

شناخت آسیب‌های امنیتی پروتکل احراز اصالت MAP و ارائه پروتکل بهبود یافته امن

محمد مردانی شهربابک^{۱*}، شهاب عبدالملکی^۲

۱- استادیار دانشگاه جامع امام حسین (ع)، ۲- دانشجوی کارشناسی ارشد دانشگاه آزاد اسلامی واحد علوم و تحقیقات

(دریافت: ۹۵/۰۲/۰۶، پذیرش: ۹۵/۰۴/۲۷)

چکیده

امروزه سامانه‌های شناسایی با امواج رادیویی (RFID) عمدتاً در زندگی روزمره اشخاص استفاده می‌شود. این سامانه‌ها در زمینه‌هایی نظیر پزشکی، نظامی و تشخیص هواپیماهای خودی از دشمن کاربرد دارد. از حیث اهمیت امنیت این سامانه‌ها، پروتکل‌های متفاوتی برای احراز هویت پیشنهاد شده است. آقای پنگ و همکاران، یک پروتکل احراز هویت مبتنی بر استاندارد EPC C-1 G-2 ارائه دادند. طراحان آن ادعا کردند که از لحاظ امنیتی و محرمانگی امن و در مقابل حملات مقاوم است. در این مقاله نشان داده می‌شود که برخلاف ادعای طراحان، پروتکل مذکور امن نیست و در مقابل حملاتی نظیر کشف کلید، جعل برچسب، جعل کارت‌خوان و ناهم‌زمانی ضعف دارد. همچنین به‌منظور افزایش امنیت کاربران این سامانه‌ها، یک پروتکل بهبود یافته پیشنهاد داده و نشان می‌دهیم این پروتکل در مقابل حملات ذکر شده امن است. علاوه بر پیچیدگی و کارایی پروتکل‌های ارائه شده با پروتکل پیشنهادی مقایسه گردیده و نشان داده می‌شود که با تغییراتی در پروتکل، مشکلات امنیتی آن به طور کامل برطرف شده است. در نهایت، امنیت پروتکل بهبود یافته با برخی از پروتکل‌های مشابه مقایسه می‌شود.

کلید واژه‌ها: پروتکل‌های احراز هویت، سامانه‌های RFID، استاندارد EPC C1 G2، امنیت، محرمانگی، حمله‌ها.

Identification of Security Weaknesses on MAP Authentication Protocol and Presentation of Improved Protocol

M. M. Shahrabak*, S. Abdolmaleky

Imam Hossein University

(Received: 25/04/2016; Accepted: 17/07/2016)

Abstract

Nowadays, RFID technology is using widely in our daily lives. This technology is used in many applications such as healthcare, military, identifying friend or foe of fighter-aircraft and etc. In order to provide security and privacy of RFID systems, different authentication protocols have been proposed. Pang proposed a RFID authentication protocol based on EPC C-1 G-2 standard. They claimed that their protocol is secure against various attacks and provides user privacy. It is shown that protocol proposed by pang is not safe and suffer from attacks such as tag ID exposure, tag impersonation, reader impersonation and de-synchronization. Also to enhance security of RFID users, an improved version of this protocol is proposed, showing that it is secure against aforementioned attacks. In addition, complexity and efficiency of the improved protocol is compared with some similar ones and it is shown that the improved protocol is completely secure. Eventually, security of the improved protocol is compared with some existed protocols.

Keywords: Authentication Protocols, RFID Systems, EPC C-1 G-2 Standard, Security, Confidentiality, Attacks.

