۳: گروه‌بندی و اولویت‌بندی.

اولویت گروه	قانون اولویت‌بندی	ضریب اهمیت	بازه زمانی T_i
اولویت بالا	$m <$ اولویت حسگر[1]	α	$T_i = \alpha \times T$
اولویت متوسط	$k <$ اولویت حسگر $< m$	β	$T_i = \beta \times T$
اولویت پایین	اولویت حسگر $< k$	γ	$T_i = \gamma \times T$

1. Sensor Priority

پس از محاسبه اولویت گره‌ها، مطابق جدول ۳ گره‌های با اولویت بزرگ‌تر از m در گروه با اولویت بالا قرار می‌گیرند و در گروه با اولویت متوسط، گره‌هایی که اولویت آنها بین m و k می‌باشد و در نهایت گره‌های با اولویت کمتر از k در گروه اولویت پایین قرار دارند. سپس برای هر گروه، یک ضریب اهمیت $(Index)$ در نظر گرفته می‌شود که با کمک آن، بازه زمانی مورد نیاز برای احراز هویت استاتیک هر اولویت مشخص می‌گردد. بازه زمانی T_i هر اولویت به صورت $T_i = Index \times T$ محاسبه می‌شود و T بازه زمانی پایه است. مقادیر T، m، k، α، β و γ متناسب با اهمیت ترافیک و گروه‌بندی و با نظر یک فرد خبره مقداردهی می‌گردد.

۳-۲ احراز هویت استاتیک و مداوم

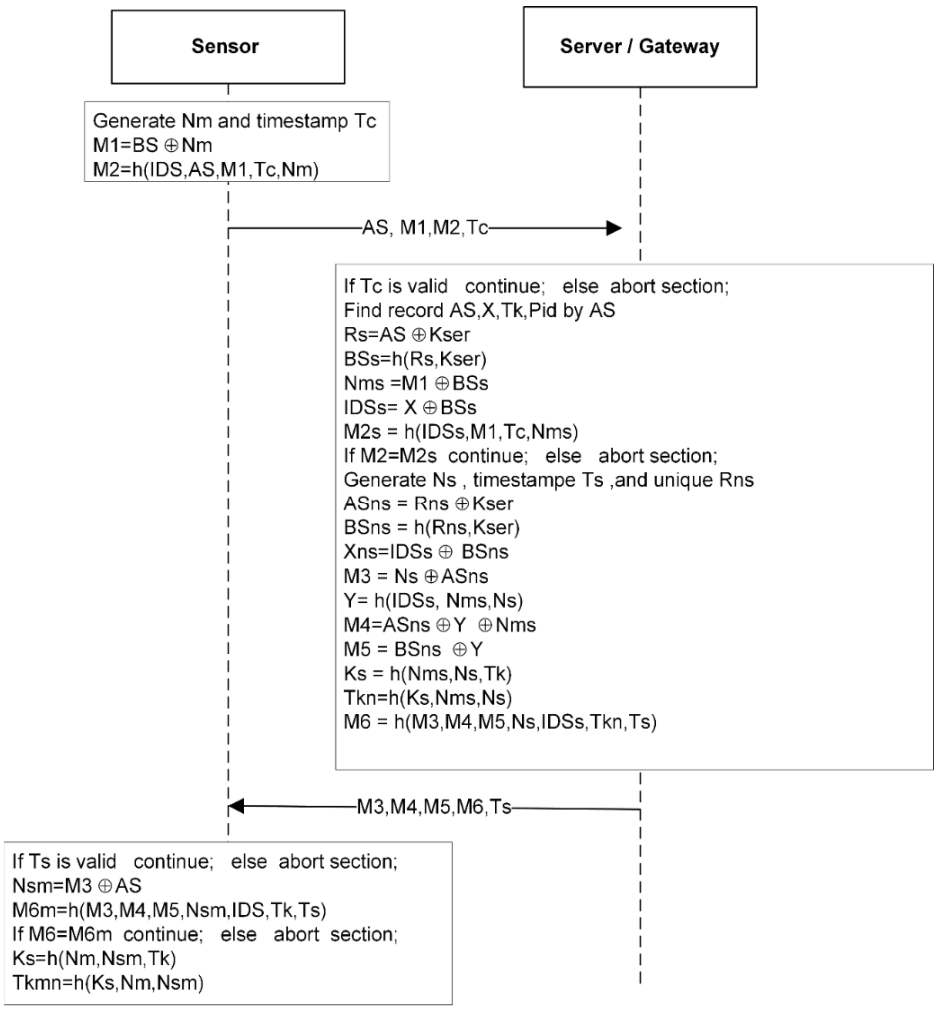
در احراز هویت استاتیک، در ابتدای ارتباط یک فرایند کامل انجام می‌شود و توکنی ایجاد می‌گردد. تا اتمام اعتبار توکن، احراز هویت مداوم می‌تواند به طور پیوسته گره را بررسی و تأیید کند. در روش پیشنهادی برای اعضای گروه i، احراز هویت استاتیک در ابتدای بازه زمانی T_i انجام می‌گیرد و توکنی ایجاد می‌گردد و تا اتمام بازه T_i، اعضای گروه با کمک آن توکن احراز هویت مداوم انجام می‌دهند. بنابراین طبق شکل ۲ برای گره‌های گروه‌ها، احراز هویت استاتیک و مداوم انجام می‌گیرد.

۳-۳ فازهای احراز هویت

روش احراز هویت پیشنهادی از ۴ فاز تشکیل شده است: فاز مقداردهی اولیه، فاز ثبت نام، فاز احراز هویت استاتیک و فاز احراز هویت مداوم. در جدول ۴ نمادها و پارامترهای مورد نیاز این بخش بیان گردیده و در ادامه فازهای مربوط به روش پیشنهادی توضیح داده شده است.

۳-۳-۱ فاز مقداردهی اولیه

این مرحله توسط مدیر سیستم انجام می‌گیرد.



شکل ۳: فاز احراز هویت استاتیک.

جدول ۴: نمادها و پارامترهای استفاده‌شده در روش پیشنهادی.

نماد	توضیحات
ID_{ser}	شناسه سرور
IDS	شناسه گره سنسور
Pid	اولویت گروه
Ti	بازه زمانی تعریف‌شده برای بازه زمانی احراز هویت استاتیک و مداوم
AS, BS, X	پارامترهای احراز هویت
Nm, Ns	پارامترهای موقتی محرمانه
$h(.)$	تابع یک‌طرفه Hash
Tc, Ts	مهر زمانی
Xor	عملگر XOR
,	الحاق دو رشته
Tk	توکن
Ks	کلید جلسه
$Kser$	کلید اصلی سرور

- مدیر سیستم، یک کلید اصلی (K_{ser}) برای سرور در نظر می‌گیرد.
- این کلید در حافظه سرور ذخیره می‌گردد.

۳-۳-۲ فاز ثبت نام گره سنسور توسط مدیر سیستم

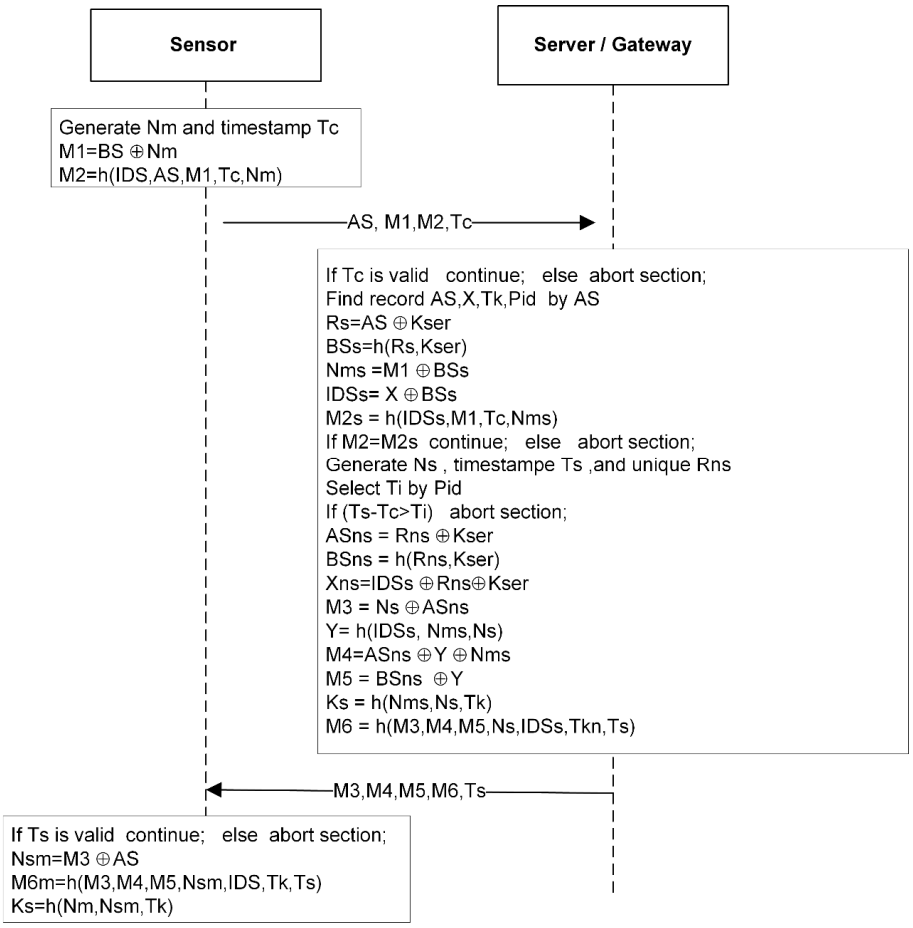
- توسط سرور برای هر گره یک شناسه یکتای IDS، مقدار یکتای r و مقدار تصادفی Tk در نظر گرفته می‌شود.
- توسط مدیر سیستم، مقادیر زیر محاسبه می‌گردد

$$AS = r \oplus Kser$$
$$BS = h(r, Kser)$$
$$X = IDS \oplus h(r, Kser)$$

- رکورد IDS, AS, BS, Tk در حافظه گره سنسور ذخیره می‌شود.
- رکورد AS, X, Tk در حافظه سرور ذخیره می‌گردد و همچنین مقدار شناسه گروه برای هر سنسور نیز ذخیره خواهد شد.

