ارائه روشی جدید جهت مقابله با حمله روم شرقی در سیستم های تشخیص نفوذ با ساختار سلسله‌مراتبی پویا در شبکه‌های موردی سیار

احمد حقیقی[1]، سمانه حاجی رمضان[2]، کیارش میزانیان باغ گلستان[3]

۱- دانشجوی کارشناسی ارشد، دانشکده مهندسی برق و کامپیوتر، دانشگاه یزد، haghighi.ahmad@stu.yazd.ac.ir

۲-دانشجوی کارشناسی ارشد، دانشکده فنی مهندسی، دانشگاه شاهد samaneh.hajiramezan92@gmail.com

۳-استادیار، دانشکده مهندسی برق و کامپیوتر، دانشگاه یزد، k.mizanian@yazd.ac.ir

خلاصه

پیشرفت چند سال اخیر در زمینه فن آوری ارتباطات بی‌سیم موجب پیدایش معماری های جدیدی بر این مبنا شده است. که از میان آن‌ها می توان شبکه‌های موردی سیار را نام برد. وجود مزایایی مانند سرعت زیاد توسعه، سادگی، هزینه بر پایی پایین، عدم نیاز به زیرساخت از پیش تعیین شده و پیکره بندی خودکار در این شبکه‌ها باعث افزایش محبوبیت و کاربرد آن گردیده است.

خصیصه‌های یادشده ازجمله عدم وجود زیرساخت باعث افزایش آسیب‌پذیری این شبکه‌ها گردیده و راهکارهای امنیتی قبلی پاسخ‌گو نیست. لذا نیاز است تا برای رفع مشکل تدابیر جدید امنیتی اتخاذ شود. یکی از روش‌های موجود استفاده از سیستم‌های تشخیص نفوذ با ساختار سلسله‌مراتبی پویا می‌باشد. یکی از مشکلات این روش این است که گره‌های مخرب می‌توانند با همکاری هم یک گره مخرب را به‌عنوان رأس سلسله‌مراتب انتخاب کنند. در این مقاله راهکاری را پیشنهاد داده‌ایم که می‌تواند تا حد قابل قبولی در رفع این مشکل مؤثر واقع گردد.

کلمات کلیدی: سیستم تشخیص نفوذ، شبکه‌های موردی سیار، ساختار سلسله‌مراتبی پویا، حمله‌ی روم شرقی

۱. مقدمه

در سال‌های اخیر محبوبیت و فراوانی دستگاه‌های سیار و اتصالات بی‌سیم به‌طور چشم‌گیری افزایش‌یافته است. عدم نیاز به وجود زیرساخت جهت ارتباط، قابلیت سیار بودن دستگاه‌ها، خود پیکربندی و بی‌سیم بودن اتصالات، از بارزترین خصیصه‌های شبکه‌های موردی سیار است که خود دلیلی بر افزایش روزافزون محبوبیت و کاربرد این نوع از شبکه‌ها می‌باشد[1]. از مهم‌ترین کاربردهای شبکه‌های موردی سیار می‌توان به کاربرد آن در امور نظامی برای ارتباط سربازان و ادوات جنگی با یکدیگر و مراکز فرماندهی، عملیات نجات در مواردی مانند زلزله، سیل و آتش‌سوزی که بسترهای ارتباطی از بین می‌روند، کاربردهای محلی مانند کنفرانس‌ها، کلاس‌های درس و سایر کاربردها که در آن‌ها شرکت‌کنندگان از طریق اتصال گوشی‌ها و لپ‌تاپ‌ها و سایر تجهیزات خود به یکدیگر، به تبادل اطلاعات می‌پردازند، سیستم‌های رأی‌گیری و یا کاربرد آن در موارد شبکه‌های خانگی برای اتصال دستگاه‌ها به یکدیگر اشاره نمود.

