مجله سیاست دفاعی
سال بیستم، شماره ۷۹، تابستان ۱۳۹۱

پروتکلی دفاعی جهت امن‌سازی پیام‌های کوتاه در مناطق عملیاتی

شهریار محمدی [۱]
فرزاد توکلی [۲]

تاریخ دریافت مقاله: ۱۳۹۱/۰۱/۲۸
تاریخ تأیید مقاله: ۱۳۹۱/۰۳/۰۹
صفحات مقاله: ۲۱۱ – ۱۸۳

چکیده:

یکی از مشکلات عمده امنیتی، تبادل پیام در مناطق عملیات نظامی، دسترسی و رمزگشایی پیام توسط گیرنده مورد نظر و همچنین جلوگیری از دسترسی‌های غیر مجاز توسط مهاجمین در مسیر کانال تبادلی می‌باشد. روش‌های معمول رمز نگاری پیام‌های نظامی عمدتاً از امکان دستیابی دشمن به پیام حین ارسال، مصون نمی‌باشند. واژه رمزنگاری مبتنی بر موقعیت مکانی[۳]، در یک تعریف ساده به مفهوم روشی از رمزنگاری است که متن رمز شده فقط در یک محل خاص قابل رمزگشایی است. این الگوریتم‌ها جایگزین روش‌های سنتی رمزنگاری نیستند بلکه یک لایه امنیتی اضافی فراتر از آنچه رمزنگاری‌های سنتی فراهم می‌کنند، ایجاد می‌کنند. در این مقاله، یک پروتکل جدید رمزنگاری مبتنی بر موقعیت مکانی پیشنهاد شده است که امکان تبادل اطلاعات رمز شده را به صورت کاملاً امن، مناسب مناطق عملیاتی در اختیار کاربران قرار می‌دهد. گرچه تمرکز مدل رمزنگاری بر روی پیام کوتاه تبیین شده اما مدل پیشنهادی برای رمزنگاری هر نوع داده مانند صدا و تصویر و ... قابل تعمیم می‌باشد. یکی از چالش‌های پیش روی صنایع با فناوری بالا علی‌الخصوص در صنایع نظامی، تضمین استفاده از ادوات نظامی در یک منطقه خاص مثلاً در سطح یک کشور می‌باشد. در این مدل پیشنهادی می‌توان کلید مجوز راه اندازی یک سیستم نظامی و یا کلید رمزگشایی یک پیام محرمانه مانند دستورالعمل‌های دفاعی و یا نظامی را با موقعیت مکانی استخراج شده از سیستم موقعیت یاب جهانی[۴] تلفیق کرده و یک لایه امنیتی دیگر به سطوح امنیتی پیش بینی شده برای سیستم اضافه نمود. به طور کلی ویژگی مدل پیشنهادی در سه موضوع قابل طرح است:

۱- استادیار گروه مدیریت فناوری اطلاعات (IT)، دانشکده مهندسی صنایع، دانشگاه صنعتی خواجه نصیرالدین طوسی.
۲- پژوهشگر مرکز مطالعات دفاعی و امنیت ملی، گروه مدیریت فناوری اطلاعات (IT)، دانشکده مدیریت، دانشگاه تهران.
3 – Geo-encryption
4 – GPS

۱- استفاده از الگوریتم‌های سبک رمزنگاری و درهم سازی منتج از پروژه eSTREAM-Profile (software) *۲- عدم استفاده از الگوریتم رمزنگاری نامتقارن به دلیل ردپا و اثر[1] بالا و نیاز به مدیریت کلید ۳- استفاده از الگوریتم بهینه شده دفی هلمن[2] به منظور تولید کلید نشست با استفاده از یک رمز عبور کوتاه.*

* * * * *

واژگان کلیدی

رمزنگاری، پیام کوتاه، موقعیت مکانی، رمز دنباله‌ای، توابع درهم ساز، سرویس‌های مبتنی بر موقعیت.

مقدمه

رمزنگاری مبتنی بر موقعیت، امنیت را مناسب با مناطق عملیات نظامی از طریق اختلاط موقعیت مکانی، زمان و حتی سرعت با استفاده از فرآیندهای رمزنگاری و رمزگشایی ارتقا می‌بخشد. اما از منظر رمز نگاری این امر به سادگی محقق نمی‌شود و با مسائلی همچون تولید و انتقال کلید درگیر است.

خط مشی رمزنگاری مبتنی بر موقعیت، بر اساس الگوریتم‌های رمزنگاری و پروتکل‌هایی بنا نهاده شده است به نحوی که یک لایه امنیتی اضافی فراتر از آن‌چه رمزنگاری مرسوم فراهم می‌کند، ایجاد می‌کند. این خط مشی اجازه می‌دهد داده‌ها در مکان(ها) یا محدوده (ها) ی مشخصی رمزگذاری و رمزگشایی شوند مثلاً در محدوده یک منطقه نظامی. مضافاً به این‌که می‌توان محدودیت زمانی را نیز همانند محدودیت مکانی به سیستم اعمال نمود یعنی پیام‌ها در محدوده زمانی و مکانی خاص رمزگشایی شوند. این فناوری هم در ادوات ثابت و هم در ادوات همراه مانند موبایل کاربرد داشته و طیف وسیعی از داده‌های به اشتراک گذاشته شده و راهبردهای انتشار و توزیع داده را پشتیبانی می‌کند(Scott & Denning, 2003).

1 - Footprint
2 - Diffe-Hellman

نکته مهمی که در این نوع رمزنگاری باید همواره مورد توجه باشد تمهیداتی است که جلوی عملیات ضد امنیتی مهاجم را در خصوص میانبر زدن ویژگی‌های مبتنی بر موقعیت بگیرد.

از سوی دیگر طی این سال‌ها الگوریتم‌های متعددی برای امنیت تبادل اطلاعات مطرح شده است، از الگوریتم‌های رمزنگاری کلید متقارن و نامتقارن گرفته تا رمزنگاری دنباله‌ای و بلاکی. اما این روش‌ها مستقل از مکان می‌باشند یعنی فرستنده پیام رمز شده قادر به محدود کردن مکان گیرنده برای رمزگشایی پیام نمی‌باشد. اگر الگوریتم و یا پروتکلی طراحی شود که این قابلیت را فراهم آورد برای افزایش امنیت داده در فضای انتقال بی سیم که به صورت عمده‌ای در مناطق عملیات نظامی مورد استفاده‌اند، بسیار کارآمد خواهد بود.

این لایه امنیتی در مواردی که لازم است پیامی فقط در محدوده خاص جغرافیایی قابل رمزگشایی باشد، بسیار کارا خواهد بود. مثلاً در صنایع نظامی برای حفظ امنیت پیام‌ها در جریان عملیات نظامی علیه دشمن، با استفاده از این طرح اول نیاز به افشای پیام قبل از موعد مقرر نبوده و دوم در مکانی خارج از محدوده عملیات پیام قابل رمزگشایی نیست. علاوه بر این، با استفاده از این طرح می‌توان بر دغدغه استفاده از ادوات نظامی علی‌الخصوص با فناوری بالا در خارج از منطقه مورد توافق فروشنده و خریدار فائق آمد.

در مدل پیشنهادی از پارامترهای موقعیت مکانی (طول/عرض جغرافیایی) به عنوان کلید رمزگذاری داده استفاده شده است. البته دستگاه‌های موقعیت یاب جهانی علاوه بر این دو پارامتر، مقادیر دیگری مانند سرعت، زمان و ارتفاع را نیز در اختیار می‌گذارند که می‌توان از آن‌ها برای محدود کردن شرایط رمزگشایی استفاده نمود مثلاً رمزگشایی در موقعیت منطقه‌ای خاص و در محدوده زمانی مشخص.

پارامترهای رمزگشایی (طول/ عرض جغرافیایی) توسط دستگاه موقعیت یاب جهانی استخراج شده و پیام رمز شده فقط در همان مکان مورد انتظار قابل رمزگشایی خواهد بود. اما از آنجایی‌که دقت این دستگاه‌ها یکسان نبوده و علاوه بر آن در شرایط جوی مختلف، دقت آنها متفاوت می‌باشد، نمی‌توان انتظار داشت که دقیقاً در یک نقطه مشخص برای رمزگشایی

قرار گرفت. از اینرو مسافتی را به عنوان بازه قابل اغماض در نظر گرفته تا پیام در آن محـدوده قابل رمزگشایی باشد.