۳-۳-۳ فاز احراز هویت استاتیک

در مرحله احراز هویت استاتیک (شکل ۳)، گره و سرور به طور متقابل یکدیگر را احراز هویت می‌کنند و با هم بر روی توکن بعدی مذاکره می‌نمایند. این توکن برای احراز هویت مداوم در طول بازه زمانی T_i مورد استفاده قرار می‌گیرد و پس از اتمام دوره زمانی مجدد با استفاده از احراز هویت استاتیک تولید می‌شود. مقدار توکن جدید بر اساس مقدار توکن قبلی، مقادیر Nm (مقدار تصادفی تولیدشده توسط گره) و Ns (مقدار تصادفی تولیدشده توسط سرور) و تابع Hash به دست می‌آید. در ضمن مقادیر Nm و Ns به صورت Hashشده با مقادیر دیگر بین گره و سرور مبادله می‌شوند. همچنین از سه مقدار شناسه یکتای IDS، مقدار یکتای r و مقدار تصادفی Tk برای احراز هویت استفاده می‌شود. با استفاده از این مقادیر در فاز ثبت نام، مقادیری بر روی سرور و گره قرار گرفته‌اند. مهم‌ترین دلیل استفاده از ۲ مقدار r و مقدار تصادفی Tk، امکان انجام احراز هویت متقابل می‌باشد و در ادامه مراحل این فاز بیان شده‌اند.



شکل ۴: فاز احراز هویت مداوم.

مرحله ۱

- ابتدا گره سنسور، مقدار Nm و مهر زمانی Tc را ایجاد می‌کند و بعد از آن مقدار $M۱ = BS \oplus Nm$ را محاسبه کرده و سپس $M۲ = h(IDS, AS, M۱, Tc, Nm)$ را محاسبه می‌نماید.
- گره سنسور مقادیر AS، $M۱$، $M۲$ و Tc را به سرور ارسال می‌کند.

مرحله ۲

- سرور ابتدا شرط $t_{new} - Tc < \Delta t$ را بررسی می‌کند و در صورتی که شرط درست نباشد، جلسه کاری خاتمه می‌یابد.
- سرور وجود AS را در پایگاه داده‌اش بررسی می‌کند و در صورتی که وجود نداشته باشد، جلسه کاری خاتمه می‌یابد. در غیر این صورت مقادیر AS، X و Tk را بازیابی می‌کند.
- سرور $Rs = AS \oplus Kser$ و $BSs = h(Rs, Kser)$ و همچنین $Nms = M۱ \oplus BSs$، $IDSs = X \oplus BSs$ و $M۲s = h(IDSs, AS, M۱, Tc, Nms)$ را محاسبه می‌کند.
- اگر $M۲ = M۲s$ بود، عملیات ادامه پیدا می‌کند و در غیر این صورت عملیات شکست می‌خورد.
- سرور مقدار Ns، مهر زمانی Ts و مقدار یکتای Rns را ایجاد می‌کند.
- سرور $ASns = Rns \oplus Kser$، $BSns = h(Rns, Kser)$، $Xns = IDSs \oplus BSns$، $M۳ = Ns \oplus BSns$، $Y = h(IDSs, Nms, Ns)$، $M۴ = ASns \oplus Y \oplus Nms$، $M۵ = BSns \oplus Y$ و $Ks = h(Nms, Ns, Tk)$ را محاسبه می‌کند.
- سرور $Tkn = h(Ks, Nms, Ns)$ را محاسبه می‌کند.
- سرور $M۶ = h(M۳, M۴, M۵, Ns, IDSs, Tkn, Ts)$ را محاسبه می‌کند.
- سرور مقادیر $M۳, M۴, M۵, M۶, Ts$ را به گره سنسور ارسال می‌کند.

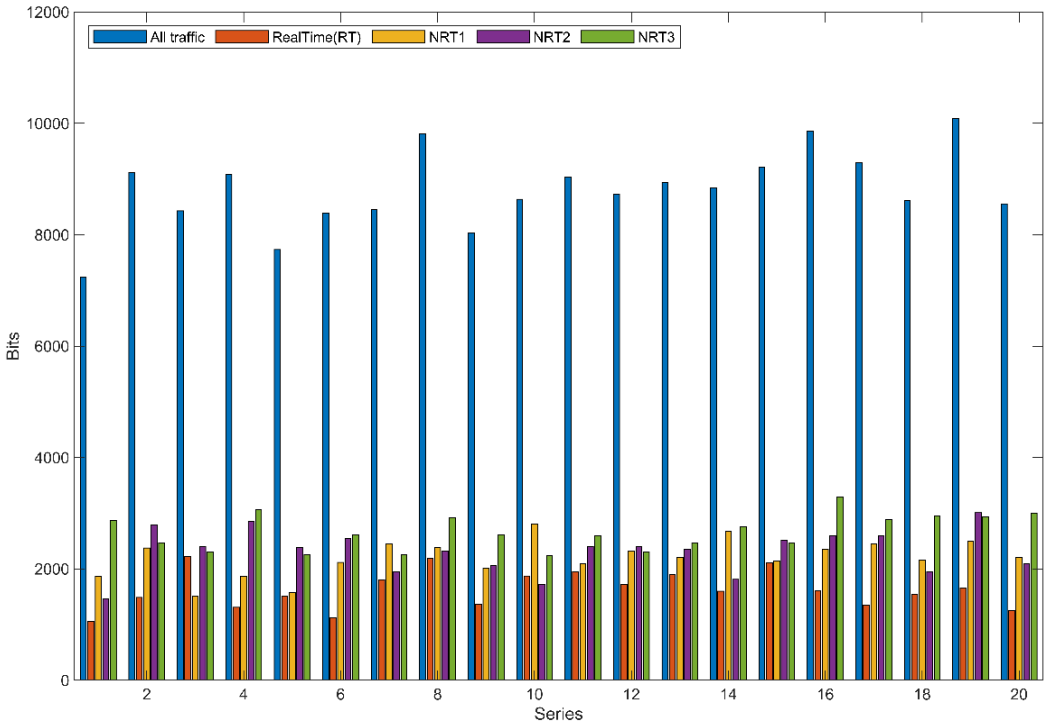
مرحله ۳

- گره سنسور ابتدا شرط $t_{new} - Tc < \Delta t$ را بررسی می‌کند و در صورتی که شرط درست نباشد، جلسه کاری خاتمه می‌یابد.
- سپس گره سنسور مقادیر $Nsm = M۳ \oplus AS$ و $M۶m = h(M۳, M۴, M۵, Nsm, IDS, Tk, Ts)$ را محاسبه می‌کند.
- اگر $M۶ = M۶m$ بود، عملیات ادامه پیدا می‌کند و در غیر این صورت عملیات شکست می‌خورد.
- گره سنسور مقادیر $Ks = h(Nm, Nsm, Tk)$ و $Tkmn = h(Ks, Nm, Nsm)$ را محاسبه می‌کند.

پس از احراز هویت استاتیک، گره سنسور برای بازه زمانی، احراز هویت مداوم انجام می‌دهد. در این بازه زمانی از توکن ایجادشده Tk استفاده خواهد شد.

۳-۳-٤ فاز احراز هویت مداوم

در مرحله احراز هویت مداوم (شکل ۴)، گره و سرور به طور متقابل یکدیگر را احراز هویت می‌کنند. باید توجه داشت که احراز هویت مداوم بین دو احراز هویت استاتیک و در بازه زمانی T_i انجام می‌شود. از آنجا که احراز هویت مداوم پس از احراز هویت استاتیک موفقیت‌آمیز اتفاق می‌افتد، توکن Tk و بازه زمانی T_i وجود دارد و در این بازه زمانی، احراز هویت



شکل ۵: ترافیک واقعی و کلاس‌بندی ترافیک در شبکه.

مداوم انجام می‌گیرد. در ادامه، مراحل این احراز هویت ارائه شده است.

مرحله ۱

- ابتدا گره سنسور مقدار Nm و مهر زمانی Tc را ایجاد می‌کند و بعد از آن مقدار $M1 = BS \oplus Nm$ را محاسبه کرده و سپس مقدار $M2 = h(IDS, AS, M1, Tc, Nm)$ را محاسبه می‌نماید.
- گره سنسور مقادیر AS، $M1$، $M2$ و Tc را به سرور ارسال می‌کند.

مرحله ۲

- سرور ابتدا شرط $t_{new} - Tc < \Delta t$ را بررسی می‌کند و در صورتی که شرط درست نباشد، جلسه کاری خاتمه می‌یابد.
- سرور وجود AS را در پایگاه داده‌اش بررسی می‌کند. در صورتی که وجود نداشته باشد، جلسه کاری خاتمه می‌یابد و در غیر این صورت مقادیر AS، X، Tk و Pid را بازیابی می‌کند.
- سرور $Rs = AS \oplus Kser$ و $BSs = h(Rs, Kser)$ و همچنین $Nms = M1 \oplus BSs$، $IDSs = X \oplus BSs$ و $M2s = h(IDSs, AS, M1, Tc, Nms)$ را محاسبه می‌کند.
- اگر $M2 = M2s$ بود، عملیات ادامه پیدا می‌کند و در غیر این صورت، عملیات شکست می‌خورد.
- سرور مقدار Ns، مهر زمانی Ts و مقدار یکتای Rns را ایجاد می‌کند.
- سپس سرور بر اساس مقدار Pid، مقدار T_i را به دست می‌آورد. T_i بازه زمانی مربوط به گروه سنسور را نشان می‌دهد.
- سرور شرط $Ts - Tc > Ti$ را بررسی می‌کند و اگر شرط برقرار باشد، جلسه کاری خاتمه می‌یابد.
- در ادامه سرور مقادیر $ASns = Rns \oplus Kser$، $BSns = h(Rns, Kser)$، $Xns = IDSs \oplus BSns$، $M3 = Ns \oplus BSns$، $Y = h(IDSs, Nms, Ns)$، $M4 = ASns \oplus Y \oplus Nms$ و $M5 = BSns \oplus Y$ را محاسبه می‌کند.
- سرور $Ks = h(Nms, Ns, Tk)$ را محاسبه می‌کند.
- سرور $M6 = h(M3, M4, M5, Ns, IDS, TKn, Ts)$ را محاسبه می‌کند.
- سرور مقادیر $M3, M4, M5, M6, Ts$ را به گره سنسور ارسال می‌کند.