در اصل شبکه‌های موردی سیار برای محیط‌های اشتراکی طراحی شده‌اند. برای استفاده از آن‌ها در محیط‌های خصمانه از مسیریابی مبتنی بر اعتماد استفاده می‌شود. اما به دلیل این ساختار مبتنی بر اعتماد مورد حملات زیادی قرار می‌گیرند. چراکه یک گره مورد اعتماد می‌تواند مورد حمله قرار گیرد و به یک گره مخرب تبدیل گردد و سپس در عملیات مسیریابی شرکت کند. بنابراین حضور سیستم‌های تشخیص نفوذ برای شناسایی و جلوگیری از حملات ضروری

---

۱ دانشجوی کارشناسی ارشد، دانشکده مهندسی برق و کامپیوتر، دانشگاه یزد

۲ دانشجوی کارشناسی ارشد، دانشکده فنی مهندسی، دانشگاه شاهد

۳ استادیار، دانشکده مهندسی برق و کامپیوتر، دانشگاه یزد

است. سیستم تشخیص نفوذ یک سیستم (نرم‌افزار یا سخت‌افزار) است که رویدادها را به‌منظور کشف حادثه‌ها نظارت می‌کند. یکی از مهم‌ترین اجزای سیستم تشخیص نفوذ امکان لاگ کردن است. که برای تحلیل رویدادها به‌منظور تشخیص فعالیت‌های غیرمعمول توسط کاربر استفاده می‌شود.

سیستم تشخیص نفوذ بر اساس موقعیت مکانی به دو نوع عمده مبتنی بر شبکه و مبتنی بر میزبان طبقه‌بندی می‌شود.[2]

سیستم تشخیص نفوذ مبتنی بر شبکه[1] روی روتر یا ابزارهای زیرساختی شبکه قرار می‌گیرند. این تکنیک اغلب برای حمله‌کننده و دیگران نامحسوس است و به همان نسبت بسیار تأثیرگذار است. علاوه بر این گره‌ها مجبور به دادن ظرفیت و زمان CPU به عملیات تشخیص نفوذ نیستند. البته به دلیل فقدان زیرساخت شبکه این نوع قابل به‌کارگیری روی شبکه‌های موردی سیار نیست.

سیستم تشخیص نفوذ مبتنی بر میزبان[2]. یک نرم‌افزار سیستمی است که حمله‌های روی لایه کاربردی یا عملیاتی سیستم راشناسایی می‌کنند. بنابراین واکنش گره‌های خاص می‌تواند کشف و گزارش شود. ازآنجایی‌که این سیستم‌ها محاسبات را روی گره انجام می‌دهند؛ ظرفیت در دسترس CPU (در شبکه‌های موردی سیارتوان باتری دستگاه) را کاهش می‌دهد.

سیستم‌های تشخیص نفوذ توزیع‌شده ترکیبی از چند سیستم تشخیص نفوذ مبتنی بر میزبان و مبتنی بر شبکه و یا چند سیستم تشخیص نفوذ مبتنی بر میزبان می‌باشد. که حالت دوم می‌تواند در شبکه‌های موردی سیار به کار گرفته شود. انتخاب مدل سازمان‌دهی اولین گام برای طراحی معماری در هر سیستم تشخیص نفوذ توزیع‌شده می‌باشد. از رایج‌ترین مدل‌های معماری این سیستم، همتا به همتا، سلسله‌مراتبی ایستا و سلسله‌مراتبی پویا می‌باشد.

در مدل سلسله‌مراتبی پویا گره‌ها بر اساس پارامترهایی چون نیروی پردازش، اتصال(تعداد گره‌هایی که می‌توانند به یک گره وصل شوند.)، ظرفیت حافظه، مجاورت با الگوریتم‌های [3,4,5,6] خوشه‌بندی می‌شوند. زمان که سلسله‌مراتب ساخته شد گره‌ها در امتداد خط سلسله‌مراتبی گزارش می‌دهند. آن‌ها سعی می‌کنند تشخیص نفوذ را در پایین‌ترین لایه‌ی ممکن به‌منظور حداقل کردن سربار اتصال و دوره‌ی عکس‌العمل انجام دهند.

در این معماری مشکل حمله‌ی روم شرقی[3] وجود دارد. به این معنی که ممکن است گره‌های مخرب با یکدیگر همکاری کنند و یک گره مخرب را به‌عنوان گره رأس انتخاب کنند. در این مقاله یک بهبود برای رفع مشکل حمله‌ی روم شرقی ارائه شده است. که به میزان قابل‌توجهی از انتخاب گرهی مخرب به‌عنوان رأس جلوگیری می‌کند.