رمزنگاری مبتنی بر موقعیت[1]

در یک تعریف ساده به روشی از رمزنگاری است که متن رمز شده فقـط در یـک محـل خاص قابل رمزگشایی است. اگر تلاشی برای رمزگشایی داده‌های رمز شـده در مکـانی دیگـر صورت پذیرد فرایند رمزگشایی با شکست مواجه شـده و اطلاعـاتی را در خصـوص پیـام بـر نمی‌گرداند. تجهیزاتی که برای رمزگشایی استفاده مـی‌شـود موقعیـت مکـانی را بـا اسـتفاده از حسگرهای موقعیت یاب مانند یک سیستم موقعیت یاب جهانی و یا دیگر سیستم‌های موقعیت یاب مبتنی بر فرکانس رادیویی تعیین می‌کنند(Scott & Denning, 2003).

سیستم موقعیت یاب جهانی یک سیستم ناوبری مبتنی بر ماهواره بوده کـه از شـبکه‌ای از ۲٤ ماهواره در مدار زمین که متعلق به وزارت دفاع آمریکاست، تشکیل شده است. این سیستم در اصل برای کاربردهای نظامی بوده است اما در سال ۱۹۸۰ توسط دولت امریکا برای استفاده غیرنظامی در اختیار قرار گرفت. البته به دلیل سیاسـت وزارت دفـاع آمریکـا در خصـوص «در دسترس بودن انتخابی»[2] که از طریق تضعیف توان سیگنال‌های ماهواره‌ای به منظـور مقابلـه بـا دشمن در استفاده از سیگنال‌های پرقدرت دستگاه‌های موقعیت یاب جهانی، تـا مـاه مـی سـال ۲۰۰۰ به انجام می‌رسید، عملاً این دستگاه‌ها برای کاربردهای عمومی به‌کار نمی‌رفت. اما پـس از این تاریخ استفاده از دستگاه‌های موقعیت یاب جهانی دستی، برای ناوبری در فواصـل چنـد متر امکان پذیر گردید.

با ظهور فناوری سیستم موقعیت یاب مکانی جهانی تفاضـلی[3]، دسترسـی بـه دقـت‌هـای کمتر از یک متر نیز محقق گردیده است. امروزه گیرنده‌های موقعیت یـاب مکـانی در زنـدگی روزمره ما عمومیت پیدا کرده‌اند مثلاً در سیستم ناوبری خـودرو، هوانـوردی و در کاربردهـای

1 – Geo-encryption
2 – Selective Availability (SA)
3 – Differential GPS (DGPS)

متعدد ورزشی مانند کوه‌نوردی، شکار و غیره. اگرچه در گذشته این سیستم به عنوان یک دستگاه جانبی از طریق کابل و یا بلوتوث به دستگاه‌های تلفن همراه متصل می‌شد اما امروزه به جزئی لاینفک از آن بدل شده است.

مروری بر فعالیت‌های انجام شده

فراگیری ادوات مکان یاب باعث اهمیت‌بخشی به سرویس‌های مبتنی بر موقعیت گردیده است. در سال‌های اخیر این سرویس به عنوان فیلدی از محاسبات همراه گسترش یافته است. تعریف ساده‌ای از سرویس‌های مبتنی بر موقعیت توسط انجمن GSM که کنسرسیومی از ۶۰۰ اپراتور می‌باشد به شرح ذیل ارائه شده است: »سرویس‌های مبتنی بر موقعیت، سرویس‌هایی هستند که از موقعیت هدف برای ارزش افزوده به سرویس استفاده می‌شود« در اینجا منظور از هدف همان موجودیتی است که تعیین موقعیت می‌شود و لزوماً استفاده کننده از سرویس نمی‌باشد (Kupper, 2005).

سرویس‌های مبتنی بر موقعیت به چهار دسته قابل تقسیم می‌باشند: سرویس‌های اورژانسی مانند هشدارهای امنیتی، امنیت اجتماعی و سرویس‌های اطلاعاتی مانند اخبار، آب و هوا، خرید و ...، سرویس‌های ردگیری مانند ردگیری نظامی، ره گیری کالا و ... و در نهایت سرویس‌های سرگرمی مانند بازی، موزیک و غیره (Liao & Chao, 2008).

یکی از کاربردهای اولیه رمزنگاری مبتنی بر موقعیت در خصوص مسئله حق نشر[1] فیلم‌های دیجیتال مطرح شده است. این ایده توسط دنینگ و اسکات[2] مطرح شد و به عبارتی می‌توان گفت مفهوم رمزنگاری مبتنی بر موقعیت اولین بار توسط این دو نفر ارائه گردید. در این طرح با استفاده از کلید رمزنگاری مبتنی بر موقعیت، امکان نمایش فیلم در سالن‌های سینمای مجاز در محدوده مکانی خاص امکان پذیر گردید. البته لازمه این کار به‌کارگیری

1 – Copyright
2 – L. Scott, D. Denning

ویدئو پروژکتورهای دیجیتال بوده که قابلیت اتصال به شبکه‌های ارتباطی را در خود دارند (Scott & Denning, 2003).

بر اساس این ایده اولیه کاربردهای مختلفی برای این تکنیک مطرح گردید. لیائو و چائو[1] مدلی برای رمزنگاری پیام مبتنی بر موقعیت کاربران تلفن همراه ارائه کردند. الگوریتم ارائه شده توسط آن‌ها LDEA[2] نامیده شده و بر اساس رمزنگاری هیبرید عمل می‌کرده است یعنی برای رمزنگاری پیام از الگوریتم کلید متقارن و برای تبادل کلید آن از رمزنگاری کلید نامتقارن استفاده شد (Liao & Chao, 2008).

ژانگ[3] برای تصدیق هویت و تحقق امنیت تبادل اطلاعات بین گره‌های شبکه‌های حسگر استفاده از موقعیت مکانی گره‌های مجاور را مطرح کرد. در طرح دیگری یک مدل به نام LEDS[4] برای امنیت تبادل اطلاعات در گره‌های شبکه‌های حسگر مبتنی بر موقعیت جغرافیایی گره ارائه گردید (Ren et al., 2008).

در شبکه اختصاصی حمل و نقل[5] یان و الاریو[6] مدلی برای تحقق محرمانگی تبادل اطلاعات و تصدیق هویت ارائه کردند که در آن اطلاعات موقعیت مکانی شامل مکان، سرعت و زمان به عنوان کلید رمزنگاری پیام استفاده می‌شد. در این مدل نیز از دو مرحله رمزنگاری/رمزگشایی استفاده شده است یکی برای پیام و دیگری برای انتقال کلید در کانال غیر امن (Yan & Olariu, 2009).

کیاو[7] از دانشگاه استنفورد یک پروتکل امنیتی مبتنی بر سیگنال Loran ارائه کرد. Loran یک سیستم ناوبری فرکانس پایین می‌باشد که در مقابل دخل و تصرف و استفاده غیر مجاز

1 – H.C. Liao, Y.H. Chao
2 – Location-dependent data encryption algorithm
3 – Y. Zhang
4 – Location-aware End-to-end Data Security
5 – Vehicular adhoc network (VANET)
6 – G. Yan, S. Olariu
7 – D. Qiu

مقاوم می‌باشد. این پروتکل به عنوانTESLA [1] ارائه و پیاده سازی شد. نتایج این تحقیق نشان داد که پروتکل مذکور در مقابل حملات جعل هویت[2] موقعیت مکانی بسیار مستحکم و قوی عمل می‌کند (Qiu, 2007).

مدل پیشنهادی برای رمزنگاری مناسب در مناطق عملیاتی

پروتکل پیشنهادی در این نوشتار برای تحقق رمزنگاری پیام کوتاه تبیین می‌شود اما با توجه به ساختار مدل قادر به تعمیم به انواع مختلف انتقال داده می‌باشد مثلاً به‌جای انتقال پیام کوتاه می‌توان رمزنگاری را بر روی یک فایل حاوی پیام انجام داد. پروتکل رمزنگاری پیام کوتاه مبتنی بر موقعیت مکانی از چهار ماژول اصلی به شرح ذیل تشکیل شده است:

- ماژول استخراج پارامترهای موقعیت مکانی؛
- ماژول تولید کلید؛
- ماژول رمزنگاری/رمزگشایی؛
- ماژول مولد پیام کوتاه رمز شده.