*Corresponding Author E-mail: Mmardani@ihu.ac.ir

۱. مقدمه

در سال های اخیر، استفاده از سامانه های «شناسایی از طریق امواج رادیویی»^۱ (RFID) رشد چشم گیری داشته است. این سامانه ها در دستگاه های کنترل دستیابی [۱]، گذرنامه های الکترونیکی [۲]، کنترل تردد [3]، دستگاه های ضد سرقت اتومبیل ها، تشخیص هوای ماه های جنگنده خودی از دشمن و خیلی از موارد دیگر مورد استفاده قرار می گیرد [۴ و ۵]. این سامانه در ارتش برای برنامه ریزی و تدارکات جنگی بصورت گسترده استفاده می شود [۲۰] و همینطور یک روش شناخته شده برای دنبال کردن و ردگیری وسایل نقلیه است [۲۱]. سامانه های RFID، قابلیت شناسایی خودکار اشیاء، با استفاده از یک تراشه ی کوچک ارزان قیمت را فراهم می کنند. این تراشه ی کوچک به اصطلاح «برچسب»^۲ یا «برچسب هوشمند»^۳ نامیده می شود. داده های ذخیره شده در این برچسب ها توسط دستگاه هایی تحت عنوان «کارت خوان»^۴ خوانده می شود. در حالت کلی سامانه های RFID از سه بخش اصلی تشکیل شده اند که شامل برچسب، کارت خوان و «سرویس دهنده ی نهایی»^۵ است. تمام اطلاعات و کلیدهای مخفی برچسب ها در سرویس دهنده ی نهایی ذخیره می شود. کارت خوان مابین برچسب و سرویس دهنده ی نهایی قرار می گیرد و وظیفه ی تبادل داده بین آن ها را بر عهده دارد.

در سال های اخیر، یکی از استانداردهای محبوب برای برچسب های فعال استاندارد «EPC C1 G2»^۶ است که توسط سازمان «EPCglobal»^۷ ارائه شده است [۶ و ۷]. استاندارد EPC C1 G2 برای تشخیص خطا در داده های مبادله شده، از عملگرهایی نظیر XOR، CRC^۸ و PRNG^۹ استفاده می کند. تلاش محققان برای طراحی پرتکلی امن و استاندارد می باشد که در این راستا یه و همکاران [۱۰]، یون و همکاران [۱۸]، خیابو و همکاران [۲۲] و همچنین علوی و همکاران [۲۳] پروتکل های احراز هویت بر روی این استاندارد را پیشنهاد کرده اند. از آنجایی که این سامانه جدید بوده و بصورت فراگیر مورد استفاده قرار می گیرد، در کشور ما نیز مورد استفاده قرار گرفته که تحقیقاتی در این راستا به نشر رسیده است [۱۹]. تا کنون پروتکل های احراز هویت زیادی مبتنی بر استاندارد EPC C1 G2 ارائه شده است [۸ - ۱۰]. با وجود این که همه ی آن ها سعی در فراهم آوردن امنیت و محرمانگی کاربران RFID بودند، ولی ایرادهایی بر همه ی آن ها وارد شده است. «یه» و همکارانش [۱۰] پروتکل احراز هویت متقابل تحت عنوان SRP را برای سامانه های RFID پیشنهاد کردند. حبیبی و همکارانش [۱۱] و

[۱۲] نشان دادند که پروتکل SRP ضعف هایی دارد. همچنین، «پنگ» و همکارانش [۱۳] «پروتکل احراز هویت متقابل» (MAP) مطابق با استاندارد EPC C1 G2 برای سامانه های RFID ارائه کردند. آن ها ادعا کردند پروتکل MAP امن است و می تواند امنیت کاربران RFID را تأمین کند.

در این مقاله نشان می دهیم برخلاف ادعای طراحان پروتکل MAP، این پروتکل همچنان امن نیست و ضعف هایی بر این پروتکل وارد است. سه حمله ی کشف کلید پروتکل، جعل هویت برچسب و جعل هویت کارت خوان بر روی این پروتکل انجام می پذیرد. در ادامه اشاره ی مختصری به مفاهیم حمله های انجام شده در این مقاله می شود.