سایر بخش‌های مقاله به شرح زیر می‌باشد. در بخش ۲ پاره‌ای از تحقیقات و کارهای صورت گرفته درزمینه سیستم‌های تشخیص نفوذ در شبکه‌های موردی سیار بیان می‌شود. طرح پیشنهادی برای بهبود معماری سلسله‌مراتبی در بخش ۳ ارائه‌شده است. در بخش ۴ نتایج شبیه‌سازی عملکرد طرح پیشنهادی ارائه و در بخش ۵ جمع‌بندی آورده شده است.

۲. تحقیقات انجام شده

گسترش شبکه‌های موردی سیار و اهمیت تشخیص نفوذ در این‌گونه شبکه‌ها باعث شده است که تحقیقات فراوانی در این زمینه صورت گیرد.

D. Sterne و همکاران در سال ۲۰۰۵معماری تشخیص نفوذ مبتنی بر همکاری را ارائه دادند. در این معماری بعد از تشکیل سلسله‌مراتب گره‌ها در امتداد خط سلسله‌مراتبی با هم مذاکره می‌کنند. آن‌ها سعی می‌کنند تشخیص نفوذ را در پایین‌ترین سطح ممکن انجام دهند. به این صورت که داده‌های بدست آمده از نظارت را از پایین به بالا جمع می‌کنند و از بالا به پایین مورد تصمیم و داوری قرار می‌دهند. سیستم تشخیص نفوذ در این معماری از دو نوع نظارت استفاده می‌کند. اولی نظارت بی‌قید است؛ به این معنا که رأس‌های هر زیر خوشه بر ترافیک داخل سطح خود نظارت می‌کنند. و دومی نظارت بر جریان داده از مبدأ به مقصد تنها در گام‌های اول و آخر مسیر می‌باشد[7].

[1] NIDS
[2] HIDS
[3] Byzantine attack

Chang Katharine و همکاران در سال ۲۰۱۰ معماری تشخیص نفوذ مبتنی بر لایه کاربردی بر اساس سلسله‌مراتب را پیشنهاد دادند. این معماری از عامل محلی برای تشخیص نفوذ استفاده می‌کند. عامل‌های همراه در این معماری از هر دو تکنیک مبتنی بر ناهنجاری و مبتنی بر امضا استفاده می‌کنند. هر عامل شامل ۳ ماژول نظارت و تشخیص، ارتباطات و پاسخ می‌باشند[8].

Zhang Da و همکاران در سال ۲۰۱۰ سیستم تشخیص نفوذ مبتنی بر دادگاه را ارائه دادند. این معماری دادگاه را با اجزای اصلی خود به‌منظور شناسایی گره‌های مخرب مدل می‌کند. و شامل ماژول‌های مختلف مانیتورینگ، اتهام، صدور هشدار،داوری و همچنین گره‌های وکیل می‌باشد.[9]

Manikandan T و همکاران در سال ۲۰۱۰ معماری تشخیص گره‌های مخرب را ارائه دادند. و برای شبکه‌هایی که دارای بیش از ۵ گره یعنی $n > 4k + 1$ باشد استفاده می‌شود.در این معماری یک گره یک پیام به همه‌ی گره‌های در محدوده تحت پوشش خود می‌فرستد. گره‌هایی که بی‌خطر فرض می‌شوند به این پیام پاسخ می‌دهند و گره‌های مخرب پاسخ این پیام را نمی‌دهند. گره اولیه سپس درخواست رأی‌گیری برای گره مخرب می‌کند. پس از جمع‌آوری رأی‌ها، داده‌ها برای اثبات اینکه یک گره مخرب است جمع می‌شوند. ازآنجایی‌که که رأی نادرست و حتی گره‌های رأی نداده وجود دارد. گره ای مشکوک اعلام می‌شود که حداقل $k + 1$ گره علیه آن رأی دهند[10].

Rajaram و Palaniswami در سال ۲۰۱۰ یک سیستم تشخیص نفوذ برای شبکه‌های موردی سیار شامل یک پروتکل امنیتی بر مبنای اعتماد[۱] معرفی کردند که از یک مکانیسم لایه‌ی کنترل دستیابی رسانه[۲] استفاده می‌کند. این پروتکل احراز هویت و قابلیت اعتماد را در هر دولایه‌ی شبکه و پیوند داده[۳] ارائه می‌دهد. در فاز اول پروتکل، یک راهکار بر مبنای اعتماد جهت هدایت بسته‌ها به‌کاربرده می‌شود تا گره‌های بدخواه را شناسایی و جدا[۴] کند. پروتکل از مقداری به‌عنوان مقدار اعتماد[۵] استفاده می‌کند و هرگاه مقدار این شمارنده اعتماد، از حد آستانه‌ای پایین‌تر رفت، آن گره میانی مربوطه به‌عنوان گره بدخواه نشانه‌گذاری می‌شود. در فاز دوم پروتکل، یک مکانیسم امنیتی لایه پیوند به کار گرفته می‌شود که از احراز هویت و رمزنگاری[۶] CBC-X جهت فراهم ساختن امنیت در تبادل پیام‌های سیستم تشخیص نفوذ استفاده می‌کند[11].

