یک الگوریتم دقیق جهت شناسایی حمله تکرار گره در شبکه‌های حسگر بی‌سیم متحرک به کمک یک گره ناظر

مجتبی جمشیدی[۱]، مهدی اثنی عشری[۲]، پیمان صیدی[۳]، محمد رضا میبدی[۴]

[۱] آموزشکده فنی و حرفه‌ای سما، دانشگاه آزاد اسلامی، واحد کرمانشاه، کرمانشاه، ایران jamshidi.mojtaba@gmail.com
[۲] پژوهشکده فناوری اطلاعات، پژوهشگاه فضای مجازی، تهران، ایران esnaashari@itrc.ac.ir
[۳] آموزشکده فنی و حرفه‌ای سما، دانشگاه آزاد اسلامی، واحد کرمانشاه، کرمانشاه، ایران peyman_seidi@yahoo.com
[۴] دانشکده کامپیوتر و فناوری اطلاعات، دانشگاه صنعتی امیرکبیر، تهران، ایران mmeybodi@aut.ac.ir

چکیده

حمله تکرار گره (یا گره‌های کپی) یکی از حمله‌های مشهور و خطرناک علیه شبکه‌های حسگر بی‌سیم است. در این حمله، دشمن وارد محیط شبکه شده و یک (یا چند) گره نرمال درون شبکه را ضبط می‌نماید. دشمن سپس، تمام توابع، برنامه‌ها و مواد قفل‌گذاری درون حافظه گره ضبط شده را استخراج نموده و چندین گره کپی از آن تولید و در شبکه تزریق می‌کند. این گره‌های کپی تحت کنترل دشمن می‌باشند و از آن‌جا که دارای مواد قفل‌گذاری معتبر هستند، لذا به راحتی با دیگر گره‌های قانونی شبکه کلید مشترک برپا می‌کنند و به مخابره می‌پردازند. در این مقاله، یک الگوریتم ساده و مقاوم به کمک یک گره ناظر جهت شناسایی گره‌های کپی در شبکه‌های حسگر متحرک ارائه می‌گردد. ایده اصلی الگوریتم پیشنهادی، استفاده از اطلاعات همسایگی در طول تحرک گره‌ها در محیط شبکه جهت شناسایی گره‌های کپی است. کارایی الگوریتم پیشنهادی از نقطه نظرهای سربار ارتباطات، حافظه و پردازش ارزیابی گردیده و نتایج حاصل با دیگر الگوریتم‌های موجود مقایسه شده است که نتایج این مقایسه، برتری الگوریتم پیشنهادی را می‌رساند. هم‌چنین، الگوریتم پیشنهادی توسط شبیه‌ساز JSIM پیاده‌سازی گردیده و با انجام یک‌سری آزمایش‌ها مشخص شد الگوریتم پیشنهادی قادر به شناسایی ۱۰۰٪ گره‌های کپی است درحالی که احتمال تشخیص غلط آن نزدیک به صفر است.

کلمات کلیدی

شبکه‌های حسگر بی‌سیم، امنیت، گره ناظر، گره‌های کپی

۱- مقدمه

یک شبکه حسگر بی‌سیم از صدها تا هزاران گره حسگر منابع محدود (از نظر حافظه، انرژی، پردازش و غیره) تشکیل می‌شود. این نوع شبکه‌ها کاربردهای متنوعی در دامنه‌های نظامی، صنعتی، بهداشتی و علوم دیگر دارند. با توجه به منابع محدود گره‌های حسگر، گسترش بدون مراقبت شبکه حسگر، ماهیت بی‌سیم ارتباطات و نیز کاربرد روز افزون این نوع شبکه‌ها در دامنه‌های نظامی، برقراری امنیت در این شبکه‌ها امری بسیار مهم و چالش‌زا است [4-1].

یکی از حمله‌های خطرناک در شبکه‌های حسگر بی‌سیم، حمله تکرار گره[۱] یا گره کپی[۲] است. با توجه به گسترش بدون مراقبت گره‌ها در محیط عملیاتی، دشمن می‌تواند یک (یا چند) گره قانونی درون شبکه را ضبط و اطلاعات مهم از جمله مواد قفل‌گذاری[۳] درون حافظه آن را استخراج کند و با استفاده از این مواد قفل‌گذاری، گره‌های تکراری (یا گره‌های کپی) ایجاد کند. از آنجا که گره‌های کپی دقیقاً حاوی مشخصات و اطلاعات (از جمله شناسه، مواد قفل‌گذاری و ...) گره قانونی ضبط شده هستند، لذا قابلیت برپایی کلید با دیگر گره‌های قانونی شبکه را دارند. دشمن سپس این گره‌های کپی را در شبکه پخش می‌کند و می‌تواند از این موقعیت درون شبکه‌ای جهت راه‌اندازی حمله‌های مختلف دیگری نیز استفاده کند [5][6][7].

تاکنون الگوریتم‌های نظیر [18-8] جهت مقابله با حمله گره‌های کپی در شبکه‌های حسگر ثابت مطرح شده است. ولی این الگوریتم‌ها در شبکه‌های حسگر متحرک قابل بکارگیری نیستند. در [30-19] نیز الگوریتم‌هایی جهت مقابله با حمله گره‌های کپی در شبکه‌های حسگر متحرک ارائه شده است که به‌طور کلی دارای معایبی نظیر سربار ارتباطات و حافظه بالا، عدم مقیاس‌پذیری، فرایند پیچیده تشخیص گره‌های کپی، نیاز به تعیین مکان گره‌ها و استفاده از کلیدهای عمومی و امضاهای دیجیتال می‌باشند.

در این مقاله، یک الگوریتم دقیق، به کمک یک گره ناظر[۴] [31] و استفاده از اطلاعات همسایگی جهت شناسایی گره‌های کپی در شبکه‌های حسگر بی‌سیم متحرک پیشنهاد می‌گردد، به‌طوری که معایب الگوریتم‌های پیشین را برطرف کند. الگوریتم پیشنهادی نیاز به تعیین مکان گره‌ها، انتشار پیغام‌های ادعای مکانی، کلیدهای عمومی (و امضای دیجیتال) و فرایندهای پیچیده تشخیص گره‌های کپی ندارد و فقط از اطلاعات همسایگی جهت شناسایی گره‌های کپی بهره می‌گیرد.