مرحله ۳

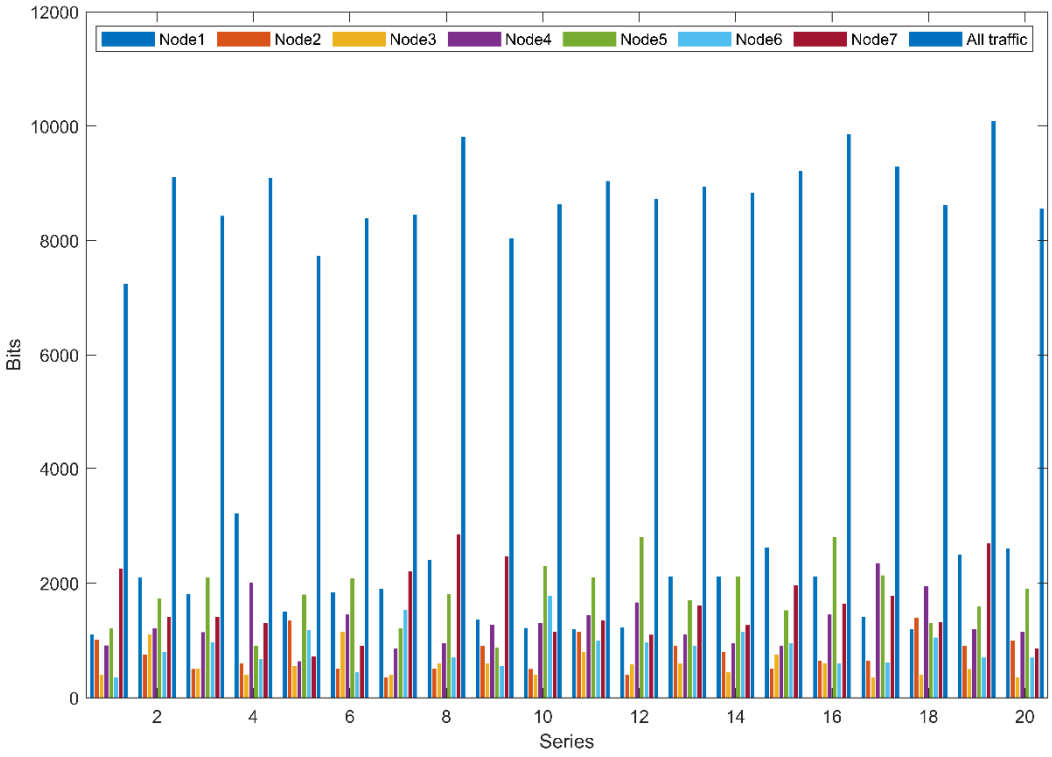
- گره سنسور ابتدا شرط $t_{new} - Tc < \Delta t$ را بررسی می‌کند و در صورتی که شرط درست نباشد، جلسه کاری خاتمه می‌یابد.
- توسط گره سنسور مقادیر $Nsm = M3 \oplus BS$ و $M6m = h(M3, M4, M5, Nsm, IDS, Tk, Ts)$ محاسبه می‌گردند.
- گره سنسور مقدار $Ks = h(Nm, Nsm, Tk)$ را محاسبه می‌نماید و کلید جلسه را به دست می‌آورد.

۴- شبیه‌سازی، تجزیه و تحلیل امنیتی

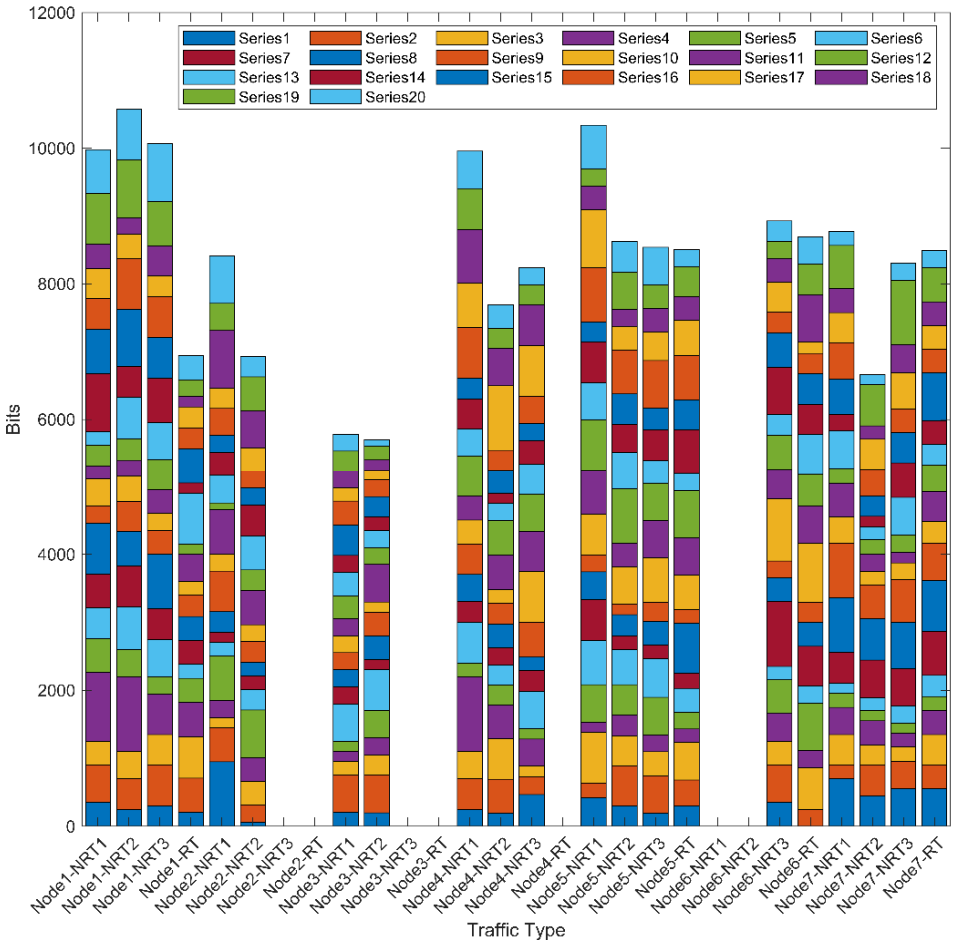
در این بخش، ابتدا اولویت‌بندی و گروه‌بندی گره‌ها شبیه‌سازی می‌شود و سپس پروتکل ارائه‌شده را از لحاظ امنیتی تحلیل می‌کنیم. برای این منظور، ما از ابزار AVISPA برای اثبات صحت امنیتی احراز هویت استفاده کرده‌ایم.

۴-۱ شبیه‌سازی اولویت‌بندی و گروه‌بندی گره‌ها

در این شبیه‌سازی ۷ گره سنسور در نظر گرفته شده و هدف، اولویت‌بندی و گروه‌بندی آنها بر اساس اولویت ترافیکشان می‌باشد. تمام گره‌ها می‌توانند هر چهار نوع کلاس ترافیک RT، NRT۱، NRT۲ و NTR۳ را داشته باشند که به ترتیب دارای وزن ۱۰، ۶، ۳ و ۱ هستند. در شکل ۵ ترافیک واقعی و کلاس‌بندی ترافیک در شبکه و در شکل ۶ ترافیک واقعی و به تفکیک گره ارائه شده است. بنابراین پروتکل پیشنهادی می‌تواند بر اساس اولویت به هر گره یک وزن اختصاص دهد.



شکل ۶: ترافیک واقعی و ترافیک به تفکیک گره.



شکل ۷: ترافیک هر گره بر اساس کلاس‌بندی ترافیک.

در شکل ۷ میزان ترافیک بر اساس کلاس‌بندی پیشنهادی در هر گره مشاهده می‌شود. بر اساس این ترافیک‌ها، وزن هر گره به منظور قرارگرفتن در گروه مورد نظر محاسبه می‌گردد. در ادامه مطابق جدول ۵، گروه‌بندی گره‌ها بر اساس وزن‌های گره‌ها محاسبه می‌گردد. تمامی گره‌ها دارای وزن کمتر از ۱۰ در گروه ۳، گره‌هایی دارای وزن بین ۱۰ و ۱۵ در گروه ۲ و در انتها گره‌های با وزن بالای ۱۵ در گروه ۱ قرار می‌گیرند.

جدول ۵: کلاس‌بندی ترافیک در هر گره با میزان اولویت.

Sensor	$RT(w=۱۰)$	$NRT۱(w=۶)$	$NRT۲(w=۳)$	$NRT۳(w=۱)$	Priority	Group ID
۱	Yes	Yes	Yes	Yes	۲۰	۱
۲	No	Yes	Yes	No	۹	۳
۳	No	Yes	Yes	No	۹	۳
۴	No	Yes	Yes	Yes	۱۰	۲
۵	Yes	Yes	Yes	Yes	۲۰	۱
۶	Yes	No	No	Yes	۱۱	۲
۷	Yes	Yes	Yes	Yes	۲۰	۱

بنابراین گره‌های ۲ و ۳ در گروه اولویت پایین یا گروه ۳ و گره‌های ۴ و ۶ در گروه اولویت متوسط یا گروه ۲ و گره‌های ۱، ۵ و ۷ در گروه اولویت بالا یا گروه ۱ گروه‌بندی می‌شوند.