ماژول استخراج پارامترهای موقعیت مکانی

بر اساس تعریف NMEA [3] فرمت مختصات بدست آمده از گیرنده‌های موقعیت یاب جهانی بر اساس استاندارد WGS84 [4] به‌جای درجه، دقیقه، ثانیه به صورت درجه، دقیقه با اعشار می‌باشد. فرمت نمایش اطلاعات در این استاندارد به شکل HDD(D)MM.MMMM است که در آن H نمایانگر نیمکره و D زاویه بر حسب درجه و M زاویه بر حسب دقیقه می‌باشد. یعنی مقدار ثانیه با تبدیل به دقیقه در قالب یک عدد با ممیز شناور نمایش داده می‌شود. واضح است که محدوده قابل قبول برای عرض جغرافیایی [90.0,90.0-] بوده که در عرض‌های شمالی مثبت و در عرض جنوبی منفی است. این محدوده برای طول جغرافیایی [180.0,180.0-]

1 – Timed Efficient Stream Loss-tolerant Authentication
2 – Spoofing attacks
3 – National Marine Electronics Association
4 – World geodetic system 1984

می‌باشد کـه بـرای شـرق مثبـت و بـرای غـرب منفـی مـی‌باشـد. مـثلاً در اسـتاندارد مـذکور E12012.5638 به معنی 121 درجه و 12.5638 دقیقه طول جغرافیایی شـرقی و N1202.3452 به معنی 12 درجه و 2.3452 دقیقه عرض جغرافیایی شمالی می‌باشد.

خوشبختانه برای استفاده از این پارامترها به عنوان کلید، واسط برنامه نویسـی کـاربردی[1] لازم تحـت عنـوان JSR 179 در محـیط J2ME وجـود دارد(Mahmoud, 2004) و از طریـق آن علاوه بر استخراج پارامترهای مکانی امکاناتی برای تبدیل انواع مختصات به یکدیگر در اختیار برنامه نویس قرار دارد. برای اسـتفاده از پارامترهـای موقعیـت مکـانی بـه منظـور تولیـد کلیـد رمزنگاری/ رمزگشایی ابتدا مقادیر طول و عرض جغرافیایی بدست آمـده از گیرنـده موقعیـت یاب جهانی که به فرمت WGS84 می‌باشد به فرمت درجه تبدیل می‌گردند. البته برای رهایی از مقادیر اعشاری دو اقدام به شرح ذیل صورت می‌گیرد:

- تبدیل به ثانیه

$$lat_s = lat_d * 3600 + lat_m * 60 + lat_s \quad (۱)$$

- استفاده از دو رقم اعشار در ثانیه و تبدیل آن به عدد صحیح

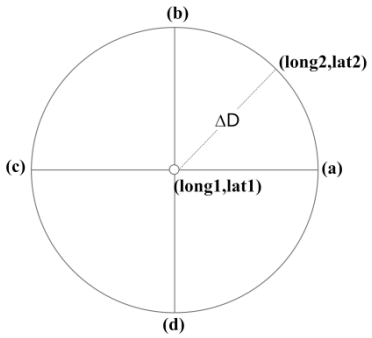
$$lat_s = lat_s * 100 \quad (۲)$$

عملیات مذکور برای طول جغرافیایی نیز محاسبه می‌شود. مقادیر مختصات بدست آمـده از روابط فوق بر حسب درجه می‌باشد اما محدوده رمزگشایی ΔD بازه قابل قبول بـرای تعیـین محدوده رمزگشایی) بر حسب متر در نظر گرفته شده اسـت. بـرای تبـدیل ایـن دو واحـد بـه یکدیگر می‌توان از رابطه فاصله دو نقطه بر روی کره زمین (Spherical law of cosines) استفاده نمود. با توجه به اینکه در این رابطه مقدار فاصله مشخص است می‌توان مختصات دوم که در فاصله ΔD از نقطه اولیه رمزنگاری را محاسبه نمود.

$$\Delta D = \text{acos}(\sin(lat1) * \sin(lat2) + \cos(lat1) * \cos(lat2) * \cos(long2 - long1)) * R \quad (۳)$$

1 – API (Application Programming Interface)

که در آن R=6371Km و شعاع متوسط زمین است. با توجه به شکل شماره‌ی (۱) نقاطی که به فاصله ΔD از نقطه اصلی مورد هدف رمزگذاری قرار دارند بر روی دایره‌ای به مرکز (long1,lat1) و شعاع ΔD قرار دارند.



شکل شماره‌ی ۱ – مختصات نقاط در محدوده‌ی نقطه هدف

نکته‌ی مهم در خصوص استفاده از پارامترهای موقعیت مکانی در تولید کلید اینجاست که اگر این مقادیر به طور مستقیم در فرآیند تولید کلید مورد استفاده قرار گیرند در آن صورت احتمال تولید کلید یکسان بسیار پایین می‌آید چرا که مثلاً با ۰.۰۱ ثانیه انحراف از موقعیت اصلی کلید جدیدی تولید می‌شود که امکان رمزگشایی را نمی‌دهد. از اینرو با استفاده از ΔD و روابط ٤ و ٥ ، مقادیر را نرمالیزه کرده و به عنوان پارامتر تولید کلید مورد استفاده قرار می‌دهیم. در هنگام رمزگشایی کافی ست علاوه بر نقطه مرکزی یک واحد قبل و بعد از هر مختصات را نیز وارد الگوریتم رمزگشایی کنیم یعنی اگر در محدوده نقطه مرکزی قرار داشته باشیم در بدترین شرایط حداکثر با اعمال ۵ رمزگشایی به کلید صحیح دسترسی خواهیم یافت و پیام رمزگشایی خواهد شد. نحوه محاسبه این ۵ نقطه به شرح ذیل می‌باشد:

$$¶ lat_{k0} = \left\lfloor \frac{lat_{s0}}{\Delta D} \right\rfloor , long_{k0} = \left\lfloor \frac{long_{s0}}{\Delta D} \right\rfloor .$$

$$¶ a \left| \begin{matrix} long_{k0}+1 \\ lat_{k0} \end{matrix} , b \right| \begin{matrix} long_{k0} \\ lat_{k0}+1 \end{matrix} , c \left| \begin{matrix} long_{k0}-1 \\ lat_{k0} \end{matrix} , d \right| \begin{matrix} long_{k0} \\ lat_{k0}-1 \end{matrix} .$$

اگر به‌جز طول و عرض جغرافیایی عوامل محدودکننده‌ی دیگری مانند زمـان و محـدوده زمانی رمزگشایی پیام نیز مورد نظر باشد باید عملیاتی مشابه برای نرمـالیزه کـردن پارامترهـا و آماده سازی برای استفاده به عنوان پارامتر تولید کلید انجام پذیرد. مثلاً اگـر بخـواهیم پیـام در محدوده زمان (تاریخ و ساعت) خاصی قابل رمزگشایی باشد باید با اسـتفاده از بـازه‌ی زمـانی پارامترهای ساعت و تاریخ را نرمالیزه کرد و سپس به عنوان پارامتر تولید کلید استفاده نمود.

تولید کلید

اگر طول و عرض جغرافیایی به تنهایی به عنوان پارامترهای تولید کلید رمزنگاری انتخاب شوند تعداد کلیدهای ممکن به اندازه کل مساحت زمین یعنی $5.11 * 10^{14} m^2$ خواهـد بـود. اگـر بـر اسـاس محدوده مساحت محدوده مکانی احتمال شکستن کلید را محاسبه کنیم این مقدار برابر خواهد بود با

$$P = \frac{\pi * \Delta D^2}{5.11 * 10^{14}}. \qquad (٤$$

مثلاً اگر $\Delta D = 25m$ فرض شود احتمال شکستن کلید برابر است با :

$$P = \frac{\pi * 25^2}{5.11 * 10^{14}} = \frac{1}{2.6 * 10^{11}}. \qquad (٥$$

بدتر از همه این‌که اگر بخواهیم رمزنگاری را در محیط واقعی انجام دهیم این احتمال بیشـتر هم خواهد شد چرا که حدود ۸۰٪ مردم بر روی ۳٪ خاک زمین زندگی می‌کنند بنابراین :

$$P = \frac{\pi * 25^2}{5.11 * 10^{14} * 0.03} = \frac{1}{7.8 * 10^{9}} \qquad (٦$$

که نشانگر این است که کلید خیلی مستحکمی نیست. از اینرو نیاز بـه یـک کلیـد اضـافی است که در هر نشست به صورت اتفاقی تولید شده و به همراه بقیه پارامترها در فرآیند تولیـد کلید نهایی شرکت کند.