Mutlu و Yilmaz در سال ۲۰۱۱ یک سیستم تشخیص نفوذ توزیع‌شده مبتنی بر همکاری معرفی کردند که بر تحلیل‌های محلی و جهانی تکیه داشت. هر گره یک موتور سیستم تشخیص نفوذ محلی دارد که یک نسخه مبتنی بر شبکه از سیستم تشخیص نفوذ را اجرا و بر فعالیت گره‌های همسایه نظارت می‌کند. وقتی یک گره رفتاری مشکوک[۷] را شناسایی می‌کند به‌منظور دریافت تمامی داده‌های مربوط به تشخیص نفوذ، یک الگوریتم تشخیص نفوذ توزیع‌شده را اجرا می‌کند. در این الگوریتم داده‌های رسیده از موتور IDS و همچنین بقیه پیام‌های هشدار[۸] که توسط IDS سایر گره‌ها در شبکه پخش‌شده است، جمع‌آوری و تجزیه تحلیل و مرتبط[۹] می‌شوند. حال اگر شواهد کافی مبنی بر نفوذ وجود داشته باشد یک پیغام هشدار در شبکه همه پخشی می‌شود. اگر پیام هشدار توسط گرهی غیرقابل‌اعتماد فرستاده‌شده باشد، پیام نادیده گرفته می‌شود. به‌محض اینکه گره یک پیام هشدار تشخیص نفوذ دریافت می‌کند عمل جلوگیری از حمله را انجام می‌دهد و میزان قابلیت اعتماد گره مربوطه را کاهش می‌دهد. این عمل توسط تمامی گره‌های که پیام هشدار را دریافت می‌کنند انجام می‌شود[12].

۳. طرح پیشنهادی

در ابتدا گره‌ها بر اساس [3] خوشه‌بندی شده‌اند. سپس برای هر خوشه یک عدد تصادفی به‌عنوان شناسه (CID) انتخاب می‌شود. هر گره CID خوشه‌ای که در آن قرار دارد را می‌داند. فرض می‌کنیم حداقل یک گره ناظر در شبکه داریم که گرهی مورد اعتماد و با آدرسی شناخته‌شده برای تمامی گره‌ها است. چنانچه

---

[۱] Trust-based
[2] Media Access Control(mac)
[3] Link
[4] Isolate
[5] Trust value
[6] Encryption
[7] Suspicious
[8] Alert
[9] Correlate

میزان توان باقی‌مانده و حافظه گره ناظر از یک حدی کمتر شد خودش با توجه به تاریخچه‌اش وظیفه‌اش را به یک گره مورد اعتماد دیگر می‌دهد و تمامی گره‌ها را مطلع می‌سازد. فرض دوم این است که گره ناظر به دلیل نگه داشت انرژی و سایر منابعش نمی‌تواند رأس شود.

جدول ۱- متغیرهای مورد استفاده در الگوریتم

نام متغیر	مقدار
sTime	حداکثر زمانی که یک گره می‌تواند رأس باشد (مقداری ثابت که با توجه به سیاست‌های امنیتی تعیین می‌شود)
dTime	مدت‌زمانی که گره توانایی رأس بودن دارد (مقداری متغیر که گره ناظر بر اساس پارامترهای رسیده از سمت گره، این مقدار را تخمین می‌زند)
CID	شناسه تصادفی مربوط به هر خوشه
hTime	min(dTime,sTime)

۳,۱. الگوریتم پیشنهادی:

۱. تمامی گره‌های هر خوشه(به‌جز رأس قبلی) پارامترهای خود(شارژ باتری باقی‌مانده، حافظه باقی‌مانده، توان پردازشی، احتمال خارج شدن از خوشه در بازه زمانی sTime) را به همراه شناسه خود و خوشه به گره ناظر ارسال می‌کنند.