حمله جعل: در این نوع حمله، مهاجم سعی می کند به نحوی بتواند خود را به جای یک برچسب و یا یک کارت خوان مجاز جا بزند. این حمله می تواند هم برای جعل هویت یک برچسب مجاز و هم جعل هویت یک کارت خوان مجاز استفاده شود. روش های گوناگونی وجود دارند که یک مهاجم را قادر می سازند تا روی یک پروتکل حمله جعل هویت انجام دهد. عامل کلیدی که یک مهاجم را قادر به انجام حمله جعل می کند دستیابی او به اطلاعات محرمانه مربوط به برچسب و یا کارت خوان است که این اطلاعات می تواند شامل شناسه، کلیدهای مخفی و حتی برخی از اطلاعات جزئی قابل استخراج از پیام های تبدالی باشند. در حمله جعل هویت یک برچسب مجاز، مهاجم با اطلاعاتی که از شنود ارتباطات مربوط به آن برچسب به دست آورده است، قادر می شود که به پرسمان های یک کارت خوان مجاز به درستی پاسخ دهد و خود را به عنوان یک برچسب مجاز به کارت خوان اثبات کند. این کار می تواند با ایجاد تغییراتی در پیام های تبدالی در نشست های قبلی نیز صورت گیرد. از طرف دیگر، یک مهاجم می تواند به کمک روش های ذکر شده در فوق، هویت یک کارت خوان و یا یک سرویس دهنده مجاز را نیز جعل کند و در برگزاری یک نشست با برچسب مورد نظر، پروتکل را به طور کامل اجرا کرده و به پایان برساند. این کار باعث می شود که مهاجم به اهدافی شامل دستیابی به اطلاعات محرمانه مربوط به آن برچسب و یا ناهم زمان کردن آن برچسب و سرویس دهنده باشد، دست یابد [۱۴].

حمله کشف کلید و شناسه: پروتکل های احراز هویت به کار گرفته شده در سامانه های RFID می توانند در برابر حملات گوناگون کشف کلید و شناسه آسیب پذیری داشته باشند. در این گونه حملات مهاجم سعی می کند با یافتن رخنه های امنیتی در پروتکل مورد نظر، با روش های متفاوتی اقدام به کشف کلید یا کلیدهای مخفی و یا شناسه ی برچسب قربانی کند. یکی از رایج ترین روش های کشف کلید و شناسه، روش جستجوی جامع فضای کلید است. اگر کلیدهای مخفی به کار گرفته شده در پروتکل، رشته بیت هایی با طول کم باشند آنگاه این فرصت برای مهاجم فراهم

¹Radio Frequency Identification (Rfid)

² Tag

³ Smart Tag

⁴ Reader

⁵ Back-End-Server

⁶ Epc Class 1 Generation 2 (Epc C-1 G-2)

⁷ Electronic Product Code

⁸ Cycle Redundancy Checksum Code (Crc)

⁹ Pseudo Random Number Generator (Prng)

برچسب خودداری کرده و برچسب قربانی، در عمل از سامانه حذف می‌شود [۱۰]. در حقیقت کلیدی که در کارتخوان و یا پایگاه داده ذخیره شده، با کلیدی که در برچسب وجود دارد متفاوت است به این دلیل که مهاجم ارتباط بین برچسب و کارتخوان را در یک زمان مشخصی مسدود کرده است. یک مهاجم با روش‌های مختلفی می‌تواند اقدام به حمله ناهم‌زمانی [۱۷].

در ادامه مقاله به بخش‌های زیر تقسیم شده است بطوری که در بخش ۲، پروتکل احراز هویت MAP مورد بازبینی قرار می‌گیرد. بخش ۳، تحلیل امنیتی پروتکل MAP و ضعف‌های آن مورد بررسی قرار می‌گیرد و حمله‌های کشف کلید شناسه، جعل هویت برچسب، جعل هویت کارتخوان و حمله‌ی ناهم‌زمانی بر روی این پروتکل انجام می‌گیرد. بعد از آن جهت تأمین امنیت کاربران RFID یک پروتکل بهبودیافته از MAP پیشنهاد می‌شود که در بخش ۴ شرح داده شده است. همچنین در بخش ۴، به تحلیل امنیتی پروتکل پیشنهادی پرداخته شده است و مقایسه‌ای از تحلیل امنیتی پروتکل پیشنهادی با برخی پروتکل‌های مشابه آورده شده است. در نهایت نتیجه‌گیری مقاله در بخش ۵ آورده شده است.