تولید کلید نشست

یکی از روش‌های تولید کلید نشست، تولید یک کلید تصادفی (Kc) در سمت رمز کننده پیـام و ارسال به صورت امن به سمت گیرنده پیام می‌باشد. با توجه به فقدان کانال امـن جهـت انتقـال

کلید نشست، اغلب از الگوریتم‌های کلید نامتقارن بـرای حفـظ محرمـانگی کلیـد مـذکور اسـتفاده می‌شود(Liao & Chao, 2008) . این مدل که اغلب به عنوان مدل هیبریـد(Scott & Denning, 2003) از آن یاد می‌شود برای رمزنگاری پیام از الگوریتم رمزنگاری متقارن و برای انتقال کلیـد از رمزنگـاری نامتقارن استفاده می‌کند. اما دو اشکال عمده بر این مدل وجود دارد که عبارتند از:

- سرعت نسبتاً پایین در مقایسه با الگوریتم‌های رمزنگاری کلید متقارن و همچنین نیاز به توان محاسباتی بالا در برنامه‌های کـاربردی کـه از ایـن نـوع رمزنگـاری اسـتفاده می‌کنند (Lisonik & Drahansky, 2008).
- مشکل مدیریت کلید خصوصی. در عمل اکثر حملات انجام شده بـر روی سیسـتم‌هـای رمزنگاری با کلید عمومی، به جای تمرکز بر روی الگوریتم‌هـای رمزنگـاری ، بـا هـدف نفوذ به سیستم مدیریت کلید انجام می‌شود (RSA Laboratories, 2000) . کلید خصوصـی باید در سمت دارنده کلید مثلاً در گوشی تلفن همراه ذخیره شود و هیچ تضمینی وجـود ندارد که ذخیره کلید بر روی دستگاه موبایل امن باشد؛ چرا که ممکن اسـت بـه راحتـی دزدیده شده یا از طریق بلوتوث هک شود. کلید خصوصی در دستگاه همراه می‌توانـد از طریق تکنیک ذخیره سازی فایل در JAR [1] و یا ذخیره سازی رکورد[2] در RMS [3] نگهداری شود (Rice & Zhu, 2009) . ذخیره سـازی فایـل در JAR اشـاره بـه ذخیـره سـازی کلیـد خصوصی در فایل برنامه کاربردی JAR در کنار فایل‌های کـلاس بسـته برنامـه کـاربردی دارد. ذخیره سازی رکورد در RMS اشاره به استفاده از زیر سیستم MIDP [4] در اسـتاندارد J2ME [5] دارد (Giguere, 2004) . متأسفانه ابزارهای مجانی زیادی وجود دارند که قادر بـه استخراج کد منبع[6] فایل‌های JAR و ویرایش فایل‌های کلاس می‌باشـند؛ ضـمناً ابزارهـای ویرایش HEX بسیاری وجود دارند که قادر بـه اسـتخراج کلیـد از فایـل JAR مـی‌باشـند.

1 – Java ARchive
2 – Record Stored
3 – Record Management System
4 – Mobile Information Device Profile
5 – Java 2 Mobile Edition
6 – Decompile

به‌علاوه داده‌های دودویی به راحتی از RMS قابل استخراج می‌باشند. برای تحقق امنیت بیشتر می‌توان از امکانات MIDP 2.0 استفاده کرد که به دستگاه تلفن همراه اجازه تصدیق امضای دیجیتال نصب شده بر روی فایل‌های JAR را می‌دهد. در این روش گواهی کپسوله شده با فایل JAR قادر به تصدیق محتویات فایل بوده و از این طریق می‌توان از حملات مرد میانی[1] جلوگیری به عمل آورد(Lo et al., 2008).

- از این رو، برای اضافه کردن فاکتور ثانویه‌ای برای تصدیق هویت و مقابله با لو رفتن کلید خصوصی در هنگام دزدیده شدن گوشی می‌توان از یک رمز (PIN) استفاده کرد. Pin رمزی است که در صورت مراجعه فیزیکی کاربر به رمز کننده پیام مبتنی بر موقعیت، برای نصب برنامه کاربردی رمزگشا انتخاب کرده و فقط خود او و رمز کننده پیام از آن اطلاع دارند و به هیچ عنوان بر روی دستگاه گوشی همراه کاربر ذخیره نمی‌شود. در صورتیکه مراجعه فیزیکی در کار نباشد قبل از ارسال اولین پیام رمز شده مبتنی بر موقعیت، از طریق یک کانال امن مانند ارسال پستی و یا مکالمه تلفنی این رمز بین طرفین تبادل می‌شود. اما همان‌طور که گفته شد این رمز بر روی دستگاه گوشی همراه کاربر ذخیره نمی‌شود و باید به ذهن کاربر سپرده شود. از این‌رو، نمی‌تواند از پیچیدگی زیاد و یا طول زیاد برخوردار باشد و این چالشی در خصوص استفاده از *Pin* به عنوان کلید رمزگذاری می‌باشد.
- در مدل پیشنهادی در این نوشتار برای غلبه بر این مشکل از نسخه‌ای تغییر یافته از روش SPEKE[2] که توسط جابلون[3] ارائه شد(Jablon, 1996)، استفاده شده است. این روش برای تصدیق هویت و برپایی کلید نشست از طریق یک کانال نا امن با استفاده از یک

1 – Man-in-the-middle
2 – Simple password exponential key exchange method
3 – D.P. Jablon

رمز کوتاه و بدون نگرانی از ریسک حمله مبتنی بر واژه نامه[1] بنا نهاده شده است. پایه و اساس این روش همان مدل تبادل کلید دفی هلمن[2] است با دو تفاوت :

- الگوریتم دفی هلمن به خودی خود تصدیق هویت را تأمین نمی‌کند از اینرو نسبت به حمله مردمیانی آسیب پذیر است اما مدل SPEKE به صورت انتخابی قادر به انجام این کار می‌باشد.
- در SPEKE بجای استفاده از مقدار پایه ثابت (g) دفی هلمن ، از یک تابع (*f*) که مقدار رمز (Pin) را به یک پایه برای توان رسانی آماده می‌کند، استفاده می‌شود.

همان‌طور که تشریح شد اگر مرحله تصدیق هویت را از SPEKE حذف کنیم مرحله اول تولید کلید تقریباً شبیه الگوریتم دفی هلمن می‌باشد. پارامترهای تولید کلید نشست در این الگوریتم به شرح ذیل است :

جدول شماره‌ی ۲ – پارامترهای الگوریتم دفی هلمن

pin	رمز توافق شده اولیه بین طرفین
p	یک عدد اول بزرگ
f(pin)	تابعی که Pin را به مقدار مناسب پایه دفی هلمن تبدیل می‌کند
R_A,R_B	اعداد تصادفی انتخاب شده توسط طرفین
Q_A,Q_B	مقادیر نمایی محاسبه و ارسال شده توسط طرفین
H(m)	تابع درهم ساز (Hash)
K_C	کلید نشست تولید شده

همان‌طور که در شکل شماره‌ی (۲) نشان داده شده است در طرح ساده این الگوریتم (بدون تصدیق هویت)، فقط به تبادل دو پیام برای تولید کلید نشست نیاز است. با انتخاب تابع مناسب *f* مثلاً به شکل زیر می‌توان از حمله مرد میانی به شکل موثری جلوگیری کرد (Hallsteinsen et al., 2007):

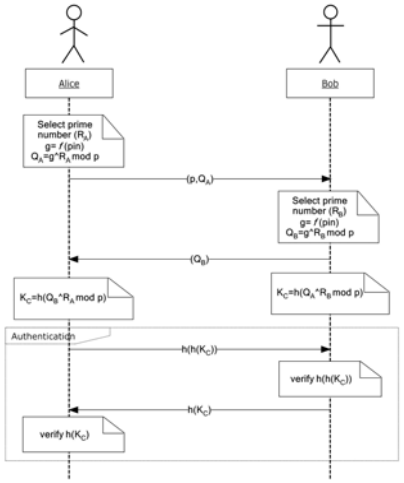
$$g = f(pin) = hash(pin)^2 \quad (۷)$$

1 – Dictionary attack
2 – Diffie-Hellman

ترکیب و درهم سازی

پس از آماده شدن پارامترهای تولید کلید و خروج از مرحله قبل، کلیه پارامترهای مولد کلید شامل مقادیر نرمالیزه شده طول/عرض جغرافیایی، محدوده فاصله (ΔD و کلید نشست (K_C) به یکدیگر الحاق شده و نتیجه آن به یک تابع درهم ساز سبک داده می‌شود تا علاوه بر تولید کلید با طول یکسان، استخراج مختصات مکانی از کلید لو رفته امکان پذیر نباشد.