۲. گره ناظر، ID گره‌هایی را که پارامترهایشان از حد آستانه بالاتر بودند و شرایط رأس شدن دارند را درون یک لیست می‌ریزد.

۳. گره ناظر ، یک عدد تصادفی در بازه [1,m] تولید می‌کند.(m تعداد اعضای داخل آرایه است)، درنتیجه شماره‌ی یکی از خانه‌های آرایه به دست می‌آید و یک گره به‌تصادف انتخاب می‌شود و نامزد رأس شدن می‌گردد.

۴. سپس گره ناظر از سایر گره‌های ناظر (در صورت وجود) و سایر اعضای خوشه درخواست می‌کند تا در مورد قابل‌اعتماد بودن گره نامزد انتخاب‌شده رأی دهند و همچنین خود به توجه به تاریخچه‌اش رأی می‌دهد.

۵. گره ناظر رأی‌ها را جمع‌آوری کرده و طبق فرمول ۱ مقدار مربوط به قابلیت اعتماد گره را محاسبه می‌کند. درصورتی‌که مقدار به‌دست‌آمده از حد آستانه‌ای بیشتر باشد، گره تصادفی را به اعضای خوشه به‌عنوان رأس معرفی می‌کند. این مقدار آستانه با توجه به فرمولی که گره‌ها برای محاسبه میزان اعتماد به کار می‌برند تعیین می‌شود.

۶. گره ناظر برای رأس جدید انتخاب‌شده، مقدار dTime را محاسبه کرده، سپس به میزان hTime به گره اجازه می‌دهد رأس باشد. درصورتی‌که این زمان تمام شود و یا اینکه گره رأس از محدوده خوشه خارج شود و یا با وقوع حادثه‌ای گره نتواند به رأس بودن ادامه دهد، الگوریتم از سر گرفته شده و رأس جدیدی انتخاب می‌گردد.

$$\mathrm{trust_i} = (\alpha(\sum m_{ji})/y + \beta(\sum n_{si})/z_i) \quad (۱)$$

که در آن: $\alpha+\beta=1$

m_{ji} رأی گره ناظر j در مورد گره i (بین ۰ تا ۱)

n_{si} رأی گره s از همسایه‌های گره i در مورد گره i (بین ۰ تا ۱)

z_i تعداد گره‌هایی که با گره i در یک خوشه هستند

y تعداد گره‌های ناظر (حداقل یک)

α ضریب تأثیر برای نظرات گره‌های رأس (بین ۰ تا ۱)

β ضریب تأثیر برای نظرات گره‌های همسایه (بین ۰ تا ۱)

$\mathrm{trust_i}$ نتیجه رأی‌گیری در مورد میزان اعتماد به گره i (مقدار بین ۰ تا ۱)

الگوریتم فوق برای کل خوشه‌های سطح آخر سلسله‌مراتبی اجرا می‌شود و رئوس خوشه‌های سطح آخر تعیین می‌شود. حال برای سطح بعدی سلسله‌مراتبی به میزان زیادی اطمینان داریم گره‌های رأس مخرب نیستند. و می‌خواهیم پدر برای این رئوس انتخاب کنیم، لذا کافی است مراحل ۱ تا ۳ را انجام دهیم و گره پدر را بیابیم. این کار این‌قدر تکرار می‌شود که ریشه‌ی سلسله‌مراتب به دست آید.

درصورتی‌که در شبکه چندین گره ناظر وجود داشته باشد، اعضای هر خوشه به‌صورت تصادفی بر سر یکی از گره‌های ناظر توافق می‌کنند و پارامترهای خود را برای آن گره ارسال می‌کنند و در صورت عدم وجود گرهی ناظر، گره‌های هر خوشه می‌توانند هریک عددی تصادفی تولید نمایند و برای تمامی همسایه‌های خود در خوشه ارسال کنند، سپس با مرتب‌سازی ID های مربوط به گره‌های خوشه ، با جمع این اعداد تصادفی می‌توان به عددی تصادفی رسید که به کمک این عدد می‌توان گرهی را به‌عنوان ناظر به‌تصادف انتخاب کرد که تمامی گره‌ها از آن خبر داشته باشند. هرچند برای انتخاب این ناظر می‌توان رأی‌گیری کرد و میزان اعتماد به این گره را پرس‌وجو نمود ولی همچنان مشکل این روش اعتماد و اطمینان به ناظر انتخابی است، چراکه نقشی کلیدی‌تر و حساس‌تر از رأس‌ها دارد و به‌سادگی نمی‌توان به رأی‌گیری اکتفا نمود. لذا فرض می‌کنیم در شبکه حداقل یک گره ناظر با تعاریف پاراگراف اول از این بخش موجود است.