پس از مشخص‌کردن گروه‌بندی گره‌ها، بر اساس جدول ۳ ضریب زمانی برای گروه اول، دوم و سوم به صورت $\alpha=۳$، $\beta=۲$ و $\gamma=۱$ تعیین می‌گردد. گروه اول به دلیل داشتن ترافیک بااهمیت و حساس‌بودن اطلاعات رد و بدل شده دارای ضریب ۳ است و بازه زمانی برای هر احراز هویت استاتیک برابر با ۳T می‌باشد. در حالی که برای سایر گره‌های موجود در گروه‌های دوم و سوم بازه زمانی ۲T و T در نظر گرفته شده است.

۴-۲ شبیه‌سازی با AVISPA

در این بخش، شبیه‌سازی با استفاده از AVISPA ارائه شده و نشان می‌دهد که پروتکل در مقابل حملات امنیتی مقاوم می‌باشد. توضیحات و اطلاعات بیشتر در این خصوص را در [۴۱] تا [۴۳] می‌توانید پیدا کنید. ابتدا به ارزیابی احراز هویت استاتیک می‌پردازیم. برای ارزیابی، جلسه‌های sensor، server، goal و environment در زبان HLPSL ارائه شده‌اند. در شکل ۸- الف کد HLPSL برای جلسه سنسور ارائه شده است. در جلسه sensor، ابتدا مقادیر AS، BS و Tk مقداردهی اولیه می‌شوند و سپس عدد تصادفی Nm و مهر زمانی Tc تولید شده و عملیات Hash و XOR روی مقادیر فاز مقداردهی اولیه و مقادیر تولیدشده انجام می‌گردد و با استفاده از $SND(AS,M۱,M۲,Tc)$، مقادیر AS، $M۱$، $M۲$ و Tc را به سرور ارسال می‌کند. نوع تعریف $channel(dy)$ نشان می‌دهد که کانال از مدل تهدید Dolev-Yao می‌باشد.

$secret(Nm',sec۱,\{M,S\})$ بیان می‌کند که مقدار Nm' فقط برای M و S شناخته‌شده است و همچنین تابع $witness(M,S,server\_node\_nm,Nm')$ مشخص می‌کند که M مقدار Nm' را به صورت جدید برای S تولید کرده است. در مرحله بعد، سنسور مقادیر $M۳,M۴,M۵,M۶,Tc$ را با استفاده از $RCV(M۳,M۴,M۵,M۶,Tc)$ دریافت می‌کند. پس از انجام یک سری محاسبات بر روی داده‌ها با استفاده از تابع $request(S,M,server\_node\_ns,Ks')$، سنسور سرور را احراز هویت می‌کند. در شکل ۸- ب کد HLPSL برای جلسه سرور ارائه شده است. در جلسه server، ابتدا مقادیر AS، X و Tk مقداردهی اولیه می‌شوند و سپس با استفاده از $RCV(AS,M۱,M۲,Tc)$، مقادیر AS، $M۱$، $M۲$ و Tc دریافت می‌گردد. پس از انجام یک سری عملیات، با استفاده از $request(S,M,server\_node\_nm,Nms')$، سرور سنسور را احراز هویت می‌کند. در ادامه مهر زمانی Ts و مقادیر تصادفی Ns و Rns ایجاد می‌شوند و سپس با استفاده از تابع $secret(Ks',sec۲,\{M,S\})$ بیان می‌کند که مقدار Ks' فقط برای M و S شناخته‌شده است. در ادامه با استفاده از $witness(M,S,server\_node\_ns,Ks')$ مشخص می‌کند که S مقدار Ks' را به صورت جدید برای M تولید کرده است.

در شکل ۸- ج نقش‌ها برای goal و HLPSL environment ارائه شده است که با استفاده از دستور $sececy\_of\ sec۱,sec۲$ دو هدف محرمانگی را مشخص می‌کند و با دستور $authentication\_on\ server\_node\_nm$ و دستور $authentication\_on\ server\_node\_ns$ دو احراز هویت را مشخص می‌نماید. پس از اجرای ابزار AVISPA، دو هدف محرمانگی و دو احراز هویت تأیید می‌شود.

در ادامه، نتایج شبیه‌سازی ابزار AVISPA برای $OFMC$ و $CL-AtSe$ در احراز هویت استاتیک ارائه شده است. نتایج نشان می‌دهد که طرح پیشنهادی تحت مدل‌های $OFMC$ (شکل ۹- الف) و $CL-AtSe$ (شکل ۹- ب) امن است.

در ادامه به ارزیابی احراز هویت مداوم می‌پردازیم. جلسه‌های sensor، server، goal و environment در زبان HLPSL ارائه شده‌اند. کد HLPSL برای جلسه سنسور در شکل ۱۰- الف ارائه شده است. در این جلسه، ابتدا مقادیر AS، BS و Tk مقداردهی اولیه می‌شوند و سپس عدد تصادفی Nm و مهر زمانی Tc تولید شده و عملیات Hash و XOR بر روی مقادیر فاز مقداردهی اولیه و مقادیر تولیدشده انجام می‌گردد و با استفاده از $SND(AM,M۱,M۲,Tc)$، مقادیر AS، $M۱$، $M۲$ و Tc را به سرور ارسال می‌کند.

در این شبیه‌سازی نیز کانال از مدل تهدید Dolev-Yao می‌باشد. $secret(Nm',sec۱,\{M,S\})$ بیان می‌کند که مقدار Nm' فقط برای M و S شناخته‌شده است. همچنین $witness(M,S,server\_node\_nm,Nm')$ مشخص می‌کند که M مقدار Nm' را به صورت جدید برای S تولید کرده است. در مرحله بعدی، سنسور با استفاده از $RCV(M۳,M۴,M۵,M۶,Tc)$ مقادیر $M۳,M۴,M۵,M۶,Tc$ را دریافت می‌کند و پس از انجام یک سری محاسبات بر روی داده‌ها و سنسور با استفاده از تابع $request(S,M,server\_node\_ns,Ks')$، سرور را احراز هویت می‌نماید.

در شکل ۱۰- ب کد HLPSL جلسه سرور ارائه شده است. در این جلسه ابتدا مقادیر AS، X و Tk مقداردهی اولیه می‌شوند و سپس مقادیر AS، $M۱$، $M۲$ و Tc با استفاده از $RCV(AS,M۱,M۲,Tc)$ دریافت می‌گردد. پس از انجام یک سری عملیات، سرور با استفاده از $request(S,M,server\_node\_nm,Nms')$، سنسور را احراز هویت می‌کند. در ادامه مهر زمانی Ts و مقادیر تصادفی Ns و Rns ایجاد

```
role server(M,S: agent,
  IdS,R,Tok,Kser:message,
  Hash:hash_func,
  SND, RCV: channel(dy))
played_by S
def=
 local State:nat,
 AS,BS,X,Tk,M1,M2,IDSs,M2s,M2S,
 Tc,Rs,BSs,Nms,Ns,Ts,Rns,ASns,BSns,
 Xns,M3,Y,M4,M5,Ks,Tkn,M6:message
 init State:= 1
  /\ AS:= xor(R,Kser)
  /\ X:= xor(IdS,Hash(R,Kser))
  /\ Tk:=Tok
 transition
1. State = 1 /\ RCV(AS,M1,M2,Tc)=|>
  State':=3
  /\ Rs' := xor(AS,kser)
  /\ BSs' := Hash(Rs',kser)
  /\ Nms' := xor(M1,BSs')
  /\ IDSs' := xor(X,BSs')
  /\ M2S' := M2
  /\ M2s' := Hash(IDSs',AS,M1,Tc,Nms')
2. State = 3 /\ M2S'=M2s' =|>
  State' := 5
  /\ Ns' := new()
  /\ Ts' := new()
  /\ Rns' := new()
  /\ ASns' := xor(Rns',kser)
  /\ BSns' := Hash(Rns',kser)
  /\ Xns' := xor(IDSs,Hash(Rns',kser))
  /\ M3' := xor(Ns',ASns')
  /\ Y' := Hash(IDSs.Nms.Ns')
  /\ M4' := xor(xor(ASns',Y'),Nms)
  /\ M5' := xor(BSns',Y)
  /\ Ks' := Hash(Nms,Ns,Tk)
  /\ Tkn' := Hash(Nms.Ns'.Tk)
  /\ M6' :=
Hash(M3.M4.M5.Ns'.IDSs.Tkn'.Ts')
  /\ secret(Ks',sec2,{M,S})
  /\ witness(M,S,server_node_ns,Ks')
  /\ request(S,M,server_node_ns,Ks')
  /\ SND(M3,M4,M5,M6,Ts)
end role
```

(الف)

```
role sensor(M,S: agent,
  IdS,R,Tok,Kser:message,
  Hash: hash_func,
  SND, RCV: channel(dy))
 played_by M
 def=
 local State:nat,
AS,BS,Tk,IDS,Tc,Nm, Nsm,M6m,Ks,
M1,M2,M3, M4, M5,M6,Ts,Tkmn,
Y, ASmn,BSmn,M6N:message
 init State:= 0
  /\ AS:= xor(R,Kser)
  /\ BS:= Hash(R.Kser)
  /\ Tk:=Tok
  /\ IDS:=IdS
 transition
1. State = 0 /\ RCV(start) =|>
  State' := 2
  /\ Nm' := new()
  /\ Tc' := new()
  /\ M1' := xor(BS,Nm')
  /\ M2' := Hash(IDS.AS.M1'.Tc'.Nm')
  /\ secret(Nm',sec1,{M,S})
  /\ SND(AS,M1',M2',Tc')
  /\witness(M,S,server_node_nm,Nm')
  /\request(S,M,server_node_nm,Nm')
2. State = 2/\RCV(M3,M4,M5,M6,Ts)=|>
  State' :=4
  /\ Nsm' := xor(M3,AS)
  /\ M6N' := M6
  /\ M6m':=
Hash(M3.M4.M5.Nsm'.IDS.Tk.Ts)
  3. State = 4 /\ M6m'=M6N' =|>
   State' :=6
   /\ Ks' := Hash(Nm.Nsm.Tk)
   /\ Tkmn' := Hash(Ks'.Nm.Nsm)
  end role
```

(ب)

```
role session (M,S: agent, IdS,R,Tok,
    Kser:message,Hash : hash_func)
def=
 local SA, SB, RA, RB: channel (dy)
 composition
sensor(M,S, IdS,R,Tok,Kser,Hash,SA,RA)
/\server(M,S, IdS,R,Tok,Kser,Hash,SB,RB)
end role

role environment()
def=
 local
 Snd, Rcv: channel(dy)
 const m,s: agent,
 idS,r,tk,kser:message,
 h : hash_func,
 server_node_nm,server_node_ns,
 sec1,sec2: protocol_id
 intruder_knowledge = {m,s}
 composition
 session(m,s,idS,r,tk,kser,h)
 /\session(s,m,idS,r,tk,kser,h)
end role

goal
 secrecy_of sec1 , sec2
 authentication_on server_node_nm
 authentication_on server_node_ns
end goal
environment()
```

(ج)

شکل ۸: کدهای احراز هویت استاتیک در HLPSL، (الف) نقش سرور، (ب) نقش سنسور و (ج) نقش‌ها برای جلسه goal و environment.