ادامه این مقاله بدین ترتیب سازماندهی می‌شود: در بخش ۲، کارهای گذشته، در بخش ۳، مدل سیستم و در بخش ۴، الگوریتم پیشنهادی ارائه می‌گردد. نتایج شبیه‌سازی در بخش ۵ و در آخر نتیجه‌گیری آمده است.

۲- کارهای گذشته

۲-۱- الگوریتم‌های خاص شبکه‌های حسگر ثابت

در [12-8] الگوریتم‌هایی احتمالاتی مبتنی بر ارسال پیغام‌های ادعای مکانی و رمزنگاری کلید عمومی استفاده می‌کنند در [13] یک پروتکل به نام SET مطرح شده است که از عملیات مجموعه‌ایی (اجتماع و اشتراک) روی زیرمجموعه‌های انحصاری در شبکه جهت تشخیص گره‌های کپی استفاده می‌کند. در [14] دو الگوریتم مبتنی بر رویکرد "چندپخشی محلی" برای تشخیص گره‌های تکراری مطرح شده است. در [15] یک الگوریتم مبتنی بر فشرده‌سازی داده‌ها به نام CSI جهت تشخیص گره‌های کپی مطرح شده است. در [16] یک الگوریتم مبتنی بر مختصات مکانی و تابع درهم‌سازی جغرافیایی جهت مقابله با حمله تکرار گره مطرح شده است. در [17] دو روش توزیعی جهت شناسایی حمله تکرار گره بر اساس مجموعه‌های متقاطع و مدل تصدیق هویت دو مرحله‌ای سلول محصور شده در شبکه‌های حسگر سلول‌بندی شده، مطرح شده است. در [18] نیز چهار الگوریتم جهت شناسایی حمله تکرار گره مطرح گردیده که از فیلترهای Bloom[32] جهت فشرده‌سازی اطلاعات در حسگرها و نیز دو تکنیک ارسال سلولی و ارسال متقاطع جهت افزایش نرخ تشخیص استفاده می‌کند.

۲-۲- الگوریتم‌های خاص شبکه‌های حسگر متحرک

ایده اصلی الگوریتم [19]، به نام XED، تولید و ارسال اعداد تصادفی توسط گره‌ها در هر بار رویارویی دو گره است. ایده اصلی الگوریتم‌های ارائه شده در [20]، [21] و [22] برگرفته از این حقیقت است که یک گره متحرک ضبط نشده (قانونی) نباید هرگز در سرعتی بیش از حداکثر سرعت سیستم پیکربندی شده حرکت کند. وجود گره‌های کپی u، سبب می‌شود گره‌های دیگر گمان کنند این گره u با سرعتی بیش از حداکثر سرعت از پیش تعریف شده حرکت می‌کند. در این صورت، می‌توانند گره u را به عنوان گره کپی علامت زنند. ایده اصلی دو الگوریتم ارائه شده در [23] برگرفته از این ملاحظه است که برای یک شبکه بدون گره تکراری، در یک دوره زمانی مشخص با طول T، تعداد دفعات رویارویی گره u با یک گره خاص v به احتمال زیاد باید محدود باشد. برای یک شبکه با دو گره تکراری v، تعداد دفعات رویارویی گره u با گره v در یک دوره زمانی با طول T باید بزرگتر از یک آستانه مشخص باشد. بر طبق این ملاحظه، هر گره قابلیت شناسایی گره‌های تکراری را دارد.

ایده اصلی الگوریتم [24] بر این اساس است که کل شبکه به سکتورهایی تقسیم می‌شود و هر سکتور یک گره مرکزی دارد که از روش‌هایی نظیر تشخیص شناسه، تشخیص همسایه‌های تکراری و آرایه‌ی ردیابی (ذخیره مکان‌های گره‌ها) استفاده می‌کند تا گره‌های حسگر را شناسایی کند. ایده اصلی الگوریتم [25]، استفاده از پروتکل پیش‌توزیع کلید جفتی مبتنی بر چندجمله‌ای و فیلترهای شمارشی Bloom جهت مقابله با حمله تکرار گره است. ایده اصلی الگوریتم [26]، SHD، مبادله لیست همسایه‌ها میان گره‌های متحرک و انتخاب گره‌های شاهد برای عمل تشخیص است. در [27] نیز دو راه‌حل به نام‌های UTLSE و MTLSE، برای شناسایی گره‌های تکراری در شبکه‌های حسگر متحرک ارائه شده است که در آن، ادعاهای مکانی فقط هنگام رویارویی دو گره شاهد، مبادله می‌گردد. ایده اصلی الگوریتم ارائه شده در [28]، استفاده از یک تصدیق هویت مبتنی بر نشانه جهت تشخیص گره‌های کپی است. در [29] نیز یک الگوریتم دیگر ارائه شده است که فقط از ارتباطات تک‌گامه و تحرک گره جهت شناسایی گره‌های تکراری در شبکه‌های حسگر متحرک استفاده می‌کند. در [30] نیز دو الگوریتم در حوزه زمان و فضا جهت شناسایی حمله گره‌های کپی در شبکه‌های موردی متحرک ارائه شده است. در این دو الگوریتم از تابع درهم‌سازی تک-راهه رمزنگاری جهت مقابله با حمله گره‌های کپی بهره گرفته شده است.

۳- فرضیات سیستم و مدل حمله

شبکه حاوی n گره حسگر است که به‌طور تصادفی در یک محیط دوبعدی پراکنده می‌شوند. هم‌چنین یک گره ناظر در شبکه گسترش می‌یابد که وظیفه آن شناسایی گره‌های کپی احتمالی در شبکه است. هر گره یک شناسه یکتا دارد و از موقعیت مکانی خود آگاه نیست. برد رادیویی تمام گره‌ها یکسان است. تمام گره‌ها متحرک می‌باشند و در طول حیات شبکه مطابق مدل‌های تحرک، نظیر Random waypoint در محیط شبکه حرکت می‌کنند. فرض می‌شود گره‌های حسگر (بجز گره ناظر) در برابر مداخله مقاوم نیستند و دشمن در صورت ضبط یک گره می‌تواند به اطلاعات محرمانه آن دسترسی داشته باشد و آن را برنامه‌ریزی مجدد کند[33][34]. هم‌چنین، با توجه به متحرک بودن گره‌های حسگر در محیط شبکه، گره‌ها می‌بایست به‌طور پریودیک (مثلاً بعد از هر t واحد زمانی یا پس از این‌که به یک مکان جدید در شبکه می‌رسند) یک پیغام "Hello"، درخواست مسیر، ارسال داده و یا زنده بودن[5] از خود منتشر کنند [35]. این عمل درواقع یکی از نیازمندی‌های شبکه‌های حسگر متحرک است تا هر گره بتواند در هر لحظه از زمان، از همسایه‌های جاری خود آگاه باشد، در صورت نیاز با آن‌ها کلیدهای امنیتی برپا کند، با آنها مخابره کند و جدول مسیریابی خود را ایجاد کند. البته در این‌جا، گره ناظر می‌تواند از ارسال پریودیک این‌گونه پیغام‌ها خودداری کند تا حضورش از دید دیگر گره‌ها مخفی بماند. چراکه این گره، وظیفه شناسایی گره‌های بدخواه دشمن را بر عهده دارد و بهتر است از دید دیگر گره‌ها مخفی بماند.