در سال‌های اخیر، به دنبال پیشرفت‌های حاصل شده در تجزیه و تحلیل توابع چکیده ساز، نیاز به توابعی جدید و امن‌تر افزایش یافته است. یکی از نتایج این نیاز مسابقه SHA-3 می‌باشد که توسط NIST [1] بنیان گذاری شده است. ٦٤ پیشنهاد در مسابقه به ثبت رسید که از بین آن‌ها ۵۱ عدد برای ارزیابی در دور اول مسابقه پذیرش شدند. بعد از حدود یک سال لیست کاندیدها در مرحله دوم به ١٤ عدد کاهش یافت و در دسامبر ۲۰۱۰، ۵ عدد[2] از این توابع به عنوان فینالیست انتخاب شدند.



شکل شماره‌ی ۲ – Sequence Diagram **تولید کلید نشست در** SPEKE

1 – National Institutes for Standards and Technology
2 – Blake,Grostl, JH,Keccak and Skein

در میان فینالیست‌ها، خانواده توابع چکیده ساز BLAKE یکی از توابعی است که امید به انتخاب دارد. در طراحی خانواده این تابع از طراحی ساده ARX استفاده شده یعنی از ترکیبی از XOR ها، چرخش‌های بیتی با مقادیر ثابت و جمع پیمانه‌ای $(\mathrm{mod}\ 2^n)$ محاسباتی ساده می‌باشند، استفاده شده است(Dunkelman & Khovratovich, 2011) .

BLAKE خانواده‌ای از چهار تابع چکیده‌ساز می‌باشد : BLAKE-224, BLAKE-256, BLAKE-384 و BLAKE-512. جدول شماره‌ی (۲) مشخصات کلی این توابع را نمایش می‌دهد. در مرحله دوم مسابقه SHA-2 این خانواده شامل دو مدل ۳۲ بیتی (BLAKE-256) و ٦٤ بیتی (BLAKE-512) بود که تفاوت آنها در مقادیر اولیه، گوناگونی دنباله زنی[1] و برش خروجی[2] می‌باشد.

جدول شماره‌ی ۲ – ویژگی‌های توابع چکیده ساز BLAKE (اندازه‌ها به بیت)(Aumasson et al., 2010)

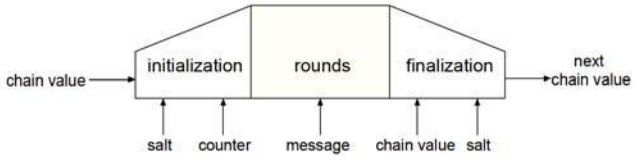
Algorithm	Word	Message	Block	Digest	Salt
BLAKE-224	32	$<2^{64}$	512	224	128
BLAKE-256	32	$<2^{64}$	512	256	128
BLAKE-384	64	$<2^{128}$	1024	384	256
BLAKE-512	64	$<2^{128}$	1024	512	256

تابع چکیده‌ساز BLAKE از روش تکرار[3] به‌کار گرفته شده در الگوریتم HAIFA تبعیت می‌کند (Biham & Dunkelman, 2006): برای فشرده سازی هر بلاک پیام با یک تابع مشخص، تابع چکیده ساز وابسته است به نمک[4] و یک شمارنده، که مشخص کننده تعداد بیت‌های چکیده شده در هر مرحله می‌باشد.

ساختار تابع فشرده ساز BLAKE از تابع LAKE به ارث برده شده است و همان‌طور که در جدول شماره‌ی (۲) نمایش داده شده است یک وضعیت داخلی بزرگ از مقدار اولیه، نمک و شمارنده، مقداردهی اولیه می‌شود. سپس این مقدار در هر مرحله (راند[5]) با مقدار پیام ترکیب

1 - Padding
2 - Truncated output
3 - Iteration mode
4 - Salt: A value that parametrizes the function, and can be either public or secret.
5 - Round

می‌شود و در آخـر فشرده شده تا به زنجیـره بعـدی وارد شـود. ایـن راهبـرد local wide-pipe نامیده می‌شود (Aumasson et al., 2010).



شکل شماره‌ی ۳ – ساختار local wide-pipe در تابع فشرده ساز BLAKE (Aumasson et al., 2010)

یک دور از تابع BLAKE-256 یک جفت راند تغییر یافته از رمز دنباله‌ای ChaCha [۱] کـه خـود گونه‌ای از رمزنگاری SalSa است، می‌باشد. این تابع دارای مزیت‌هایی به شرح ذیل است :

- طراحی:
  - سادگی الگوریتم؛
  - انجام عمل چکیده سازی با نمک .
- کارایی:
  - سرعت عمل هم در پیاده سازی نرم‌افزاری و هم سخت‌افزاری؛
  - قابلیت پـردازش مـوازی و سـبک سـنگینی بـین حافظـه و تـوان عملیـاتی در پیاده‌سازی سخت‌افزاری؛
  - سادگی سبک سنگینی بین سرعت و محرمانگی از طریق تعداد دورهای قابل تنظیم؛
- امنیت:
  - مبتنی بر اجزای تجزیه و تحلیل شده پرقدرت (ChaCha)؛
  - مقاوم در مقابل حملات عمومی پیش تصویر دوم؛
  - مقاوم در مقابل حملات کانال جانبی.

۱ – گونه‌ای از الگوریتم رمزنگاری SalSa که برای تحقق پراکنش سریع‌تر (Faster Diffusion) طراحی و ارائه شد (Bernstein, 2008).

با توجه به مزایای مذکور به‌ویژه سادگی پیاده سازی و سرعت اجرای الگوریتم، تابع BLAKE-256 به عنوان تابع چکیده ساز در مدل پیشنهادی در نظر گرفته شده است. همان‌طور که در ابتدای بخش نیز تشریح شد، این تابع یک چکیده پیام ۲۵٦ بیتی را تولید می‌کند. نتایج حاصل از پیاده سازی این الگوریتم بر روی دستگاه‌های همراه مختلف گویای کارایی این الگوریتم بوده که در بخش نتایج به تفسیر مورد بررسی قرار خواهد گرفت.

ماژول رمزنگاری/ رمزگشایی

الگوریتم رمزنگاری در مدل پیشنهادی باید به گونه‌ای باشد که با توجه به محدودیت منابع (پردازنده، حافظه و انرژی) بالاترین کارایی را ارائه کند. رمزنگاری دنباله‌ای به عنوان یکی از روش‌های رمزنگاری کلید متقارن همواره به دلیل کارایی در پیاده‌سازی نرم‌افزاری و سرعت اجرا عملیات مشهور بوده است. این نوع رمزنگاری از ابتدای شروع پروژه eSTREAM در سال ۲۰۰٤ بسیار مورد توجه قرار گرفتند. در ۱۵ آوریل ۲۰۰۸ مسابقات eSTREAM به پایان رسید و بر اساس گزارش نهایی (S. Babbage, C. Cannière, A. Canteaut, C. Cid, H. Gilbert, T. Johansson, M. Parker, B. Preneel, V. Rijmen, and M. Robshaw, 2008)، HC-128 (Wu, 2005)، Rabbit (Boesgaard et al., 2005)، Salsa20/12 (Bernstein, 2005)، SOSEMANUK (Berbain et al., 2005) به عنوان لیست نهایی پروژه‌ی مذکور اعلام شدند.

Salasa20 یک رمز دنباله‌ای است که توسط دانیل برنشتاین[1] در سال ۲۰۰۵ به عنوان یکی از کاندیدهای پروژه eSTREAM مطرح شد. برنشتاین در ضمن انواع ۸ و ۱۲ دوری از این الگوریتم به نام‌های Salsa20/12, Salsa20/8 را برای ارزیابی عمومی به ثبت رسانید (Bernstein, 2005)، اگرچه آن‌ها به عنوان کاندیدهای رسمی در پروژه‌ی eSTREAM نبودند.