می‌شوند و سپس با استفاده از تابع $secret(Ks', sec۲, \{M,S\})$ بیان می‌گردد که مقدار Ks' فقط برای M و S شناخته‌شده است. در ادامه با استفاده از $witness(M,S,server\_node\_ns,Ks')$ مشخص می‌کند که S مقدار Ks' را به صورت جدید برای M تولید کرده است.

همچنین نقش‌های جلسه goal و environment (شکل ۱۰- ج) در زبان HLPSL برای احراز هویت مداوم ارائه شده که مثل احراز هویت استاتیک می‌باشد.

نتایج شبیه‌سازی احراز هویت مداوم برای $OFMC$ و $CL-AtSe$ نشان می‌دهند که طرح پیشنهادی تحت مدل‌های $OFMC$ (شکل ۱۱- الف) و $CL-AtSe$ (شکل ۱۱- ب) امن بوده و در برابر حملات فعال و غیر فعال از جمله پخش مجدد و حملات مرد میانی مقاوم است.

۵- آنالیز امنیتی

۵-۱ حمله استراق سمع

از آنجایی که مهاجم، تمام اطلاعات موجود در کانال مشترک را می‌تواند به دست آورد، بنابراین به راحتی به مقادیر AS، $M۱$، $M۲$، $M۳$، $M۴$، $M۵$، $M۶$، Tm و Ts مربوط به احراز هویت‌ها نیز می‌تواند دسترسی پیدا کند. از طرفی ما نیاز داریم که از مقادیر IDS، $Kser$ و کلید جلسه Ks محافظت کنیم. مهاجم در هر دو احراز هویت استاتیک و مداوم، کلید اصلی $Kser$ را از AS نمی‌تواند به دست آورد زیرا r تصادفی و تازه بوده و آن را نمی‌داند. در ضمن امکان به دست آوردن مقادیر Nm و Ns از $M۱$ یا $M۳$ برای مهاجم میسر نمی‌باشد، زیرا مهاجم BS را نمی‌داند و Nm و Ns تصادفی و تازه هستند. همچنین مهاجم نمی‌تواند اطلاعات مفیدی از مقادیر $M۴$ و $M۵$ به دست آورد، زیرا $Y = h(IDSs, Nms, Ns)$، $M۴ = ASns \oplus Y \oplus Nms$ و $M۵ = BSns \oplus Y$ بوده و مهاجم Nm، Ns و IDS را نمی‌داند. از طرفی از آنجایی که مقادیر $M۲$ و $M۶$ به صورت رمزشده بین گره‌ها جابه‌جا می‌شوند، پس مهاجم نمی‌تواند به آنها نیز دسترسی پیدا کند و بنابراین طرح ما از حمله استراق سمع جلوگیری می‌کند.

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
 /home/span/span/testsuite/results/ContinuousNode.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.02s
  searchTime: 0.07s
  visitedNodes: 92 nodes
  depth: 10 plies
```

(الف)

```
SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL

PROTOCOL
  /home/span/span/testsuite/results/ContinuousNode.if

GOAL
  As Specified

BACKEND
  CL-AtSe

STATISTICS

  Analysed   : 34 states
  Reachable  : 32 states
  Translation: 0.11 seconds
  Computation: 0.01 seconds
```

(ب)

شکل ۹: نتایج احراز هویت استاتیک، (الف) نتیجه $OFMC$ و (ب) نتیجه $CL-AtSe$.

```
role server(M,S: agent,
    IdS,R,Tok,Kser:message,
    Hash:hash_func,
    SND, RCV: channel(dy))
 played_by S
 def=
  local State:nat,
  AS,BS,X,Tk,M1,M2,IDSs,M2s,M2S,
  Tc,Rs,BSs,Nms,Ns,Ts,Rns,ASns,BSns,
  Xns,M3,Y,M4,M5,Ks,Tkn,M6:message
  init State:= 1
   /\  AS:= xor(R,Kser)
   /\ X:= xor(IdS,Hash(R,Kser))
   /\ Tk:=Tok
 transition
 1. State = 1 /\ RCV(AS,M1,M2,Tc)=|>
   State':=3
   /\ Rs' := xor(AS,kser)
   /\ BSs' := Hash(Rs',kser)
   /\ Nms' := xor(M1,BSs')
   /\ IDSs' :=  xor(X,BSs')
   /\ M2S' := M2
   /\ M2s' := Hash(IDSs',AS,M1,Tc,Nms')
 2. State = 3   /\ M2S'=M2s' =|>
   State' := 5
   /\ Ns' := new()
   /\ Ts' := new()
   /\ Rns' := new()
   /\ ASns' := xor(Rns',kser)
   /\ BSns' := Hash(Rns',kser)
   /\ Xns' := xor(IDSs,Hash(Rns',kser))
   /\ M3' := xor(Ns',ASns')
   /\ Y' := Hash(IDSs.Nms.Ns')
   /\ M4' := xor(xor(ASns',Y'),Nms)
   /\ M5' := xor(BSns',Y)
   /\ Ks' := Hash(Nms,Ns,Tk)
   /\ M6' :=
Hash(M3.M4.M5.Ns'.IDSs.Tkn'.Ts')
   /\ secret(Ks',sec2,{M,S})
   /\ witness(M,S,server_node_ns,Ks')
   /\ request(S,M,server_node_ns,Ks')
   /\ SND(M3,M4,M5,M6,Ts)
 end role
```

(الف)

```
role sensor(M,S: agent,
    IdS,R,Tok,Kser:message,
    Hash: hash_func,
    SND, RCV: channel(dy))
  played_by M
  def=
  local State:nat,
 AS,BS,Tk,IDS,Tc,Nm, Nsm,M6m,Ks,
M1,M2,M3, M4, M5,M6,Ts,Tkmn,
 Y, ASmn,BSmn,M6N:message
  init State:= 0
   /\  AS:= xor(R,Kser)
   /\  BS:= Hash(R.Kser)
   /\ Tk:=Tok
   /\ IDS:=IdS
  transition
1. State = 0 /\ RCV(start)  =|>
   State' := 2
   /\ Nm' := new()
   /\ Tc' := new()
   /\ M1' := xor(BS,Nm')
   /\ M2' := Hash(IDS.AS.M1'.Tc'.Nm')
   /\ secret(Nm',sec1,{M,S})
   /\ SND(AS,M1',M2',Tc')
   /\witness(M,S,server_node_nm,Nm')
   /\request(S,M,server_node_nm,Nm')
2. State = 2/\RCV(M3,M4,M5,M6,Ts)=|>
   State' :=4
   /\ Nsm' := xor(M3,AS)
   /\ M6m':=
Hash(M3.M4.M5.Nsm'.IDS.Tk.Ts)
    /\ Ks' := Hash(Nm.Nsm.Tk)
   end role
```

(ب)

```
role session (M,S: agent, IdS,R,Tok,
     Kser:message,Hash : hash_func)
 def=
  local SA, SB, RA, RB: channel (dy)
  composition
 sensor(M,S, IdS,R,Tok,Kser,Hash,SA,RA)
 /\server(M,S, IdS,R,Tok,Kser,Hash,SB,RB)
 end role

 role environment()
 def=
  local
  Snd, Rcv: channel(dy)
  const m,s: agent,
  idS,r,tk,kser:message,
  h : hash_func,
  server_node_nm,server_node_ns,
  sec1,sec2: protocol_id
  intruder_knowledge = {m,s}
 composition
  session(m,s,idS,r,tk,kser,h)
  /\session(s,m,idS,r,tk,kser,h)
 end role

 goal
  secrecy_of sec1 , sec2
  authentication_on server_node_nm
  authentication_on server_node_ns
 end goal
environment()
```

(ج)

شکل ۱۰: کد احراز هویت مداوم، (الف) نقش سرور، (ب) نقش سنسور و (ج) نقش‌ها برای جلسه goal و environment.

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
 /home/span/span/testsuite/results/ContinuousNode.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.02s
  searchTime: 0.07s
  visitedNodes: 92 nodes
  depth: 10 plies
```

(الف)

```
SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL

PROTOCOL
  /home/span/span/testsuite/results/ContinuousNode.if

GOAL
  As Specified

BACKEND
  CL-AtSe

STATISTICS

  Analysed   : 34 states
  Reachable  : 32 states
  Translation: 0.11 seconds
  Computation: 0.01 seconds
```

(ب)

شکل ۱۱: نتایج احراز هویت مداوم، (الف) نتیجه $OFMC$ و (ب) نتیجه $CL-AtSe$.

۵-۲ گمنامی و غیر قابل ردیابی بودن گره

در طرح ما، هرگز شناسه گره (IDS) به طور مستقیم در کانال جابه‌جا نمی‌شود و همچنین مهاجم از طریق حمله استراق سمع، IDS را نمی‌تواند به دست آورد. از طرفی مقادیر r، Nm، Ns، Tm و Ts در احراز هویت استاتیک و احراز هویت مداوم کاملاً تصادفی هستند. پس مقادیر AS، $M۱$، $M۲$، $M۳$، $M۴$، $M۵$ و $M۶$ در هر مرحله احراز هویت متفاوت می‌باشند. بنابراین طرح ما، گمنام‌بودن گره و عدم ردیابی آن را تضمین می‌کند.