هم‌چنین فرض می‌شود شبکه حسگر در یک محیط خصمانه گسترش می‌یابد، بنابر این، شبکه ناامن بوده و دشمن می‌تواند گره‌هایی را ضبط کند و کپی‌هایی از این گره‌های ضبط شده را ایجاد و سپس در شبکه تزریق کند. هم‌چنین فرض می‌شود هر گره کپی نیز در هر دوره زمانی t، یک پیغام "Hello"، درخواست مسیر، ارسال داده یا زنده بودن منتشر می‌کند. گره‌های کپی، می‌توانند متحرک باشند و یا دشمن آنها را به‌طور ثابت در شبکه مستقر نماید. هیچ یک از این دو حالت تأثیری بر الگوریتم پیشنهادی ندارد.

۴- الگوریتم پیشنهادی

ایده اصلی الگوریتم پیشنهادی جهت شناسایی گره‌های کپی، نظارت بر ترافیک و تحرک گره‌های حسگر توسط تک گره ناظر در شبکه است. درواقع، گره ناظر با توجه به پیغام‌های "Hello" (و یا درخواست مسیر، ارسال داده یا زنده بودن) منتشر شده توسط گره‌ها در طول حیات شبکه، از همسایه‌های جاری خود آگاه شده و از همین اطلاعات همسایگی جهت شناسایی گره‌های کپی استفاده می‌کند. الگوریتم پیشنهادی فقط و فقط بر روی تک گره ناظر بار شده و اجرا می‌گردد.

الگوریتم پیشنهادی از ۳ فاز تشکیل شده است. در فاز اول، گره ناظر پیکربندی می‌شود. در فاز دوم، گره ناظر ترافیک و تحرک دیگر گره‌های شبکه را نظارت نموده و اطلاعات لازم را در حافظه خود ثبت می‌نماید. در فاز سوم، گره ناظر باتوجه به اطلاعات جمع‌آوری نموده در طی فاز دوم، اقدام به شناسایی گره‌های کپی می‌کند. در ادامه به تشرح این ۳ فاز می‌پردازیم.

۴-۱- فاز اول (پیکربندی گره ناظر)

قبل از گسترش گره ناظر در محیط عملیاتی مدنظر، در حافظه آن یک ماتریس بالامثلثی به نام *history*، مطابق شکل (۱)،تنظیم می‌شود. تعداد سطر و ستون‌های این ماتریس برابر تعداد کل گره‌ها در شبکه، یعنی n، است. شناسه هر گره، N_i، در سطر اول و قطر این ماتریس ذخیره می‌گردد. هم‌چنین به هر سلول از این ماتریس یک مقدار (اعشاری) اولیه، مطابق رابطه (۱) نسبت داده می‌شود.

$$history[N_i][N_j] = \frac{1}{n \times (n-1)/2} \qquad (۱)$$

واضح است، حاصل‌جمع مقادیر تمام سلول‌های این ماتریس برابر ۱ خواهد بود. سپس گره ناظر در شبکه گسترش می‌یابد.

N_1	N_2	...	N_{n-1}	N_n
N_1				
	N_2			
		...		
			N_{n-1}	
				N_n

شکل (۱): ساختار ماتریس history گره ناظر

۴-۲- فاز دوم (نظارت بر ترافیک و تحرک گره‌ها)

نظارت بر ترافیک و تحرک گره‌ها، به‌طور متناوب در طول حیات شبکه، توسط گره ناظر انجام می‌گیرد. اجرای این فاز شامل چندین دور نظارت بر ترافیک و تحرک گره‌ها در شبکه است. مطابق مدل تحرک تصادفی درنظر گرفته شده، در هر دور از این فاز، هر گره یک مقصد تصادفی برای خود انتخاب نموده و شروع به حرکت به سوی آن مقصد می‌کند. پس از رسیدن به مقصد، برای مدت زمانی (مثلاً t ثانیه) در آنجا ساکن مانده و شروع به ارسال پیغام‌های "Hello"، داده‌ای، درخواست مسیر یا ... می‌کند. این سبب می‌شود گره ناظر در زمان‌های مختلف از حیات شبکه، از همسایه‌های جاری خود آگاه شود. گره‌ها سپس شروع به حرکت به سوی یک مقصد تصادفی دیگر می‌کنند و به این ترتیب دور بعدی فاز دوم آغاز می‌شود.

گره ناظر در پایان هر دور از فاز دوم، ماتریس *history* خود را مطابق الگوریتم ارائه شده در شکل (۲) بروزرسانی می‌کند. گره ناظر ابتدا تعداد همسایه‌های جاری خود را شمارش نموده و سپس به ازای هر یک از آنها، یعنی $N_i \in CurrentNeighbor$، دو فرآیند زیر را انجام می‌دهد:

۱- به ازای هر $N_j \notin CurrentNeighbor$، از مقدار درون سلول متناظر با این دو گره N_i و N_j، یعنی $history[N_i][N_j]$ یا $history[N_j][N_i]$، به میزان ضریب α کاهش داده و در یک متغیر انباشته‌گر به نام *Sum* ذخیره می‌کند.

۲- به ازای هر $N_j \in CurrentNeighbor$، به‌طوری که $N_j \neq N_i$ باشد، سهمی از *Sum* را به مقدارسلول متناظر با این دو گره N_i و N_j، یعنی $history[N_i][N_j]$ یا $history[N_j][N_i]$، اضافه می‌کند.