رمز دنباله‌ای Salsa20 بر روی کلمات ۳۲ بیتی[2] عمل می‌کند، به‌طوری‌که به عنوان ورودی یک کلید ۲۵٦ بیتی $k = (k_0, k_1, \ldots, k_7)$, و یک مقدار ٦٤ بیتی به عنوان عدد یکبار

1 – Daniel Bernstein
2 – 32-bit words

مصرف[1] به‌صورت $v = (v_0, v_1)$ می‌باشد و یک‌سری از کلیدهای بلاک ۵۱۲ بیتی تولید می‌کند. i_{th}- بلاک خروجی تابع Salsa20 می‌باشد که به عنوان کلید، عدد تک‌شمار(یک‌بار مصرف) و یک شمارنده ٦٤ بیتی $t = (t_0, t_1)$ طابق با عدد صحیح i می‌باشد. این تابع بر روی یک ماتریس 4×4 کلمات ۳۲ بیتی که به شکل زیر نوشته شده‌اند عمل می‌نماید:

$$x = \begin{pmatrix} x_0 & x_1 & x_2 & x_3 \\ x_4 & x_5 & x_6 & x_7 \\ x_8 & x_9 & x_{10} & x_{11} \\ x_{12} & x_{13} & x_{14} & x_{15} \end{pmatrix} = \begin{pmatrix} c_0 & k_0 & k_1 & k_2 \\ k_3 & c_1 & v_0 & v_1 \\ t_0 & t_1 & c_2 & k_4 \\ k_5 & k_6 & k_7 & c_3 \end{pmatrix} \quad (۸)$$

مقادیر c_i، مقادیر ثابتی مطابق جدول (۲) می‌باشند. موارد تشریح شده در بالا همگی در مورد کلید با طول ۲٥٦ بیت می‌باشند. اگر کلید k، ۱۲۸ بیتی باشد بیت‌های کلید ۲٥٦ بیتی در ماتریس با مقدار $k = k||k'$ خواهند شد.

جدول شماره‌ی ۳ – مقادیر ثابت SalSa20

	Round	F_0	F_1	F_2	F_3
C_0	61707865	73726966	6f636573	72696874	72756f66
C_1	3320646E	6d755274	7552646e	6d755264	75526874
C_2	79622D32	30326162	3261626d	30326162	3261626d
C_3	6B206574	636f6c62	6f6c6230	636f6c62	6f6c6230

اگر طول کلید ذکر نشود ۲٥٦ بیت به عنوان طول کلید در نظر گرفته می‌شود. بلاک از جریانی از کلیدها Z به صورت زیر تعریف می‌شود:

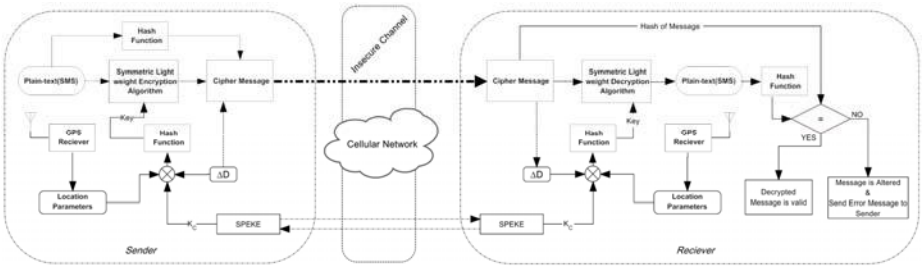
$$Z = X + X^{20} \quad (۹)$$

در این رابطه علامت + نمایانگر جمع عدد صحیح به صورت کلمه‌ای (۸ بیتی) $X^r = Round^r(X)$، در آن تابع Round بر اساس تکرار الگوریتم Salsa می‌باشد. تابع Round بر اساس روابط غیرخطی زیر که Quarterround function نیز نامیده می‌شوند به‌دست می‌آید. این تابع از انتقال بردار (X_0, X_1, X_2, X_3) به (Z_0, Z_1, Z_2, Z_3) طریق محاسبات پی‌درپی حاصل می‌شود.

$$z_1 = x_1 \oplus [(x_3 + x_0) <<< 7]$$

1 – Nonce

$$z_2 = x_2 \oplus [(x_0 + z_1) <<< 9]$$
$$z_3 = x_3 \oplus [(z_1 + z_2) <<< 13$$
$$z_0 = x_0 \oplus [(z_2 + z_3) <<< 18$$



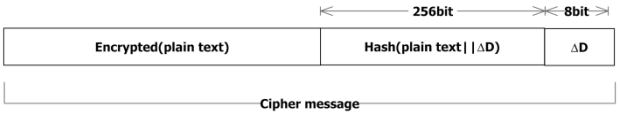
ماژول پیام کوتاه رمز شده

پس از رمزگشایی پیام در سمت فرستنده، به منظور تحقق تمامیت و جامعیت پیام[1]، چکیده پیام با استفاده از همان تابع درهم ساز استفاده شده برای تولید کلید، محاسبه شده و به انتهای پیام رمز شده الحاق می‌گردد. علاوه بر این، چون در سمت گیرنده پیام برای تولید کلید مناسب رمزگشایی داشتن مقدار ΔD ضروری است این مقدار نیز بدون رمزنگاری به پیام ارسالی الصاق می‌گردد. با در نظر گرفتن ۸ بیت برای این مقدار، قادر به تعیین محدوده رمزگشایی بین ۱ تا ۲۵۶ متر خواهیم بود. اگر محدوده بیشتری مورد نظر باشد باید ۱۶ بیت برای آن در نظر گرفته شود. از این‌رو، ۸ بیت آخر پیام را به این مقدار و ۲۵۶ بیت بعدی را به مقدار Hash و باقی‌مانده، پیام رمز شده خواهد بود.

در سمت گیرنده با جدا سازی مقادیر انتهایی پیام دریافت شده از شبکه و استخراج کلید ΔD مناسب رمزگشایی تولید شده و پیام رمزگشایی می‌شود و در مرحله بعد برای تعیین جامعیت پیام مجدداً چکیده پیام رمزگشایی شده محاسبه شده با مقدار استخراج شده از انتهای پیام دریافتی مقایسه می‌شود. اگر مقادیر یکسان باشد پیام مورد قبول بوده و در غیر این صورت

1 – Message integrity

پیغام خطایی به فرستنده پیام ارسال می‌گردد. شکل شماره‌ی (۵) ساختار پیام کوتاه ارسالی را نمایش می‌دهد.



شکل شماره‌ی ۵ – ساختار پیام کوتاه رمز شده

پیاده سازی

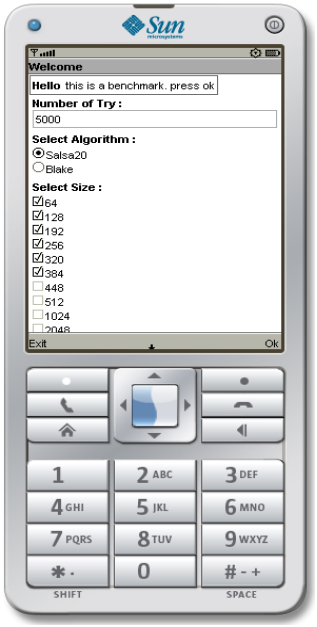
به منظور اثبات مدل پیشنهادی دو ماژول مهم رمزنگاری و درهم ریزی (چکیده سازی) که در میان ماژول‌های سیستم بیشترین مصرف کننده منابع سیستم از نظر پردازنده و حافظه می‌باشند، پیاده سازی شده و زمان اجرای الگوریتم‌های مذکور در سیستم‌های مختلف همراه با قابلیت‌های پردازشی و حافظه کم، متوسط و بالا مورد آزمایش و محک قرار گرفته است.

در طراحی و پیاده‌سازی برنامه‌ی تست کارایی الگوریتم‌های رمزنگاری/ رمزگشایی و همچنین الگوریتم درهم‌سازی از سه کلاس جاوا به شرح ذیل استفاده شده است:

- Benchmark: حاوی رابط کاربری به منظور انجام تست‌های مورد نظر بر روی الگوریتم‌ها با قابلیت انتخاب طول پیام و همچنین محاسبه زمان اجرای الگوریتم بر روی انواع دستگاه‌های همراه میزبان.
- Salsa: حاوی توابع رمزنگاری/ رمزگشایی الگوریتم Salsa20 با طول کلید ۲۵٦ بیتی می‌باشد.
- Blake: حاوی تابع چکیده ساز BLAKE-256 بوده که ورودی آن پیام با طول دلخواه و خروجی آن چکیده پیام به طول ۲۵٦ بیت می‌باشد.