۲. بازبینی پروتکل MAP

پنگ و همکارانش [۱۳] پروتکل احراز هویت متقابلی مطابق با استاندارد EPC C-1 G-2 برای سامانه‌های RFID ارائه کردند. در شکل (۱) روند کار این پروتکل احراز هویت نمایش داده شده است. در پروتکل MAP کانال ارتباطی بین کارتخوان و سرویس‌دهنده‌ی نهایی امن در نظر گرفته شده است، در مقابل کانال مخابراتی بین برچسب و کارتخوان ناامن فرض شده است. در ادامه این پروتکل را به صورت مختصر بازبینی می‌کنیم. نمادهای استفاده شده در پروتکل در جدول (۱) نشان داده شده است

می‌شود تا با دستیابی به داده‌های ردیاب شده در یک نشست و سپس انجام یک جستجوی جامع روی فضای رشته کلید به کار گرفته شده، موفق به کشف آن کلید شود.

در جستجوی جامع، روش‌های دیگری نیز برای کشف کلید و شناسه وجود دارند که به کارگیری این روش‌ها از یک سو به نوع پروتکل به کار گرفته شده و از سوی دیگر به هوش و خلاقیت مهاجم در این حملات بستگی دارد [۱۵ و ۱۶].

حمله ناهم‌زمانی: پروتکل‌های احراز هویت به کار گرفته شده در سامانه‌های RFID می‌توانند در برابر حملات گوناگون کشف کلید و شناسه آسیب‌پذیر باشند. برای مقاوم بودن در برابر حملات ردیابی، بسیاری از پروتکل‌های احراز هویت از یک فرایند به‌روزرسانی در درون خود استفاده می‌کنند. این فرایند به‌روزرسانی باعث می‌شود تا از انجام یک نشست موفقیت‌آمیز بین برچسب و کارتخوان، طرفین مقادیر سری خود شامل شناسه، کلیدهای مربوطه و سایر مقادیر مخفی را به‌روزرسانی کرده و در نشست بعدی از مقادیر جدیدی استفاده کنند. انجام به‌روزرسانی مقادیر باعث می‌شود تا حتی اگر یک مهاجم به اطلاعات مخفی مربوط به یک برچسب در زمان t دسترسی پیدا کرد دیگر قادر نباشد تا تراکنش‌های بین آن برچسب و کارتخوان‌ها را در زمان‌های $t' > t$ ردیابی کند، زیرا مقادیر استفاده شده برای شناسایی برچسب در زمان‌های t و t' متفاوت خواهند بود.

بالین حال همین ویژگی باعث به وجود آمدن آسیب‌پذیری‌های دیگری برای پروتکل‌های مورد بحث می‌شود. در واقع اگر برچسب و سرویس‌دهنده به هر دلیلی نتوانند مقادیر یکسانی را به‌روزرسانی کنند این امر موجب نشود تا مقادیر مخفی ذخیره شده در برچسب و سرویس‌دهنده متفاوت باشد و در نتیجه در نشست‌های بعدی هنگامی که برچسب قصد شناسایی خود به سرویس‌دهنده را دارد، به دلیل عدم تطابق مقادیری که او در اختیار دارد با مقادیری که در سرویس‌دهنده موجود است، سرویس‌دهنده از تأیید هویت آن