به‌طور خلاصه، الگوریتم بروزرسانی، به ازای هر دو گره‌ای که به صورت انحصاری (یعنی فقط یکی از آن دو) در همسایگی گره ناظر ظاهر شده باشند، از مقدار سلول متناظر با این دو گره، به یک میزان مشخص (ضریب α) کاهش و به سلول‌های متناظر با آن دو گره‌هایی که به‌صورت همزمان در همسایگی گره ناظر حضور پیدا کرده‌اند اضافه می‌کند.از این‌رو، مقدار هر سلول، مثلاً $history[N_i][N_j]$، متأثر از تعداد دفعاتی خواهد بود که این دو گره N_i و N_j به‌طور همزمان یا انحصاری در همسایگی گره ناظر حضور پیدا نموده‌اند. هرچه این دو گره به تعداد دفعات بیشتری به‌طور همزمان در همسایگی گره ناظر حضور پیدا کنند، مقدار سلول متناظر با آنها در ماتریس *history* افزایش می‌یابد و بلعکس، هرچه این دو گره به تعداد دفعات بیشتری به‌طور انحصاری (فقط یکی از آن دو) در همسایگی گره ناظر حضور پیدا کنند، مقدار سلول متناظر با آنها در ماتریس *history* کاهش می‌یابد. باید توجه شود، چنانچه در یک دور از اجرای فاز دوم، هیچ یک از دو گره N_i و N_j در همسایگی گره ناظر حضور پیدا نکنند، مقدار سلول متناظر با این دو گره بدون تغییر خواهد ماند.

بنابر این، در یک شبکه بدون وجود گره‌های تکراری، بعد از اجرای R دور از فاز دوم، مقدار تمام سلول‌های ماتریس *history* تقریباً برابر خواهد بود (با یک واریانس کم). ولی چنانچه تعدادی گره تکراری، مثلاً با شناسه N_i، در شبکه وجود داشته باشد، در این صورت، گره با شناسه N_i به مراتب بیشتر از حالت نرمال، به‌همراه سایر گره‌ها در همسایگی گره ناظر ظاهر می‌شود. این سبب می‌شود سلول‌هایی که از سطر و ستون N_i می‌گذرند، به مراتب مقدار بیشتری نسبت به سایر سلول‌ها داشته باشند. در فاز سوم الگوریتم پیشنهادی، از همین موضوع جهت شناسایی گره‌های کپی بهره گرفته می‌شود.

توجه شود الگوریتم بروزرسانی ماتریس *history* به نحوی است که حاصل‌جمع مقادیر تمام سلول‌های این ماتریس، در تمام لحظات اجرای الگوریتم پیشنهادی، همواره برابر مقدار ۱ خواهد شد. با افزایش تعداد دورهای فاز دوم الگوریتم پیشنهادی، مقادیر سلول‌های متناظر با گره‌های تکراری رشد بیشتری داشته و اختلاف فاحشی با سایر سلول‌ها خواهند داشت.

۴-۳- فاز سوم (شناسایی گره‌های تکراری)

پس از پایان اجرای فاز دوم الگوریتم پیشنهادی، گره ناظر اقدام به علامت زدن گره‌های کپی می‌کند. همان‌طور که اشاره شد، پس از پایان فاز دوم، مقدار سلول‌های عبوری از سطر و ستون متناظر با گره تکراری، به مراتب بیشتر از سایر سلول‌ها خواهد شد. گره ناظر با بهره‌گیری از این موضوع می‌تواند به راحتی و با دقت بالا گره‌های تکراری راشناسایی کند.

فرآیند فاز سوم چنین است که گره ناظر به ازای هر N_i، حاصل‌جمع مقادیر سلول‌های عبوری از سطر و ستون N_i را محاسبه نموده و چنانچه مقدار حاصل بزرگتر از آستانه T_s باشد، گره N_i را تکراری (کپی) تلقی می‌کند. آستانه T_s بسیار ساده و از رابطه زیر محاسبه می‌گردد:

$$E_v = (n-1) \times \frac{1}{n \times (n-1)/2} \qquad (۲)$$

$$T_s = 2 \times E_v$$

که در این‌جا، E_v مقدار مورد انتظار برای حاصل‌جمع مقادیر سلول‌های عبوری از سطر و ستون متناظر با یک گره در حالت معمول (عدم وجود گره‌های تکراری در شبکه) است. از آنجا که وجود گره‌های تکراری سبب می‌شود حاصل جمع مقادیر سلول‌های عبوری از سطر و ستون متناظر با یک گره خاص (همان گره تکراری) بیشتر از مقدار مورد انتظار (E_v) در حالت معمول گردد، لذا گره ناظر، چنانچه حاصل‌جمع مقادیر سلول‌های عبوری از سطر و ستون متناظر با یک گره خاص را بزرگتر از دو برابر مقدار مورد انتظار (E_v) ببیند، این گره را به عنوان گره تکراری علامت می‌زند.

```
d = |CurrentNeighbor|
for each N_i ∈ CurrentNeighbor Do
    Sum = 0.0
    for each N_j ∉ CurrentNeighbor Do
        Sum = Sum + (α × history[N_i][N_j]) // or history[N_j][N_i]
        history[N_i][N_j] = (1 − α) × history[N_i][N_j]
    end for
    for each N_j ∈ CurrentNeighbor and N_j ≠ N_i Do
        history[N_i][N_j] = history[N_i][N_j] + Sum/(d − 1)
    end for
end for
```

شکل (۲): شبه‌کد الگوریتم بروزرسانی ماتریس history گره ناظر

۵- ارزیابی کارایی و نتایج شبیه‌سازی

۵-۱- سربار الگوریتم پیشنهادی

سربار حافظه: از آن‌جا که الگوریتم پیشنهادی فقط بر روی تک گره ناظر اجرا می‌شود، لذا هیچ سربار حافظه‌ای بر گره‌های حسگر معمولی تحمیل نمی‌کند. ولی گره ناظر یک فضا از حافظه به اندازه d برای بردار $CurrentNeighbor$ و یک فضا به اندازه $\frac{n\times(n-1)}{2}$ برای ماتریس $history$ نیاز دارد. بنابر این، سربار حافظه مربوط به تک گره ناظر از مرتبه $O(n^2)$ و سربار حافظه مربوط گره‌های حسگر معمولی برابر صفراست. در جدول (۱) سربار حافظه الگوریتم پیشنهادی و دیگر الگوریتم‌های موجود مقایسه شده است. به وضوح روشن است الگوریتم پیشنهادی از حیث سربار حافظه، برتر از سایر الگوریتم‌ها می‌باشد.