رابط کاربری تهیه شده امکان اجرای تعداد دفعات تست را برای هر یک از طول‌های پیام مختلف فراهم می‌آورد. مقدار پیش فرض قرار داده شده در برنامه، ۱۰ می‌باشد اما با توجه به تست‌های مکرر به عمل آمده مشخص شد که به دلیل عدم دسترسی MIDlet به دستوراتی به منظور بالا بردن اولویت اجرای برنامه‌ی تست و به

دست گیری کامل توان پردازشی دستگاه همراه، اگر تعداد تست‌های اجرا شده پایین باشد به مراتب نتایج حاصله خطای بیشتری دارند. این مورد در دستگاه‌های همراه دارای سیستم‌های عامل با قابلیت اجرای چند برنامه به طور هم‌زمان بیشتر دیده می‌شود؛ به‌طوری‌که اگر در زمان اجرای تست وظایف دیگری نیز بر دوش پردازنده باشد زمان‌های بدست آمده بسیار بالا و دور از ذهن خواهد بود



شکل شماره‌ی ٦ - محیط واسط کاربری برنامه تولید شده برای تست کارایی الگوریتم‌های مدل

در ضمن برنامه کاربردی تهیه شده امکان انتخاب تست بر روی الگوریتم رمزنگاری و الگوریتم درهم سازی را در یک رابط کاربری ارائه می‌کند. طول پیام‌های کوتاه مورد آزمایش به صورت انتخابی بوده و کاربر می‌تواند آزمایش را بر روی پیام‌های کوتاه از ٦٤ بایت تا ١٢٨٠٠ بایت انجام دهد.

نرم‌افزار تهیه شده به منظور تست کارایی و سرعت الگوریتم‌های رمزنگاری و درهم سازی، بر روی ١٥ دستگاه گوشی همراه با ویژگی‌های سخت‌افزاری و سیستم

عامل مختلف تست گردید. دستگاه‌های انتخاب شده طیف وسیعی از دستگاه‌های همراه را شامل می‌شدند از دستگاه‌های گران قیمت با پردازنده پیشرفته و حافظه بالا، تا نمونه‌های بسیار ارزان قیمت و با حداقل امکانات سخت افزاری.

اما در همه آن‌ها مشترک بود، قابلیت اجرای برنامه‌های Java بر روی این دستگاه‌ها بود که شرط اولیه برای استفاده کاربر از سیستم بانکداری مبتنی بر پیام کوتاه است.

نتایج نشان داد که عملیات رمزنگاری/ رمزگشایی و درهم سازی که بیشترین حافظه و زمان پردازش را به خود مشغول می‌کنند در زمان‌هایی بسیار کمتر از ثانیه قابل اجرا می‌باشند. ضمناً دستگاه‌های با حداقل حافظه نیز توانستند تست انجام شده را اجرا کرده و این عملیات را انجام دهند.

این نتایج، گویای انتخاب صحیح الگوریتم‌های رمزنگاری و درهم سازی سبک می‌باشد. اکثر مدل‌های رمزنگاری پیام کوتاه از الگوریتم‌های مانند AES و SHA-1 استفاده می‌کنند که کد برنامه آن‌ها در J2ME در اینترنت یافت می‌شود. اما کدهای استفاده شده در الگوریتم SalSa20 و BLAKE-256 برای اجرا در محیط مذکور بهینه سازی شده و حتی در برخی موارد مجدداً کد نویسی شده است.

جدول شماره‌ی ٤ – دستگاه‌های همراه مورد تست را به همراه مشخصات آن‌ها نشان می‌دهد.

Row(ID)	Brand	Model	Memory[1]	Os	Announce
1	Nokia	N73	64MB	Symbian OS 9.1, S60 3rd edition	2006
2	Sony Ericsson	C905	160MB	Java MIDP 2.0	2008
3	Sony Ericsson	K800	64MB	Java MIDP 2.0	2006
4	Nokia	6670	8MB	Symbian OS v7.0s, Series 60 v2.0	2004
5	Nokia	6710N	50MB	Symbian OS 9.3, S60 rel. 3.2	2009
6	HTC	Touch cruise	128MB	Microsoft Windows Mobile 6.0 Professional	2007

1 – Internal memory-RAM

7	Sony Ericsson	W350	14MB	Java MIDP 2.0	2008
8	Sony Ericsson	W810	20MB	Java MIDP 2.0	2006
9	Nokia	2710N	64MB	Java MIDP 2.1	2009
10	Sony Ericsson	Vivaz	75MB	Symbian Series 60, 5th edition	2010
11	Sony Ericsson	Z610i	16MB	Java MIDP 2.0	2006
12	Samsung	8910	150MB	Java MIDP 2.0	2009
13	Sony Ericsson	Aino	55MB	Java MIDP 2.0	2009
14	Nokia	N70	22MB	Symbian OS 8.1a , Series 60 UI	2005
15	Nokia	N76	96MB	Symbian OS 9.2, S60 rel. 3.1	2007

جدول شماره‌ی ۵ – نتایج حاصل از تست الگوریتم **SalSa20** بر روی دستگاه‌های همراه

Headset ID	64B	128B	196B	256B	320B	384B	Average
1	142.6	187.6	239.2	277	318	373.6	219.8571
2	207.8	230.5	423.8	554.8	668.6	802.6	412.8714
3	144	221.4	353.2	483.8	812	852.6	410
4	362.2	429.8	594.4	715.6	887.8	1309.2	614.7143
5	67.2	92.2	119.2	149.4	177	214.4	117.7714
6	88.2	136	179.4	242.2	280	299.2	175.8571
7	274.2	954	972	1069.2	1146.4	1195.4	802.6
8	324.8	935	1025.4	1135	1290.2	1326	863.4857
9	445	553.8	655.6	759	850.4	948	602.9714
10	70.2	91	113.6	138.6	163.2	200.8	112.4857
11	145.6	239	375.4	619.8	728.8	756.4	410.8571
12	16.4	64.4	81.2	97.2	114.8	129.8	73.68571
13	157.6	182.6	293.6	358.4	433	528.6	280.9714
14	193.6	237	325.8	343.8	440.8	481.2	290.8857
15	86.4	114.2	134.4	177.6	207.6	236.6	138.8286

جدول شماره‌ی ٦ – نتایج حاصل از تست الگوریتم BLAKE-256 بر روی دستگاه‌های همراه

Headset ID	64B	128B	196B	256B	320B	384B	Average
1	764.6	1198	3279.8	3895.8	4371	4754.2	3043.9
2	901.4	1273	2094.2	2507	2855.2	3244.2	2145.833
3	1524.8	1817.4	2165.2	2431.6	2766.4	3076.2	2296.933
4	1339.8	2136.4	4616	5319.2	5956.4	6522.6	4315.067
5	272.6	276.4	357.2	444.2	523.6	612.8	414.4667
6	437.6	606.2	777.4	970	1105.4	1280.4	862.8333
7	2126.8	2646.6	3134.4	3631	4069.2	4627.4	3372.567
8	2376	2985.2	3577.2	4281	4965	5665.4	3974.967
9	1022.2	1507.6	1989.6	2474.2	2964.4	3449.2	2234.533
10	174.6	236	307.6	380	448.2	531.2	346.2667
11	1352	1635.8	1976	2301	2669.6	2994.6	2154.833
12	237.4	342	447.2	554.8	660.4	767.6	501.5667
13	844.6	1325.8	1758.8	2211.6	2595	2930.6	1944.4
14	1000.2	1421.8	1696.4	2163	2535	3002.4	1969.8
15	495	732.2	969.6	1491.6	4390.8	4834	2152.2

نتایج حاصل از تست الگوریتم‌ها (SalSa, BLAKE) بر روی دستگاه‌های جدول شماره‌ی (٤)، در جداول شماره‌ی (٥) و (٦) نمایش داده شده است. مقادیر نشانگر زمان متوسط اجرای عملیات بوده و برای راحتی نمایش به میکرو ثانیه تبدیل شده‌اند. ضمناً تعداد دور تست برابر با ٥٠٠٠ در نظر گرفته شده است.

همان‌طور که از نتایج آزمایش مشخص است الگوریتم SalSa20 که به منظور الگوریتم رمزنگاری و رمزگشایی در مدل پیشنهادی مطرح شده است کارایی بسیار مطلوبی از خود نمایش می‌دهد. حتی در دستگاه‌های قدیمی با حداقل حافظه و توان پردازشی (ردیف ٤)، در زمان حدود ١.٣ میلی ثانیه قادر به رمزگذاری یک پیام به طول ٣٨٤ کاراکتر می‌باشد.