۴. شبیه‌سازی

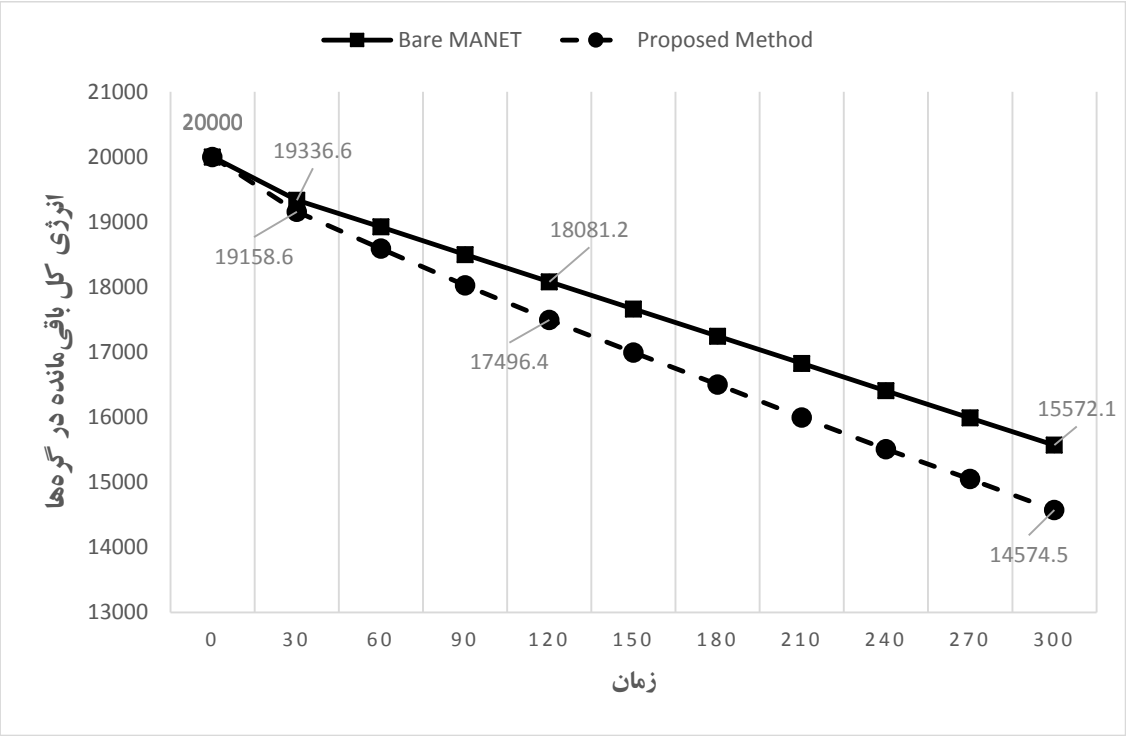
جهت شبیه‌سازی ایده پیشنهادی از برنامه NS2 [13] نسخه ۲,۳۵ در محیط Ubuntu 14.04 LTS 32bit استفاده شده است. پارامترهای شبیه‌سازی در جدول ۲ آورده شده است. در شبیه‌سازی دو مؤلفه تعداد کل بسته‌های ارسال شده و همچنین مقدار کل انرژی باقی‌مانده(جمع انرژی تمامی گره‌ها) محاسبه و در شکل‌های ۱و۲ ترسیم‌شده‌اند.