جدول (۱): مقایسه سربار حافظه و ارتباطات الگوریتم پیشنهادی با دیگر الگوریتم‌ها

الگوریتم	سربار حافظه	سربار ارتباطات
LSM[8]	$O(\sqrt{n})$	$O(n\sqrt{n})$
SET[13]	$O(n/T)$	$O(n)$
RED[9]	$O(d)$	$O(n\sqrt{n})$
RAWL[12]	$O(\log n\times\sqrt{n})$	$O(\log n\times\sqrt{n})$
XED[19]	$O(4\times d\times E[X])$	$O(1)$
SPRT[22]	$O(n\sqrt{n})$	$O(n)$
Algorithm[25]	$O(d)$	$O(n\times\log n)$
TDD, SDD[30]	$O(n)$	$O(\sqrt{n}),O(d)$
Proposed Algorithm	$0\sim O(n^2)$	صفر

سربار ارتباطات: با توجه به محدودیت‌های انرژی گره‌های حسگر، میزان انرژی مصرفی الگوریتم‌های ارائه شده برای شبکه‌های حسگر یک موضوع مهم است. عملیات ارسال بسته، دریافت بسته و پردازش از مجموعه اعمال مهمی هستند که انرژی مصرف می‌کنند. از آن‌جا که عمل ارسال بسته‌ها نسبت به عمل پردازش بسته‌ها و دریافت بسته‌ها انرژی خیلی بیشتری مصرف می‌کند لذا محاسبه تعداد بسته‌های ارسالی که به دلیل استفاده از یک الگوریتم خاص به شبکه تحمیل می‌شود (یا همان سربار ارتباطات)، یک معیار مهم جهت ارزیابی کارایی الگوریتم‌های مطرح شده برای شبکه‌های حسگر است.

از آن‌جا که الگوریتم پیشنهادی، فقط از پیغام‌های "Hello"، درخواست مسیر، ارسال داده و زنده بودن جهت اجرا بهره می‌گیرد و با توجه به اینکه ارسال این دست پیغام‌ها جزء نیازمندی‌های شبکه‌های حسگر متحرک است (مستقل از وجود/ عدم وجود الگوریتم پیشنهادی)، لذا هیچ گونه سربار ارتباطاتی به گره‌های حسگر معمولی تحمیل نمی‌گردد. هم‌چنین، گره ناظر هیچ‌گونه پیغامی در راستای اجرای الگوریتم پیشنهادی ارسال نمی‌کند. لذا سربار ارتباطی برای گره ناظر نیز صفر است. در جدول (۱)، سربار ارتباطات الگوریتم پیشنهادی و سایر الگوریتم‌های موجود مقایسه شده است. نتیجه این مقایسه حاکی از برتری الگوریتم پیشنهادی از نظر سربار ارتباطاتی است.

۵-۲- نتایج شبیه‌سازی‌ها

الگوریتم پیشنهادی توسط نرم‌افزار شبیه‌ساز JSIM[36] پیاده‌سازی گردیده و با انجام یک‌سری آزمایش‌ها، کارایی آن در قالب معیارهای احتمال تشخیص و احتمال تشخیص غلط ارزیابی شده است:

احتمال تشخیص (P_d): احتمال شناسایی تمام گره‌های کپی در شبکه، پس از R دور نظارت بر ترافیک و تحرک گره‌ها در الگوریتم پیشنهادی است.

احتمال تشخیص غلط (P_f): احتمال شناسایی یک گره غیرکپی به اشتباه به عنوان یک گره کپی توسط الگوریتم پیشنهادی است.

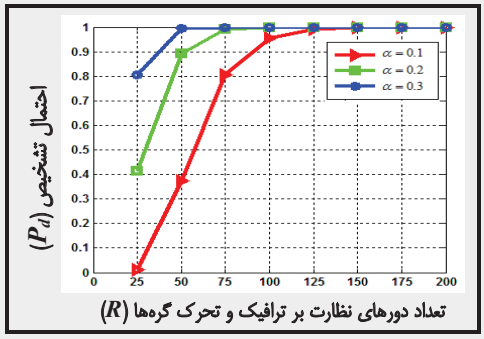
در اجرای شبیه‌سازی‌ها، فرض می‌شود شبکه حاوی n گره حسگر است که به‌طور تصادفی در یک ناحیه دوبعدی ۱۰۰×۱۰۰ متر مربع پراکنده شده‌اند. دشمن M گره قانونی درون شبکه را ضبط نموده و از روی هرکدام λ گره کپی ایجاد و در شبکه منتشر می‌کند. بنابر این، محیط عملیاتی شبکه حاوی $M\times\lambda$ گره بدخواه است. تعداد دورهای فاز دوم الگوریتم پیشنهادی نیز R درنظر گرفته شده است. برد رادیویی تمام گره‌ها نیز ۱۵ متر درنظر گرفته شده است (بجز آزمایش آخر). هم‌چنین، ما از مدل حرکت در نظر گرفته شده در مرجع [19] جهت حرکت گره‌ها در محیط عملیاتی استفاده می‌کنیم. به‌منظور اطمینان از اعتبار نتایج، نتایج نهایی از میانگین ۱۰۰ تکرار بدست آمده است.

آزمایش ۱: هدف این آزمایش، بررسی تأثیر پارامترهای α و R بر کارایی الگوریتم پیشنهادی است. در این آزمایش، تعداد کل گره‌ها در شبکه $n=300$، تعداد گره‌های ضبط شده $M=5$ و از هر گره ضبط شده به تعداد $\lambda=5$ گره کپی در شبکه منتشر می‌شود. این آزمایش به ازای مقادیر مختلف α و $R=25$ تا $R=200$ دور از اجرای فاز دوم الگوریتم پیشنهادی ارزیابی شده است. شکل‌های (۳) و (۴) نتایج حاصل از این آزمایش را به ترتیب در قالب احتمال تشخیص و احتمال تشخیص غلط نشان می‌دهند.