۵-۳ حمله تلاش مجدد

در هر دو احراز هویت استاتیک و مداوم، گره یک مهر زمانی Tm را تولید می‌کند و سپس $M۲=h(IDS,AM,M۱,Tm,Nm)$ را محاسبه می‌نماید. همچنین مهاجم نمی‌تواند IDS و Nm را به دست آورد، پس تولید $M۲$ معتبر جدید غیر ممکن است و بنابراین مهر زمانی Tm قابل تغییر نیست. سرور پس از دریافت Tm، اعتبار زمانی Tm را بررسی می‌کند. از طرفی سرور نیز یک مهر زمانی Ts ایجاد می‌نماید و $M۶=h(M۳,M۴,M۵,Nsm,IDS,Tk,Ts)$ را محاسبه می‌کند. مهاجم نمی‌تواند IDS، Nm و Tk را به دست آورد. گره نیز پس از دریافت Ts، اعتبار زمانی Ts را بررسی می‌کند و بنابراین ایجاد یک $M۶$ معتبر جدید غیر ممکن است که این موضوع تضمین می‌کند زمان Ts قابل تغییر نیست.

۵-۴ رازداری رو به جلو و رو به عقب

در طرح ما، کلید جلسه با استفاده از تابع Hash و به صورت $Ks=h(Nms,Ns,Tk)$ به دست می‌آید که Nms و Ns مقدار تصادفی تولیدشده در احراز هویت‌های استاتیک و مداوم می‌باشند. همچنین $Tkn=h(Ks,Nms,Ns)$ است و در ضمن مقادیر Ks، Nms و Ns در هر احراز هویت ایجاد می‌شوند. این وابستگی به مقادیر لحظه‌ای باعث می‌شود که اگر به هر نحوی کلید جلسه توسط مهاجم کشف شود، او نتواند کلیدهای جلسه قبلی یا بعدی را حدس بزند. پس با توجه به این که مقادیر Nms و Ns در هر احراز هویت استاتیک به صورت تصادفی و جدید هستند، بنابراین Ks در هر احراز هویت متفاوت خواهد بود. حتی اگر مهاجم تمام داده‌های قبلی کانال را به دست آورد و کلید اصلی $Kser$ را کشف کند، در این صورت او می‌تواند Nms و Ns همان احراز هویت را به دست آورد و نمی‌تواند کلیدهای جلسه قبلی را به دست آورد، زیرا Tk هرگز در کانال منتقل نمی‌شود. فقط یک راه برای به دست آوردن Tk وجود دارد و آن از طریق مشاهده و ضبط اطلاعات گره می‌تواند باشد. با وجود این در صورت مشاهده و ضبط اطلاعات گره، مهاجم فقط می‌تواند آخرین Tk را به دست آورد و تمام Tkها را نمی‌تواند به دست آورد، زیرا پارامتر دور قبلی Tk توسط تابع رمز شده‌اند. بنابراین طرح ما دارای رازداری کامل است.

۵-۵ حملات قطع ارتباط

در این نوع از حمله، مهاجم در هر مرحله از احراز هویت، ارتباط را مسدود می‌کند و این موضوع باعث می‌شود که طرفین ارتباط نتوانند با همدیگر تبادل اطلاعات انجام دهند. اگر مهاجم ارتباط را در مرحله ۱ مسدود کند، گره فقط باید احراز هویت جدیدی را مجدداً شروع کند. اگر مهاجم ارتباطات را در مرحله ۲ مسدود کند، تاپل جدید و قدیم AS,X,Tk در حافظه سرور وجود دارد. هنگامی که گره مجدداً احراز هویت جدیدی را شروع می‌کند، سرور می‌تواند از تاپل بدون به روز رسانی AS,X,Tk با موفقیت استفاده کند.

۵-۶ حفظ حریم خصوصی

در این طرح، شناسه گره (IDS) به طور مستقیم در کانال جابه‌جا نمی‌شود و ضمناً مهاجم از طریق حمله استراق سمع، IDS را نمی‌تواند به دست آورد. مقادیر r، Nm، Ns، Tm و Ts در احراز هویت استاتیک و احراز هویت مداوم کاملاً تصادفی انتخاب می‌شوند. بنابراین مقادیر AS، $M۱$، $M۲$، $M۳$، $M۴$، $M۵$ و $M۶$ در هر مرحله احراز هویت، متفاوت می‌باشند و امکان به دست آوردن IDS از طریق این اطلاعات وجود ندارد. این امر باعث می‌شود تا حریم خصوصی از نوع گمنامی داشته باشیم.

۶- ارزیابی و مقایسه عملکرد

۶-۱ هزینه حافظه مصرفی

در روش پیشنهادی، در هر گره S اطلاعات شامل IDS، AS، BS، Tk و کلید جلسه Ks برای احراز هویت ذخیره می‌شود. سرور نیز

جدول ۶: هزینه ذخیره‌سازی طرح ما.

دستگاه	فضای ذخیره‌شده بر حسب بیت
گره سنسور	۱۲۸۰
سرور	$۷۷۶i + ۲۵۶$

جدول ۷: هزینه ارتباطات در طرح ما.

ارتباطات بین گره‌ها	هزینه ارتباطات
سرور ← سنسور	۸۳۲ بیت
سنسور ← سرور	۱۰۸۸ بیت
کل	۱۹۲۰ بیت

جدول ۸: هزینه محاسباتی طرح ما.

گره	هزینه محاسباتی احراز هویت استاتیک	هزینه محاسباتی احراز هویت مداوم
سنسور	$۵T_{Hash} + ۵T_{Xor}$	$۴T_{Hash} + ۳T_{Xor}$
سرور	$۷T_{Hash} + ۹T_{Xor}$	$۶T_{Hash} + ۱۰T_{Xor}$

به ازای هر گره، اطلاعات AS_i ، X_i ، Tk_i و PID را ذخیره می‌نماید. همچنین سرور کلید سرور $Kser$ را باید ذخیره کند. در روش پیشنهادی از الگوریتم درهم‌ریزی SHA−۲۵۶ استفاده شده و این الگوریتم دارای خروجی ۲۵۶بیتی می‌باشد.

طول IDS, AS, BS, Tk و X و کلید جلسه Ks و کلید سرور $Kser$ با هم مساوی بوده و ۲۵۶ بیت می‌باشد. همچنین برای ذخیره PID نیاز به ۸ بیت حافظه داریم. بنابراین هر گره S، ۱۲۸۰ بیت اطلاعات را در حافظه خود ذخیره می‌کند و هر سرور نیاز به ذخیره $۷۷۶i + ۲۵۶$ بیت دارد که i تعداد گره ثبت‌شده است. جدول ۶ هزینه ذخیره‌سازی طرح ما را نشان می‌دهد.

۶-۲ هزینه ارتباطات

در مرحله اول، گره S مقادیر $AS, M۱, M۲, Tc$ را به گره سرور ارسال می‌کند و ضمناً مهر زمانی را ۶۴ بیت در نظر می‌گیریم. بنابراین هزینه ارتباطات گره سنسور به سرور $۸۳۲ = ۲۵۶ + ۲۵۶ + ۲۵۶ + ۶۴$ بیت خواهد بود. در مرحله دوم سرور مقادیر $M۳, M۴, M۵, M۶, Ts$ را به گره S ارسال می‌کند که برابر ۱۰۸۸ بیت می‌باشد. جدول ۷ هزینه ارتباطات این طرح را نشان می‌دهد.

۶-۳ هزینه محاسباتی

در این روش، احراز هویت‌ها با استفاده از تابع Hash و عملگر XOR بیان شده‌اند و از T_{Hash} برای نشان‌دادن زمان تابع Hash و از T_{Xor} جهت نشان‌دادن زمان تابع XOR استفاده می‌کنیم. در احراز هویت استاتیک، گره سنسور، ۵ عملیات تابع Hash و ۵ عملیات XOR را انجام می‌دهد و سرور، ۷ عملیات Hash و ۹ عملیات XOR را انجام می‌دهد. بنابراین هزینه محاسبه احراز هویت استاتیک گره سنسور برابر $۵T_{Hash} + ۵T_{Xor}$ می‌شود و هزینه محاسبه گره سرور $۷T_{Hash} + ۹T_{Xor}$ است. برای احراز هویت مداوم، گره سنسور، ۴ عملیات Hash و ۳ عملیات XOR را انجام می‌دهد و سرور، ۶ عملیات Hash و ۱۰ عملیات XOR را انجام می‌دهد. بنابراین هزینه محاسبه احراز هویت مداوم برای گره سنسور برابر $۴T_{Hash} + ۳T_{Xor}$ می‌شود و هزینه محاسبه گره سرور برابر $۶T_{Hash} + ۱۰T_{Xor}$ است. جدول ۸ هزینه محاسبه طرح ما را نشان می‌دهد.

جدول ۹: مقایسه ویژگی‌های امنیتی و عملکرد.

ویژگی‌های امنیتی	[۳۷]	[۲۵]	[۲۴]	[۳۹]	[۸]	طرح ما
رازداری رو به جلو[1]	✗	✗	✗	✓	✓	✓
حمله عدم همگامی[2]	✗	✓	✗	✓	✗	✓
حمله جعل هویت	✓	✓	✓	✓	✓	✓
احراز هویت متقابل	✗	✓	✓	✓	✓	✓
گمنامی گره	✓	✓	✗	✓	✗	✓
مقیاس‌پذیری	✗	✗	✗	✗	✗	✓

1. Perfect Forward Secrecy
2. Desynchronization Attack

جدول ۱۰: نمادها و توضیحات آنها.