Server [$C_{inew}, C_{iold}, K_{inew}, K_{iold}, EPC_S, D_i$]	Reader []	Tag [EPC_S, C_i, K_i]
Search database $C_i \stackrel{?}{=} C_{inew} \text{ OR } C_{iold}$ If OK, get EPC_S compute: $K_i = EPC_S \oplus M_1 \oplus N_1$ $N_2 = CN_2 \oplus PRNG(K_i)$ $M_2 \oplus K_i \stackrel{?}{=} CRC(EPC_S \oplus N_2 \oplus C_i)$	$N_1 \in_R \{0,1\}^l$	(1) $N_1 \rightarrow$
If OK, compute: $M_3 = CRC(EPC_S \oplus (N_2 \gg l/4)) \oplus K_i$ $C_{iold} \leftarrow C_i$ $C_{inew} \leftarrow PRNG(N_1 \oplus N_2) \oplus K_i$ $K_{iold} \leftarrow K_i$ $K_{inew} \leftarrow K_i \oplus (N_2 \gg l/4)$	(3) C_i, M_1, CN_2, M_2, N_1 \leftarrow	(2) C_i, M_1, CN_2, M_2 \leftarrow
	(4) $(D_i, M_3) \rightarrow$	(5) $M_3 \rightarrow$
		Received N_1 Generates $N_2 \in_R \{0,1\}^l$ Compute: $M_1 = EPC_S \oplus N_1 \oplus K_i$ $CN_2 = N_2 \oplus PRNG(K_i)$ $M_2 = CRC(EPC_S \oplus N_2 \oplus C_i) \oplus K_i$ $M_3 \oplus K_i \stackrel{?}{=} CRC(EPC_S \oplus (N_2 \gg l/4))$ If OK: $C_i \leftarrow PRNG(N_1 \oplus N_2) \oplus K_i$ $K_i \leftarrow K_i \oplus (N_2 \gg l/4)$

شکل ۱. پروتکل MAP [۱۹].

می‌کند. در ادامه با استفاده از این مسئله که $K_i = K_{old}$ یا $K_i = K_{new}$ ، سرویس‌دهنده نهایی مقدار $N_2 = CN_2 \oplus PRNG(K_i)$ را محاسبه کرده و سپس چک می‌کند که مقدار $M_2 \oplus K_i$ برابر $CRC(EPC_S \oplus N_2 \oplus C_i)$ است یا نه؟ این فرایند تا زمانی که برچسب اصیل پیدا شود، ادامه می‌یابد. در غیر این صورت، کارت‌خوان پیام خطا از سرویس‌دهنده نهایی دریافت می‌کند و پروتکل ساقط می‌شود.

۴) ارسال سرویس‌دهنده نهایی به کارت‌خوان

الف) بعد از احراز هویت موفق سرویس‌دهنده نهایی مقدار $M_3 = CRC(EPC_S \oplus (N_2 \gg 1/4)) \oplus K_i$ را محاسبه می‌کند و مقادیر (D_i, M_3) را به کارت‌خوان ارسال می‌کند.

ب) سپس سرویس‌دهنده نهایی مقادیر خود را به صورت زیر به روزرسانی می‌کند:

$$C_{iold} \leftarrow C_i, C_{inew} \leftarrow PENG(N_1 \oplus N_2)$$

$$K_{iold} \leftarrow K_i, K_{inew} \leftarrow K_i \oplus (N_2 \gg 1/4) \quad (1)$$

اگر آن‌ها برابر بودند، برچسب با موفقیت سرویس‌دهنده نهایی را احراز هویت می‌کند و پارامترهای زیر را به صورت زیر به روزرسانی می‌کند:

$$C_i \leftarrow PENG(N_1 \oplus N_2)$$

$$K_i \leftarrow K_i \oplus (N_2 \gg 1/4) \quad (2)$$

در غیر این صورت برچسب مرحله را متوقف می‌کند و پروتکل ساقط می‌شود.

۳. حمله به پروتکل MAP

در این بخش پروتکل MAP را مورد تحلیل امنیتی قرار می‌دهیم. در ادامه نشان می‌دهیم که این پروتکل در مقابل حملاتی نظیر کشف کلید شناسه، جعل کارت‌خوان، جعل برچسب و ناهم‌زمانی ضعف دارد.