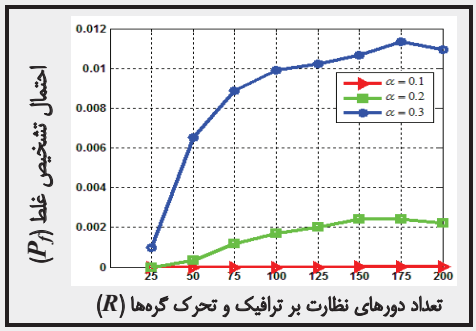
نتایج این آزمایش در شکل (۳) نشان می‌دهد بعد از R=25 دور از فاز دوم، به ازای $\alpha=0.1,0.2,0.3$، احتمال تشخیص الگوریتم به ترتیب برابر $P_d=0.0128$، $P_d=0.41$ و $P_d=0.80$ می‌باشد. هم‌چنین، به ازای $\alpha=0.1,0.2,0.3$، احتمال تشخیص به ترتیب بعد از R=50,75,125 دور از فاز دوم، برابر $P_d=1$ خواهد بود. واضح است با افزایش پارامتر α، در هر دور از فاز دوم که یک گره کپی، نظیر N_i، در همسایگی گره ناظر ظاهر شود، مقدار بیشتری از سلول‌های سایر گره‌ها کاهش و به سلول‌های عبوری از این گره کپی اضافه می‌گردد. در نتیجه حاصل‌جمع سلول‌های عبوری از سطر و ستون N_i خیلی سریع‌تر از آستانه T_s تجاوز خواهد کرد. هم‌چنین، بسیار روشن است که با افزایش R، احتمال تشخیص الگوریتم پیشنهادی افزایش می‌یابد چراکه گره‌های کپی با شناسه مثلاً N_i به تعداد دفعات بیشتری به همراه سایر گره‌های قانونی در همسایگی گره ناظر ظاهر می‌شود و در نتیجه سلول‌های عبوری از سطر و ستون متناظر با N_i مقدار بزرگتری به خود می‌گیرند که این سبب افزایش احتمال تشخیص خواهد شد.

از طرف دیگر، با افزایش پارامتر α و R، احتمال تشخیص غلط نیز افزایش می‌یابد. به عنوان مثال، همان‌طورکه از نتایج این آزمایش در شکل

(۴) مشخص است، به ازای $\alpha = 0.1$، احتمال تشخیص غلط تقریباً $P_f = 0$ می‌باشد. درحالی که، به ازای $\alpha = 0.2$ و $\alpha = 0.3$، احتمال تشخیص غلط حداکثر برابر $P_f = 0.002$ و $P_f = 0.011$ خواهد بود. دلیل این نتیجه این است که، این احتمال وجود دارد برخی گره‌های قانونی به‌طور تصادفی، اندکی بیشتر از سایر گره‌های قانونی در همسایگی گره ناظر حضور پیدا کنند. در این صورت، چنانچه پارامتر α بزرگ انتخاب شده باشد و یا R افزایش یابد، مقدار سلول‌های عبوری متناظر با این گره‌های قانونی بیشتر رشد می‌کند و به اشتباه به عنوان گره‌های تکراری تشخیص داده خواهند شد.



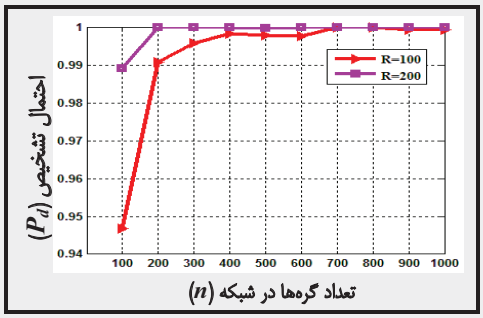
شکل (۳): تأثیر پارامتر α بر احتمال تشخیص الگوریتم پیشنهادی



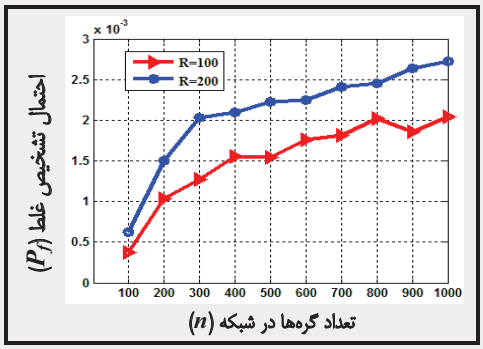
شکل (۴): تأثیر پارامتر α بر احتمال تشخیص غلط الگوریتم پیشنهادی

آزمایش ۲: هدف این آزمایش، بررسی تأثیر تعداد کل گره‌ها در شبکه، n، بر کارایی الگوریتم پیشنهادی است. در این آزمایش پارامترهای $M = 10, \alpha = 0.2, \lambda = 5$ تنظیم شده و احتمال تشخیص و احتمال تشخیص غلط الگوریتم پیشنهادی را برای $n = 100, \ldots, 1000$، یک بار به ازای R=100 و یک بار دیگر به ازای R=200 ارزیابی نموده‌ایم. شکل‌های (۵) و (۶) نتایج این آزمایش را به ترتیب در قالب احتمال تشخیص و احتمال تشخیص غلط نشان می‌دهند. نتایج این آزمایش نشان می‌دهد با افزایش تعداد کل گره‌ها در شبکه، احتمال تشخیص الگوریتم پیشنهادی به‌طرز چشمگیری افزایش می‌یابد و به $P_d = 1$ می‌رسد. این حاکی از مقیاس‌پذیری الگوریتم پیشنهادی است که آن را مناسب بکارگیری در شبکه‌هایی با چگالی‌های مختلف کرده است. با افزایش n، گره کپی u در هر دور از فاز دوم، به همراه تعداد گره‌های قانونی بیشتری در همسایگی گره ناظر حضور پیدا می‌کند. این سبب می‌شود حاصل‌جمع مقادیر سلول‌های عبوری از سطر و ستون متناظر با این گره ضبط شده خیلی سریع‌تر از آستانه T_s تجاوز کند. از این‌رو، احتمال تشخیص افزایش می‌یابد.