توضیحات	نماد	زمان اجرا بر حسب میلی‌ثانیه
زمان اجرای تابع Hash	T_{Hash}	۰٫۰۰۵۲
زمان اجرای ضرب نقطه‌ای ECC	T_{ECC}	۱٫۲۸۲۸
زمان اجرای عملگر HMAC	T_{HMAC}	۰٫۰۱۵۶
زمان اجرای عملگر استخراج‌کننده فازی	T_{Fuzzy}	۱٫۳۵۲۴

۶-۴ مقایسه با روش‌های دیگر

از آنجایی که به غیر از پروتکل [۸]، پروتکل احراز هویت مداوم سبک‌وزنی وجود ندارد، بنابراین علاوه بر [۸]، طرحمان را با پروتکل‌های سبک‌وزن [۲۴]، [۲۵]، [۳۷] و [۳۹] نیز مقایسه کرده‌ایم و نتایج در جدول ۹ ارائه شده است. نتایج نشان می‌دهند که طرح ما از حملات بیشتر جلوگیری می‌کند و ویژگی‌های امنیتی بهتری نسبت به پروتکل‌های مرتبط دارد.

برای محاسبه هزینه احراز هویت، از آنجایی که زمان مصرف عملیات اتصال و XOR در مقایسه با عملیات دیگر ناچیز می‌باشد، از آنها صرف نظر می‌کنیم و نمادهای جدول ۱۰ را به منظور بیان زمان محاسباتی پروتکل‌های مختلف بیان می‌نماییم. زمان اجرا در این جدول، با در نظر گرفتن محیط و پلتفرم تست به شرح پردازنده Intel (R) Core TM i7-4710HQ 2.50GHz، حافظه ۸ گیگابایت و ویندوز ۸ نسخه ۶۴بیتی عمل تابع SHA−۲۵۶ بوده‌اند.

در جدول ۱۱ مقایسه هزینه اجرا و هزینه ارتباطات طرح ما با کارهای مرتبط [۸]، [۲۴]، [۲۵]، [۳۷] و [۳۹] ارائه شده است. پروتکل پیشنهادی ما دارای هزینه اجرا کمتری در مقایسه با سایر پروتکل‌های مرتبط است.

طرح ما هزینه اجرای پایین‌تری در مقایسه با طرح‌های مشابه دارد و نسبت به طرح‌های مشابه، ریسک امنیتی کمتری داشته و از ویژگی‌های امنیتی مناسبی برخوردار می‌باشد.

۷- نتیجه‌گیری

اولین خط دفاعی در اینترنت اشیا، احراز هویت می‌باشد. از طرفی در بیشتر احراز هویت‌های موجود، محدودیت منابع دستگاه‌های اینترنت اشیا در نظر گرفته نشده و بنابراین ما یک پروتکل احراز هویت سبک و امن برای اینترنت اشیا ارائه دادیم که بر اساس اولویت‌بندی گره‌ها می‌باشد و در آن از احراز هویت‌های استاتیک و مداوم استفاده می‌شود. با استفاده از این پروتکل، رازداری رو به جلو را بدون استفاده از رمزگذاری نامتقارن می‌توان تضمین کرد. در طرح پیشنهادی از عملگرهای محاسباتی سبک Hash و XOR استفاده کردیم. این امر به طور قابل توجهی هزینه محاسباتی را کاهش می‌دهد تا بتوان به راحتی از این طرح در گره‌های با

جدول ۱۱: مقایسه هزینه محاسبات و هزینه ارتباطات طرح ما با کارهای مرتبط.

طرح	کل هزینه اجرا	هزینه ارتباطات
مرجع [۳۷]	$۱۳T_{Hash}$ = ۰/۰۶۷۶ میلی‌ثانیه	۳۵۸۴ بیت
مرجع [۲۵]	$۳T_{ECC}+۱۶T_{Hash}$ = ۳/۹۳۱۶ میلی‌ثانیه	۱۹۲۰ بیت
مرجع [۲۴]	$۲۶T_{Hash}+۱T_{Fuzzy}$ = ۱/۴۵۷۶ میلی‌ثانیه	۱۴۰۸ بیت
مرجع [۳۹]	$۱۲T_{Hash}$ = ۰/۰۶۲۴ میلی‌ثانیه	۳۹۰۴ بیت
احراز هویت استاتیک [۸]	$۱۶T_{Hash}+۳T_{HMAC}$ = ۰/۱۳ میلی‌ثانیه	۲۳۰۴ بیت
احراز هویت مداوم [۸]	$۸T_{Hash}+۲T_{HMAC}$ = ۰/۰۷۲۸ میلی‌ثانیه	۱۵۳۶ بیت
احراز هویت استاتیک طرح ما	$۱۲T_{Hash}$ = ۰/۰۶۲۴ میلی‌ثانیه	۱۹۰۲ بیت
احراز هویت مداوم طرح ما	$۱۰T_{Hash}$ = ۰/۰۵۲ میلی‌ثانیه	۱۹۰۲ بیت

منابع محدود استفاده نمود. طرح پیشنهادی، حریم خصوصی را از طریق گمنامی گره‌ها فراهم می‌آورد و همچنین یک احراز هویت متقابل بین دو دستگاه ایجاد می‌کند که دو طرف همدیگر را اعتبارسنجی کرده و بر روی کلید جلسه به توافق برسند. در این مقاله از ابزار AVISPA برای تأیید امنیتی طرح و آنالیز امنیتی استفاده کردیم. این طرح در مقایسه با طرح‌های سبک‌وزن، مشکلات و ریسک‌های امنیتی پایینی دارد. در ضمن در روش ما، هزینه زمانی احراز هویت نسبت به روش‌های بررسی‌شده ۱۵٪ کاهش یافته و هزینه ارتباطات ۱۹۰۲ بیت می‌باشد.

از محدودیت‌های روش ما عدم امکان جابه‌جایی گره‌ها بین گروه‌ها است. در حالی که ترافیک و اولویت گره‌ها می‌توانند تغییر کند، بنابراین امکان جابه‌جایی بین گروه‌ها برای اعضای آنها می‌تواند نیاز باشد. پس امکان جابه‌جایی اعضای گروه‌ها می‌تواند جزء کارهای پژوهشی باشد. در ضمن، ارائه یک الگوریتم هوشمند و بهینه برای اولویت‌بندی گره‌ها در گروه‌های اولویتی می‌تواند انجام شود.

مراجع

[1] D. G. O. Rourke, Internet of Things (IoT) Cybersecurity Colloquium Internet of Things Cybersecurity Colloquium, 2017.

[2] M. Ammar, G. Russello, and B. Crispo, "Internet of Things: a survey on the security of IoT frameworks," *J. Inf. Secur. Appl.*, vol. 38, no. 1, pp. 8-27, Feb. 2018.

[3] J. Li, Y. Qu, F. Chao, H. P. H. Shum, E. S. L. Ho, and L. Yang, "Machine learning algorithms for network intrusion detection," In L. F. Sikos (Ed.), *AI in Cybersecurity*, pp. 151-179, Vol. 151, Springer, 1989.

[4] R. Mahmoud, T. Yousuf, F. Aloul, and I. Zualkernan, "Internet of Things (IoT) security : current status, challenges and prospective Measures," in *Proc. 10th In. Conf. for Internet Technology and Secured Transactions, ICITST'15*, pp. 336-341, London, UK, 14-16 2015.

[5] M. F. Aziz, A. N. Khan, J. Shuja, I. A. Khan, F. G. Khan, and A. ur R. Khan, "A lightweight and compromise-resilient authentication scheme for IoTs," *Trans. on Emerging Telecommunications Technologies*, vol. 33, no. 3, pp. 1-17, Nov. 2019.

[6] M. Abomhara and G. M. Køien, "Security and privacy in the Internet of Things: current status and open issues," in *Proc. Int. Conf. Priv. Secur. Mob. Syst.*, 8 pp., Aalborg, Denmark, 11-14 May 2014.

[7] I. Alqassem and D. Svetinovic, "A taxonomy of security and privacy requirements for the Internet of Things (IoT)," in *Proc. IEEE Int. Conf. Ind. Eng. Eng. Manag.*, pp. 1244-1248, Bandar Sunway, Malaysia, 9-12 Dec. 2014.

[8] Y. H. Chuang, N. W. Lo, C. Y. Yang, and S. W. Tang, "A lightweight continuous authentication protocol for the Internet of Things," *Sensors,* vol. 18, no. 4, pp. 1-26, Apr. 2018.

[9] I. Traore, *et al.*, "Dynamic sample size detection in learning command line sequence for continuous authentication," *IEEE Trans. Syst. Man, Cybern. Part B*vol. 42, no. 5, pp. 1343-1356, Oct. 2012.

[10] S. Mondal and P. Bours, "Continuous authentication in a real world settings," in *Proc. 8th Int. Conf. on Advances in Pattern Recognition, ICAPR'15*, 6 pp., Kolkata, India, 4-7 Jan. 2015.

[11] A. B. Buduru and S. S. Yau, "An effective approach to continuous user authentication for touch screen smart devices," in *Proc. IEEE Int. Conf. on Software Quality, Reliability and Security, QRS'15,* pp. 219-226, Vancouver, Canada, 3-5Aug. 2015.

[12] S. Mondal and P. Bours, "Continuous authentication and identification for mobile devices: combining security and forensics," in *Proc. IEEE Int.Workshop on Information Forensics and Security, WIFS'15*, 6 pp., Rome, Italy, 16-19 Nov. 2015.

[13] M. L. Brocardo, I. Traore, and I. Woungang, "Toward a framework for continuous authentication using stylometry," in *Proc. IEEE 28th Int. Conf. on Advanced Information Networking and Applications*, pp. 106-115, Victoria, Canada, 13-16 May 2014.

[14] C. Shen, Z. Cai, and X. Guan, "Continuous authentication for mouse dynamics: a pattern-growth approach," in *Proc. IEEE/IFIP Int. Conf. on Dependable Systems and Networks, DSN'12*, 12 pp., Boston, MA, USA, 25-28 Jun. 2012.

[15] O. O. Bamasag and S. Arabia, "Towards continuous authentication in internet of things based on secret sharing scheme, " in *Proc. of the Workshop on Embedded Systems Security, WESS'15*, 8 pp., Amsterdam, The Netherlands, 4-9 Oct 2015.