جدول ۲- پارامترهای شبیه‌سازی

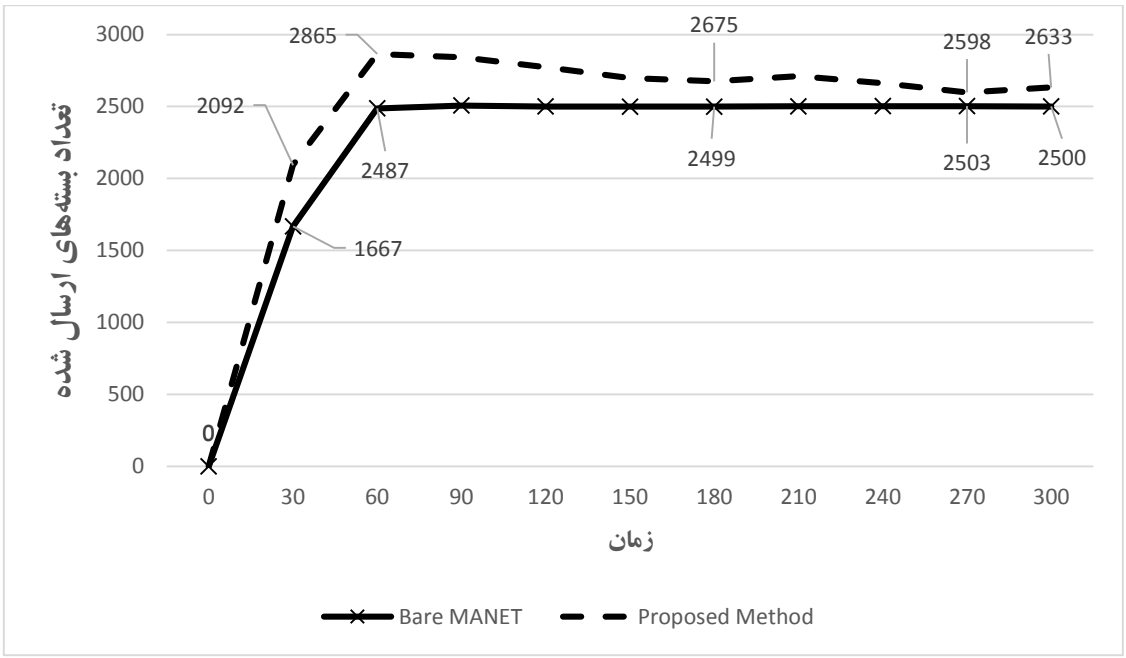
۳۰۰ ثانیه	زمان شبیه‌سازی
۵۰	تعداد گره‌های متحرک
۱	تعداد گره‌های ناظر
۶۷۰ * ۶۷۰	توپولوژی
AODV	پروتکل مسیریابی
متغیر	سرعت گره‌ها
۴۰۰ ژول	انرژی اولیه هر گره
1 CBR and 1 FTP	ترافیک

انتخاب آستانه‌ها و همچنین الگوریتم مناسب جهت رأی‌گیری وابسته به کاربرد و محیط عملیاتی دارد. ما در این شبیه‌سازی برای محاسبه میزان اعتماد گره‌ها به یکدیگر از نسبت تبادلات موفق استفاده کرده‌ایم، به این معنی که اگر در تراکنش‌هایی که قبلاً صورت گرفته است، منع سرویسی از سمت مقابل رخ نداده باشد، اعتبار افزایش و در غیر این صورت کاهش یابد و همچنین گره‌هایی که تبادلی نداشته‌اند اعتبار ۰,۵ را ارسال نموده‌اند. و به گره‌ها با اعتماد بالای ۰,۷ اجازه کاندیدا شدن داده‌ایم.

همان‌طور که انتظار می‌رود در ابتدای شبیه‌سازی به دلیل تشکیل خوشه‌ها و رأی‌گیری در هر خوشه تعداد بسته‌های ارسالی بیشتر از بقیه زمان‌ها بوده و همچنین انرژی با شیب تندتری کاهش‌یافته است، در الگوی حرکتی تصادفی تعیین‌شده برای گره‌ها از میانه‌های شبیه‌سازی به بعد سرعت حرکت گره‌ها کاهش‌یافته و لذا به‌تبع آن خروج گره‌ها از خوشه و لذا رأی‌گیری مجدد کاهش می‌یابد، همچنین انرژی با شیب ملایم‌تری کاهش‌یافته است.