هم‌چنین، نتایج حاصل در شکل (۶) نشان می‌دهد با افزایش پارامتر n، احتمال تشخیص غلط الگوریتم پیشنهادی نیز افزایش می‌یابد. به عنوان مثال، بعد از R=200 دور از فاز دوم، زمانی که $n = 100$ گره در شبکه وجود داشته باشد، احتمال تشخیص غلط $P_f = 0.0006$ است. درحالی که اگر $n = 1000$ گره در شبکه وجود داشته باشد، احتمال تشخیص غلط $P_f = 0.0027$ خواهد بود. یعنی از $n = 1000$ گره، حداکثر ۳ گره به اشتباه به عنوان گره‌های کپی تشخیص داده می‌شود که مقدار بسیار ناچیز و قابل تحملی است. دلیل این نتیجه این است، چنانچه تعداد گره‌ها در شبکه افزایش یابد، احتمال این‌که برخی گره‌های قانونی به تصادف و به تعداد دفعات نسبتاً زیاد در همسایگی گره ناظر حضور پیدا کنند، افزایش می‌یابد. این سبب می‌گردد حاصل‌جمع مقادیر سلول‌های عبوری از سطر و ستون‌های متناظر با این گره‌های قانونی بیشتر از آستانه T_s شود و درنتیجه الگوریتم پیشنهادی برخی گره‌های قانونی را به اشتباه به عنوان گره‌های کپی تشخیص می‌دهد.



شکل (۵): تأثیر پارامتر n بر احتمال تشخیص الگوریتم پیشنهادی



شکل (۶): تأثیر پارامتر n بر احتمال تشخیص غلط الگوریتم پیشنهادی

آزمایش ۳: هدف این آزمایش، مقایسه کارایی الگوریتم پیشنهادی با چند الگوریتم دیگر از نظر معیار احتمال تشخیص است. در این آزمایش، تعداد کل گره‌ها n=1000 در نظر گرفته شده است. هم‌چنین، مشکل‌ترین حالت برپایی حمله گره‌های کپی، یعنی $\lambda = 2, M = 1$ لحاظ شده است. برد رادیویی گره‌ها نیز به گونه‌ای تنظیم شده است که هر گره تقریباً d=20 همسایه داشته باشد. جدول (۲) لیست الگوریتم‌های مورد ارزیابی به همراه پارامترهای تنظیم شده و نیز نتایج بدست آمده از آزمایش را نشان می‌دهد. همان‌طور که از نتایج این آزمایش مشخص است، در الگوریتم‌های LSM، B-MEM، BC-MEM، C-MEM و CC-MEM احتمال تشخیص $P_d < 1$ می‌باشد. ولی در الگوریتم پیشنهادی، با تنظیم پارامتر $\alpha = 0.3$، بعد از R=100 دور نظارت بر ترافیک، احتمال تشخیص $P_d = 1$ و احتمال تشخیص غلط $P_f = 0.005$ خواهد بود. چراکه الگوریتم پیشنهادی مختص شبکه‌های متحرک است و در چندین دور اجرا می‌گردد ولی سایر الگوریتم‌های مورد مقایسه خاص شبکه‌های حسگر ثابت بوده و فقط در یک دور اجرا می‌گردند. لذا احتمال تشخیص آنها کمتر از الگوریتم پیشنهادی است. البته احتمال تشخیص غلط برای

الگوریتم‌های خاص شبکه‌های حسگر ثابت (بسته به ماهیت الگوریتم) معمولاً ۰٪ است درحالی که برای الگوریتم‌های خاص شبکه‌های حسگر متحرک (بسته به ماهیت الگوریتم) معمولاً بزرگتر از ۰٪ است. از این رو، در این آزمایش، کارایی الگوریتم پیشنهادی را فقط از نظر احتمال تشخیص گره‌های کپی با سایر الگوریتم‌ها (خاص شبکه‌های ثابت) مقایسه نموده‌ایم.

جدول (۲): مقایسه کارایی الگوریتم پیشنهادی و پنج الگوریتم دیگر در قالب احتمال تشخیص گره‌های کپی

نام الگوریتم	پارامترهای تنظیم شده	احتمال تشخیص
LSM [8]	# line segment=6	0.89
B-MEM [18]	# line segment=6	0.86
BC-MEM [18]	# line segment=5	0.93
C-MEM [18]	-	0.95
CC-MEM [18]	-	0.99
Proposed Alg.	$\alpha = 0.3, R = 100$	1

۶- نتیجه‌گیری

در این مقاله، یک الگوریتم ساده و دقیق به کمک یک گره ناظر جهت مقابله با حمله گره‌های کپی در شبکه‌های حسگر متحرک ارائه گردید. ایده اصلی الگوریتم پیشنهادی، استفاده از اطلاعات همسایگی در طول تحرک گره‌ها در محیط شبکه جهت شناسایی گره‌های کپی است. کارایی الگوریتم پیشنهادی از نقطه نظرهای سربار ارتباطات، حافظه و پردازش ارزیابی گردید و نتایج حاصل با دیگر الگوریتم‌های موجود مقایسه شد که نتایج این مقایسه حاکی از کارایی مطلوب الگوریتم پیشنهادی است. هم‌چنین، نتایج شبیه‌سازی‌ها نشان داد الگوریتم پیشنهادی قادر به شناسایی ۱۰۰٪ گره‌های کپی است و احتمال تشخیص غلط آن بسیار ناچیز است.

مراجع

[1] Akyildiz I. F. and et al., "***A survey on sensor networks***", IEEE Communication Magazine, Vol. 40, pp. 102-114, 2002.

[2] Yick J., Mukherjee B. and Ghosal D., "***Wireless sensor network survey***", Computer Networks 52, pp. 2292–2330, 2008.

[3] Walters J.P., Liang Z., Shi W. and Chaudhary V., "***Wireless Sensor Network Security: A Survey***", Distributed, Grid, and Pervasive Computing, Vol. 1, Issue 2, CRC Press, pp. 1-50, 2007.

[4] Goldsmith A.J. and Wicker S.B., "***Design challenges for energy-constrained ad hoc wireless networks***", in: Proceedings of the IEEE Wireless Commuications, pp. 8–27, August 2002.

[5] Karlof C. And Wagner D., "***Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures***", AdHoc Networks, pp. 299-302, 2003.

[6] D. Liu, P. Ning, "***Establishing pairwise keys in distributed sensor networks***", in: Proceedings of the ACM Conference on Computer and Communications Security, pp. 52–61, 2003.

[7] Padmavathi G. and shanmugapriya D., "***A survey of attacks, security mechanisms and Challenges in Wireless sensor networks***", International Journal of Computer Science and Information Security (IJCSIS), Vol. 4, No. 1 & 2, 2009.

[8] Parno B., Perrig A., and Gligor V. D., "***Distributed Detection of NodeReplication Attacks in Sensor Networks***", in: Proceedings of theIEEE Symposium on Security and Privacy, 2005.