[16] H. Sethi, M. Arkko, J. Keranen, and A. Back, *Practical Considerations and Implementation Experiences in Securing Smart Object Network*s. Draft-Ietf-Lwig-Crypto-Sensors-06, 2018.

[17] C. Bormann, M. Ersue, and A. Kernen, *Terminology for Constrained-Node Networks*, no. 7228. RFC Editor, May 2014.

[18] T. Kothmayr, C. Schmitt, W. Hu, M. Brünig, and G. Carle, "DTLS based security and two-way authentication for the Internet of Things," *Ad Hoc Networks*, vol. 11, no. 8, pp. 2710-2723, Nov. 2013.

[19] E. Rescorla and N. Modadugu, *Datagram Transport Layer Security Version 1.2*. RFC 6347, Internet Engineering Task Force (IETF). 2012.

[20] P. Gope and T. Hwang, "Untraceable sensor movement in distributed IoT infrastructure," *IEEE Sens. J.*, vol. 15, no. 9, pp. 5340-5348, Sept. 2015.

[21] Y. Kawamoto, H. Nishiyama, N. Kato, Y. Shimizu, A. Takahara, and T. Jiang, "Effectively collecting data for the location-based authentication in Internet of Things," *IEEE Syst. J.*, vol. 11, no. 3, pp. 1403-1411, Sept. 2017.

[22] M. Durairaj and K. Muthuramalingam, "A new authentication scheme with elliptical curve cryptography for Internet of Things (IoT) environments," *Int. J. Eng. Technol.*, vol. 7, no. 2, pp. 119-124, 2018.

[23] M. T. Hammi, B. Hammi, P. Bellot, and A. Serhrouchni, "Bubbles of trust: a decentralized blockchain-based authentication system for IoT," *Comput. Secur.*, vol. 78, no. 1, pp. 126-142, Jul. 2018.

[24] S. Banerjee, V. Odelu, A. K. Das, S. Chattopadhyay, and Y. Park, "An efficient, anonymous and robust authentication scheme for smart home environments," *Sensors*, vol. 20, no. 4, pp. 1-19, Feb. 2020.

[25] M. Shuai, N. Yu, H. Wang, and L. Xiong, "Anonymous authentication scheme for smart home environment with provable security," *Comput. Secur.*, vol. 86, no. 3, pp. 132-146, Sept. 2019.

[26] T. Shimshon, R. Moskovitch, L. Rokach, and Y. Elovici, "Continuous verification using keystroke dynamics," in *Proc. Int. Conf. on Computational Intelligence and Security*pp. 411-415, Nanning, China, 11-14 Dec.2010.

[27] P. Porambage, C. Schmitt, P. Kumar, A. Gurtov, and M. Ylianttila, "Two-phase authentication protocol for wireless sensor networks in distributed IoT applications," in *Proc. IEEE Wireless Communications and Networking Conf., WCNC'14*, pp. 2728-2733, Istanbul, Turkey, 6-9 Apr. 2014.

[28] K. O. Bailey, J. S. Okolica, and G. L. Peterson, "User identification and authentication using multi-modal behavioral biometrics," *Comput. Secur.*, vol. 43pp. 77-89, Mar. 2014.
[29] G. Peng, G. Zhou, D. T. Nguyen, X. Qi, Q. Yang, and S. Wang, "Continuous authentication with touch behavioral biometrics and voice on wearable glasses," *IEEE Trans. Human-Machine Syst.*, vol. 47, no. 3, pp. 404-416, Jun. 2017.
[30] K. Niinuma, U. Park, and A. K. Jain, "Soft biometric traits for continuous user authentication," *IEEE Trans. Inf. Forensics Secur.*, vol. 5, no. 4, pp. 771-780, Dec. 2010.
[31] K. Mock, J. Weaver, and M. Milton, "Real-time continuous iris recognition for authentication using an eye tracker," in *Proc. of the 2012 ACM Conf. on Computer and Communications Security, CCS'12,* pp. 1007-1009, Raleigh, NC, USA, 16-18 Oct. 2012.
[32] L. Zhou, C. Su, W. Chiu, and K. Yeh, "You think, therefore you are: transparent authentication system with brainwave-oriented bio-features for IoT networks," *IEEE Trans. Emerg. Top. Comput.*, vol. 8, no. 2, pp. 303-312, Apr. 2020.
[33] P. N. Mahalle, N. R. Prasad, and R. Prasad, "Threshold cryptography-based group authentication (TCGA) scheme for the Internet of Things (IoT)," in *Proc. 4th Inte. Conf. on Wireless Communications, Vehicular Technology, Information Theory and Aerospace & Electronic Systems, VITAE'14*, 5 pp., Aalborg, Denmark, 11-14 May 2014.
[34] S. Seitz, L. Gerdes, S. Selander, G. Mani, and M. Kumar, *Use Cases for Authentication and Authorization in Constrained Environments*, RFC 7744, Internet Engineering Task Force (IETF). 2016.
[35] H. Khemissa and D. Tandjaoui, "A lightweight authentication scheme for E-health applications in the context of Internet of Things," in *Proc. 9th Int. Conf. Next Gener. Mob. Appl. Serv. Technol.*, pp. 90-95, Cambridge, UK, 9-11 Sept. 2015.
[36] H. Khemissa and D. Tandjaoui, "A novel lightweight authentication scheme for heterogeneous wireless sensor networks in the context of Internet of Things," in *Proc. Wirel. Telecommun. Symp.*, 6 pp., London, UK, 18-20 Apr. 2016.
[37] M. Hamada, S. Kumari, and A. Kumar, "Secure anonymous mutual authentication for star two-tier wireless body area networks," *Comput. Methods Programs Biomed.*, vol. 135, pp. 37-50, Jul. 2016.
[38] C. Chen, B. Xiang, T. Wu, and K. Wang, "An anonymous mutual authenticated key agreement scheme for wearable sensors in wireless body area networks*," Appl. Sci. (Basel)*, vol. 8, no. 7, pp. 1-15, Jul. 2018.
[39] Z. Xu, C. Xu, W. Liang, J. Xu, and H. Chen, "A lightweight mutual authentication and key agreement scheme for medical Internet of Things," *IEEE Access*, vol. 7, pp. 53922-53931, 2019.
[40] S. Swain, *Priority Based Rate Control in Wireless Sensor Networks*, 2013.
[41] A. Armando, D. Basin, Y. Boichut, Y. Chevalier, and L. Compagna, "The AVISPA Tool for the Automated Validation," in *Proc. Int. Conf. on Computer Aided Verification, CAV'05,* pp. 281-285, Edinburgh, Scotland, UK, 6-10 Jul. 2005.
[42] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Trans. Inf. Theory*, vol. 29, no. 2, pp. 198-208, Mar. 1983.
[43] R. Amin and G. P. Biswas, "A secure light weight scheme for user authentication and key agreement in multi-gateway based wireless sensor networks," *Ad Hoc Networks*, vol. 36no. 1, pp. 58-80, Jun. 2016.

رضا سرابی میانجی کارشناسی و کارشناسی ارشد مهندسی کامپیوتر- نرم‌افزار را به ترتیب در سال‌های ۱۳۷۹ و ۱۳۸۱ دریافت کرد و از سال ۱۳۹۵ در دکترای مهندسی کامپیوتر- سیستم‌های نرم‌افزاری دانشگاه آزاد اسلامی واحد تهران شمال مشغول تحصیل و تحقیق است. وی چندین کتاب در زمینه شبکه‌های کامپیوتری، برنامه‌نویسی و طراحی صفحات وب، ترجمه و تألیف نموده و از سال ۱۳۷۹ در دانشگاه، مشغول تدریس می‌باشد. همچنین از سال ۱۳۸۰ در تیم‌های مدیریت شبکه‌های کامپیوتری، توسعه نرم‌افزار و مدیریت پایگاه داده بانک مرکزی جمهوری اسلامی ایران مشغول به کار است. علایق تحقیقاتی وی شامل اینترنت اشیا، امنیت سیستم‌های اطلاعاتی، مدیریت و امنیت شبکه‌های کامپیوتری و مدیریت سیستم‌های پایگاه داده می‌باشد.

سام جبه‌داری به عنوان دانشیار گروه مهندسی کامپیوتر دانشگاه آزاد اسلامی واحد تهران شمال مشغول به کار است. او مدرک کارشناسی مهندسی برق مخابرات و کارشناسی ارشد مهندسی برق مخابرات خود را به ترتیب از دانشگاه صنعتی خواجه نصیر طوسی و دانشگاه آزاد اسلامی واحد تهران جنوب و مدرک دکترای مهندسی کامپیوتر را از دانشگاه آزاد اسلامی واحد علوم و تحقیقات دریافت نمود. علایق تحقیقاتی وی زمان‌بندی، QoS، MANET، امنیت شبکه‌های کامپیوتری، شبکه‌های حسگر بی‌سیم، محاسبات ابری، اینترنت اشیا و محاسبات لبه و مه است.

ناصر مدیری در سال ۱۳۶۵ مدرک کارشناسی الکترونیک را از دانشگاه ساسکس انگلستان و در سال ۱۳۶۶ مدرک کارشناسی ارشد الکترونیک را از دانشگاه ساوت‌همپتون انگلستان دریافت کرد. همچنین در سال ۱۳۶۸ مدرک دکترای مهندسی کامپیوتر را از دانشگاه ساسکس انگلستان دریافت نمود. وی از سال ۱۳۷۱ عضو هیأت علمی دانشگاه بوده و در حوزه‌های تخصصی امنیت شبکه‌های کامپیوتری، امنیت نرم‌افزارهای کاربردی و فرایند توسعه امن نرم‌افزار فعالیت می‌نماید. او چندین کتاب تخصصی در حوزه امنیت شبکه‌های کامپیوتری، امنیت نرم‌افزارهای کاربردی و فرایندهای توسعه امن نرم‌افزار چاپ کرده است. علایق تحقیقاتی وی توسعه امن نرم‌افزار، امنیت شبکه‌های کامپیوتری، اینترنت اشیا، ERP، RFID، ISO/IEC ۲۷۰۰۰ و ISO/IEC ۱۵۴۰۸ است.