این زمان در مورد یک دستگاه همراه جدید با توان پردازشی مناسب و حافظه بالا، به حدود ٠.١٢٩ میلی ثانیه می‌رسد که عملاً می‌توان آن‌را به عنوان بلادرنگ فرض نمود.

مقدار متوسط زمان رمزگذاری این الگوریتم نشانگر اجرای عملیات رمزگذاری در کمتر از یک میلی ثانیه به طور متوسط است.

در خصوص الگوریتم BLAKE-256 نیز می‌توان به نتایجی مشابه دست یافت. هرچند زمان اجرای عملیات این الگوریتم بالاتر است اما در بدترین شرایط این زمان از ٦.٥ میلی ثانیه تجاوز نمی‌کند.

نتایج حاصل از هر دو الگوریتم گویای کارایی بالای هر دو در دستگاه‌های همراه با حداقل امکانات می‌باشد. هرچند امروزه با پیشرفت فناوری در حوزه حافظه‌ها و پردازنده‌ها محدودیتی در این خصوص احساس نمی‌شود اما هدف از اجرای این تست اثبات کارایی و سازگاری الگوریتم‌های انتخابی در مدل پیشنهادی بوده است که نتایج بدست آمده به خوبی آن‌را اثبات می‌کنند.

نتیجه

تکنیک‌های رمزنگاری معمول قادر به محدودسازی کاربران ادوات همراه به رمزگشایی داده‌ها در محدوده مکانی مشخصی نیستند. برای تحقق این امر در این مقاله مدلی ارائه شده است که با استفاده از پارامترهای مختصات مکانی

قادر به محدودسازی رمزگشایی پیام در منطقه خاص عملیاتی خواهیم بود. پروتکل پیشنهادی با به‌کارگیری الگوریتم‌های سبک از نوع رمز دنباله‌ای و توابع درهم ساز سبک با حداقل منابع، محرمانگی، تمامیت و جامعیت داده و تصدیق هویت و عدم انکار را فراهم می‌آورد. هرچند مدل بیانگر رمزنگاری پیام کوتاه مبتنی بر موقعیت است اما طراحی پروتکل به گونه‌ای است که قادر به رمزنگاری انواع مختلف داده اعم از فایل داده، صدا و تصویر قابل استفاده در مناطق عملیاتی می‌باشد.

نوآوری‌های این مقاله را می‌توان به شرح ذیل بیان نمود:

- بررسی فعالیت‌های انجام شده در این حوزه تحقیقاتی و تبیین اهمیت موضوع و کاربردهای آن علی‌الخصوص در مسائل نظامی و دفاعی.
- ارائه مدل امن رمزگذاری.
- طراحی و ترسیم نمودار توالی عملیات جهت سهولت درک فرآیند عملیات تولید کلید.
- ارائه الگویی جدید جهت تبادل کلید در محیط ناامن مبتنی بر الگوریتمی تغییر یافته از الگوریتم دفی هلمن.
- استفاده از الگوریتم‌های جدید و سبک رمزنگاری و درهم‌سازی.

- اثبات سازگاری و کارایی الگوریتم‌های انتخابی بر روی دستگاه‌های مختلف همراه با استفاده از نرم افزار طراحی شده.

در ضمن، در مدل پیشنهادی فقط از مختصات جغرافیایی برای محدود سازی رمزگشایی استفاده شده اما با اعمال پارامترهای دیگر مانند سرعت، زمان و ارتفاع از سطح زمین می‌توان رمزنگاری را مستحکم‌تر نمود. این مسئله می‌تواند زمینه‌ای برای فعالیت‌های آتی باشد. ضمن اینکه پیاده سازی کل مدل نیز می‌تواند به عنوان فعالیت‌های آتی در جهت اثبات هر چه بیشتر مدل مورد نظر محققان باشد. علاوه بر موارد مذکور، پیاده سازی بقیه الگوریتم‌های رمزنگاری و درهم سازی از دو مسابقه معتبر ذکر شده خود زمینه تحقیقی بسیار مناسبی است برای انتخاب بهینه الگوریتم‌ها.

منابع

انگلیسی

1- Aumasson, J.P., Henzen, L., Meier, W. & Phan, R.C.-W., (2010). "***SHA-3 proposal BLAKE***". [Document] Available at: http://www.131002.net/blake/blake.pdf [Accessed July 2011].

2- Berbain, C. et al., (2005). "***SOSEMANUK, a fast software-oriented stream cipher***". [Online]. eSTREAM project website.

3- Bernstein, D.J., (2005). "***Salsa20/8 and Salsa20/12***". Technical Report 2006/007. ECRYPT Stream Cipher Project.

4- Bernstein, D.J., (2008). "***ChaCha, a variant of Salsa20***". [Online] eSTREAM Project Available at: http://cr.yp.to/chacha.html.

5- Biham, E. & Dunkelman, O., (2006). "***framework for iterative hash functions - HAIFA. In In Proceedings of Second NIST Cryptographic Hash Workshop***"., 2006.

6- Boesgaard, M., Vesterager, M., Christensen, T. & Zenner, E., (2005). "***The stream cipher Rabbit***". [Online]. eSTREAM project website.

7- Dunkelman, O. & Khovratovich, D., (2011). "***Iterative Differentials, Symmetries, and Message Modification in BLAKE-256. In ECRYPT II Hash Workshop***". Tallinn, Estonia, 2011.

8- Giguere, E., (2004). "**Databases and MIDP, Part 1: Understanding the Record Management System.** [Online] Available at: http://developers.sun.com/mobility/midp/articles/databaserms/.

9- Hallsteinsen, S., Jorstad, I. & Thanh, V., (2007). "***Using the mobile phone as a security token for unified authentication***". In *ICSNC '07 Proceedings of the Second*

International Conference on Systems and Networks Communications., 2007. IEEE Computer Society Washington, DC, USA.

10- Jablon, D.P., (1996). "***Strong Password-Only Authenticated Key Exchange***". *ACM SIGCOMM Computer Communication Review*, 26(5).

11- Kupper, A., (2005). "***Location-based services : fundamentals and operation***. West Sussex, England: John Wiley & Sons.

12- Liao, H.C. & Chao, Y.H., (2008). "***A New Data Encryption Algorithm Based on the Location of Mobile Users***". *Information Technology Journal*, 7(1).

13- Lisonik, D. & Drahansky, M., (2008). "***SMS Encryption for Mobile Communication***". In International Conference on Security., 2008.

14- Lo, L.C., Bishop, J. & Eloff, J.H.P., (2008). "***SMSSec: An end-to-end protocol for secure SMS***". *Computers & Security*, 27.

15- Mahmoud, Q.H., (2004). "***J2ME and Location-Based Services***". [Online] Available at: http://developers.sun.com/mobility/apis/articles/location.

16- Qiu, D., 2007. (2007) "***Security Analysis of Geoencryption: A Case Study Using Loran***". In *ION GPS/GNSS*.,.

17- Ren, K., Lou, W. & Zhang, Y., (2008). "***LEDS: Providing Location-Aware End-to-End Data Security in Wireless Sensor Networks***". *IEEE TRANSACTIONS ON MOBILE COMPUTING,* 7(5).

18- Rice, J.E. & Zhu, Y., (2009). "***A proposed architecture for secure two-party mobile payment***". In IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, PacRim'09., 2009.

19- RSA Laboratories, (2000). [Online] "***RSA Security Inc***". Available at: http://www.rsa.com/rsalabs/node.asp?id=2152.

20- S. Babbage, C. Cannière, A. Canteaut, C. Cid, H. Gilbert, T. Johansson, M. Parker, B. Preneel, V. Rijmen, and M. Robshaw, (2008). [Online] Available at: http://***www.ecrypt.eu.org***/stream/portfolio.pdf [Accessed May 2011].

21- Scott, L. & Denning, D.E., (2003). "***A Location Based Encryption Technique and Some of Its Applications***". In Proceedings of the 2003 National Technical Meeting of The Institute of Navigation. Anaheim, CA, 2003.

22- Scott, L. & Denning, L., (2003). "***Location Based Encryption & Its Role In Digital Cinema Distribution***". In *Proceedings of ION GPS/GNSS*., 2003.

23- Wu, H., (2005). "***The Stream Cipher HC-128***". [Online]. eSTREAM project website.

24- Yan, Y. & Olariu, S., (2009). "***An efficient geographic location-based security mechanism for vehicular adhoc networks***". In *IEEE 6th International Conference on Mobile Adhoc and Sensor Systems, 2009. MASS* '09. Macau, 2009