[9] Conti M., Pietro R. D., and Mancini L. V., "***A Randomized, Efficient, and Distributed Protocol for the Detection of Node Replication Attacks in Wireless Sensor Networks***", In: Proceedings of the ACM MobiHoc, 2007.

[10] Conti M. and et al., "***Distributed Detection of Clone Attacksin Wireless Sensor Networks***", IEEE TRANSACTIONS ON DEPEDABLE AND SECURE COMPUTING, 2010.

[11] Kim C., Park C., Hur J., Lee H., and Yoon H., "***A Distributed Deterministic and ResilientReplication Attack Detection Protocol inWireless Sensor Networks***", Communications in Computer and Information ScienceVolume 56, pp 405-412, 2009.

[12] Zeng Y. and et al., "***Random-Walk Based Approach to Detect Clone Attacks in Wireless Sensor Networks***", IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL. 28, NO. 5, 2010.

[13] Choi H., Zhu S., and Porta T. F. La, "***SET: Detecting Node Clones inSensor Networks***", in: Proceedings of the SecureComm '07, pp. 341–350, 2007.

[14] Zhu B., Addada V. G. K., Setia S., Jajodia S., and Roy S., "***EfficientDistributed Detection of Node Replication Attacks in Sensor Networks***", in: Proceedings of theAnnual Computer Security Applications Conference (ACSAC), 2007.

[15] Yu C.-M., Lu C.-S., Kuo S.-Y., "***CSI: Compressed Sensing-Based Clone Identification in Sensor Networks***",in: Proceedings of the8th IEEE International Workshop on Sensor Networks and Systems for Pervasive Computing, 2012.

[16] KIM C. and et al.,"***A Resilient and Efficient Replication Attack Detection Schema for Wireless Sensor Network***", IEICE TRANS. INF. & SYST., VOL. E92-D, NO. 7, 2009.

[17] Bekara C. and Laurent-Maknavicius M., "***A new protocol for securingwireless sensor networks against nodes replication attacks***", in: Proceedings of the: Third IEEE International Conferenceon Wireless and Mobile Computing, Networking and Communications WIMOB '0,Washington, DC, USA, 2007.

[18] Zhang M. and et al., "***Memory Efficient Protocols for Detecting Node Replication Attacks in Wireless Sensor Networks***", in: Proceedings of the 17th annual IEEE International Conference on Network Protocols, Princeton, NJ, USA, 2009.

[19] Yu C. M. and et al., "***Mobile Sensor Network ResilientAgainst Node Replication Attacks***" In: Proceedings of the IEEE Conference onSensor, Mesh and Ad Hoc Communications and Networks (SECON), 2008.

[20] Ho J.-W. and et al., "***Fast detection of replica nodeattacks in mobile sensor networks using sequential analysis***", In: Proceedings of the IEEEINFOCOM, pp. 1773 – 1781, 2009.

[21] Ho J.-W., Wright M., and Das S., "***Fast Detection of Mobile Replica Node Attacks in Wireless Sensor Networks Using Sequential Hypothesis Testing***", IEEE TRANSACTIONS ON MOBILE COMPUTING, VOL. 10, NO. 6, JUNE 2011.

[22] Unnikrishnan D. and et al.,"***Detecting Mobile Replica Node Attacks in Wireless Sensor Networks Using Sequential Probability Ratio Test***", in: Proceedings of the 13th International Conference on Distributed Computing and Networking (ICDCN), Hong Kong, China, 2012.

[23] Yu C.-M., Lu C.-S., and Kuo S.-Y., "***Efficient and DistributedDetection of Node Replication Attacks in Mobile Sensor Networks***", In: Proceedings of the IEEE Vehicular Technology Conf. Fall (VTC Fall),2009.

[24] Gowtham B., Sharmila S., "***Location Traced Hybrid Detection of Node Replication Attack in Mobile Wireless Sensor Network***", In: Proceedings of the Special Issue of International Journal of Computer Applications on Information Processing and Remote Computing – IPRC, 2012.

[25] Deng XM, Xiong Y., "***A new protocol for the detection of node replication attacks in mobile wireless sensor networks***", Journal of Computer Science and Technology26(4),pp. 732-743, 2011.

[26] Yxainaxniga Y. and et al," ***Single Hop Detection of Node Clone Attacks in Mobile Wireless Sensor Networks***", In: Proceedings of the International Workshop on Information and Electronics Engineering (IWIEE), Vol.29, pp. 2798–2803, 2012.

[27] Deng X., Xiong Y., and Chen D., "***Mobility-assisted Detection of the Replication Attacks in Mobile Wireless Sensor Networks***",In: Proceedings of the6th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, 2010

[28] Zhu W. T. and et al., "***Detecting node replication attacks in mobile sensor networks: theory and approaches***", Security and Communication Networks,Vol. 5, pp. 496–507, 2012.

[29] ContiM., Pietro R. D. and Spognardi A.,"***Wireless Sensor Replica Detection in Mobile Environments***", in: Proceedings of the ICDCN, pp. 249-264,2012.

[30] Xing K. an Cheng X., "***From Time Domain to Space Domain: Detecting Replica Attacks in Mobile Ad Hoc Networks***", In: Proceedings of the IEEE INFOCOM, 2010.

[31] S. Marti, T.J. Giuli, K. Lai, and M. Baker, "***Mitigating Routing Misbehavior in Mobile Ad Hoc Networks***", in: Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking, pp. 255-265, 2000.

[32] Bloom B H., "***Space/time trade-offs in hash coding with allowable errors***", *Commun. ACM*, Vol. 13(7), pp. 422-426, 1970.

[33] Shi E., Perrig A., "***Designing secure sensor networks***", *IEEE Wireless Communications,Vol. 11, pp.* 38–43, *2004*.

[34] Tumrongwittayapak C. and Varakulsiripunth R., "***Detecting Sinkhole Attacks In Wireless Sensor Networks***", in: Proceedings of the ICROS-SICE International Joint Conference, 2009.

[35] Piro C.and et al., "***Detecting the Sybil Attack in Mobile Ad hoc Networks***", in: Proceedings of the *Securecomm and Workshops*, pp 1-11, 2006.

[36] ***J-SIM Simulator***, http://www.j-sim.org.

زیرنویس‌ها

[1] node replication attack
[2] Replica node
[3] keying materials
[4] Observer Nodes
[5] Keep alive message