شکل ۱- مقدار کل انرژی باقی‌مانده در گره‌ها



شکل ۲- تعداد کل بسته‌های ارسال‌شده در بازه‌های ۳۰ ثانیه‌ای

۵. نتیجه‌گیری

یکی از راهکارهای بهبود امنیت در شبکه‌های موردی سیار به‌کارگیری سیستم‌های تشخیص نفوذ می‌باشد که به دلیل عدم وجود زیرساخت در این شبکه‌ها نظارت مرکزی و ثابت امری ناکارآمد است، لذا ساختارهای سلسله مراتبی و مخصوصاً نوع پویای آن‌ها که ساختار را بر اساس وضعیت شبکه شکل می‌دهند گزینه مناسبی برای این‌گونه از شبکه‌ها می‌باشد. ما در این مقاله جهت بهبود امنیت و کارایی سیستم‌های تشخیص نفوذ با ساختار سلسله مراتبی پویا روشی ارائه دادیم تا گره‌های رأس در ساختار سلسله مراتبی تا حد قابل قبولی از میان گره‌های مورد اعتماد انتخاب شوند و گره بدخواه به‌عنوان رأس انتخاب نگردد. روش پیشنهادی صرف‌نظر از نوع الگوریتم تشخیص نفوذی که به‌کاربرده می‌شود رأس خوشه‌ها رو همچنین رأس‌های بالاتر را به کمک رأی‌گیری و با استفاده از گره (های) خاصی بانام ناظر انتخاب می‌کند و همچنین از باقی ماندن گرهی با عنوان رأس برای مدتی طولانی جلوگیری می‌کند و درنتیجه اطمینان به مخرب نبودن رأس‌ها را افزایش داده و با حمله روم شرقی مقابله می‌کند.

۶. منابع

[1] M. S. Corson, J. P. Macker, and G. H. Cirincione, “Internet-based mobile ad hoc networking,” *IEEE Internet Comput.*, vol. 3, no. 4, pp. 63–70, 1999.

[2] V. Jaiganesh, S. Mangayarkarasi, and P. Sumathi, “Intrusion Detection Systems : A Survey and Analysis of Classification Techniques,” vol. 2, no. 4, pp. 1629–1635, 2013.

[3] N. H. V. M. C. D. K. P. P. Krishna, P. Krishna, N. H. Vaidya, M. Chatterjee, and D. K. Pradhan, “A cluster-based approach for routing in dynamic networks,” *ACM SIGCOMM Comput. Commun. Rev.*, vol. 27, no. 2, pp. 49–64, Apr. 1997.

[4] S. Basagni, “Distributed clustering for ad hoc networks,” in *Proceedings Fourth International Symposium on Parallel Architectures, Algorithms, and Networks (I-SPAN’99)*, 1999, pp. 310–315.

[5] Y. Huang and W. Lee, “A cooperative intrusion detection system for ad hoc networks,” *Proc. 1st ACM Work. Secur. ad hoc Sens. networks - SASN ’03*, p. 135, 2003.

[6] O. Kachirski and R. Guha, “Effective Intrusion Detection Using Multiple Sensors in Wireless Ad Hoc Networks,” p. 57.1, Jan. 2003.

[7] D. Sterne, P. Balasubramanyam, D. Carman, B. Wilson, R. Talpade, C. Ko, R. Balupari, C.-Y. Tseng, T. Bowen, K. Levitt, and J. Rowe, “A General Cooperative Intrusion Detection Architecture for MANETs,” *Third IEEE Int. Work. Inf. Assur.*, pp. 57–70, 2005.

[8] K. Chang and K. G. Shin, “Application-Layer Intrusion Detection in MANETs,” in *2010 43rd Hawaii International Conference on System Sciences*, 2010, pp. 1–10.

[9] D. Zhang and C. K. Yeo, “A Novel Architecture of Intrusion Detection System,” *2010 7th IEEE Consum. Commun. Netw. Conf.*, pp. 1–5, Jan. 2010.

[10] T. Manikandan, “Detection Of Malicious Nodes in MANETs,” in *Communication Control and Computing Technologies (ICCCCT), 2010 IEEE International Conference on*, 2010, pp. 788 – 793.

[11] L. Bononi and C. Tacconi, “Intrusion detection for secure clustering and routing in Mobile Multi-hop Wireless Networks,” *Int. J. Inf. Secur.*, vol. 6, no. 6, pp. 379–392, Jul. 2007.

[12] S. Mutly and G. Yilmaz, “A Distributed Cooperative Trust Based Intrusion Detection Framework for MANETs,” in *ICNS 2011, The Seventh International Conference on Networking and Services*, 2011, pp. 292–298.

[13] “The Network Simulator - ns-2.” [Online]. Available: http://www.isi.edu/nsnam/ns/. [Accessed: 31-Dec-2014].