۳-۱. حمله کشف کلید شناسه

در این بخش یک حمله کاملاً عملی را ارائه می‌کنیم که توسط این حمله شناسه برچسب قربانی فاش می‌شود. در پروتکل MAP هر برچسب یک شناسه ۱۶ بیتی EPC_S دارد که این شناسه در فرایند احراز هویت نیز نقش مهمی بازی می‌کند. به دلیل کوتاه بودن طول رشته بیت EPC_S ، طراحان پروتکل سعی کرده‌اند تا ساختار پیام‌ها را به گونه‌ای طراحی کنند که شناسه برچسب به صورت آشکار ارسال نشود و در نتیجه از خطر حمله جستجوی جامع در امان باشد. اما در ادامه نشان می‌دهیم که تدابیر آن‌ها چندان مؤثر نبوده است و می‌توان به کمک پیام‌هایی که به صورت آشکار ارسال می‌شوند شناسه برچسب را محاسبه کرد. روند انجام

جدول ۱. نمادهای استفاده شده در پروتکل MAP

نماد	توضیحات
EPC_S	یک مقدار ۱۶ بیتی که از XOR کد استاندارد EPC ایجاد شده
K_i	کلید خصوصی ذخیره‌شده در برچسب و پایگاه داده برای احراز هویت
P_i	کلید دست‌یابی ذخیره‌شده در برچسب و پایگاه داده
C_i	ایندکس ذخیره‌شده در برچسب برای جستجوی آن در پایگاه داده
RID	شناسه کارت‌خوان
P_{old}	کلید دست‌یابی قبلی ذخیره‌شده در پایگاه داده
K_{old}	کلید خصوصی قبلی ذخیره‌شده در پایگاه داده
K_{new}	کلید خصوصی جدید ذخیره‌شده در پایگاه داده
C_{old}	ایندکس قبلی پایگاه داده ذخیره‌شده در پایگاه داده
C_{new}	ایندکس جدید پایگاه داده ذخیره‌شده در پایگاه داده
D_i	ریز اطلاعات ذخیره‌شده از برچسب در پایگاه داده
h	تابع چکیده‌ساز

پروتکل MAP از دو گام اصلی تشکیل شده است که به صورت زیر قابل بیان است:

۲-۱. گام اولیه

در این گام از پروتکل که در واقع گام مقداردهی اولیه است، مقادیر اولیه پارامترهای $[K_{old}, C_{old}, K_{new}, C_{new}, EPC_S, D_i]$ که در سرویس‌دهنده نهایی بارگذاری شده‌اند، با مقادیر اولیه تصادفی K_0, P_0, C_0 که توسط تولیدکننده تولید شده‌اند، مقداردهی می‌شوند. به این ترتیب که $K_{old} = K_{new} = K_0$ و $C_{old} = C_{new} = C_0$. همچنین در این مرحله، مقادیر $[K_i, C_i, EPC_S]$ که در برچسب بارگذاری شده‌اند با مقادیر K_0 و C_0 مقداردهی می‌شوند. به این ترتیب که $K_i = K_0$ و $C_i = C_0$

۲-۲. گام احراز هویت

۱) در ابتدا کارت‌خوان عدد تصادفی N_1 را تولید می‌کند، سپس عدد N_1 را به برچسب ارسال می‌کند.

۲) در این مرحله، کارت‌خوان در ابتدا عدد تصادفی N_2 را تولید می‌کند. سپس با استفاده از عدد N_2 مقادیر $M_1 = EPC_S \oplus N_1 \oplus K_i$ ، $CN_2 = N_2 \oplus PRNG(K_i)$ ، $M_2 = CRC(EPC_S \oplus N_2 \oplus C_i) \oplus K_i$ را محاسبه کرده و به کارت‌خوان ارسال می‌کند.

۳) در این مرحله، کارت‌خوان داده‌های ارسال‌شده از برچسب را دریافت می‌کند و مقادیر (C_i, M_1, CN_2, N_2) را به سرویس‌دهنده نهایی ارسال می‌کند، و سرویس‌دهنده نهایی بعد از دریافت داده از کارت‌خوان، عملیات زیر را انجام می‌دهد: با استفاده از C_i دریافت شده، سرویس‌دهنده نهایی پایگاه داده را جهت مطابقت دادن C_i با C_{old} و C_{new} جستجو و پیدا می‌کند. سپس سرویس‌دهنده نهایی مقدار $K_i = EPC_S \oplus N_1 \oplus M_1$ را محاسبه

حمله به ترتیب زیر است.

کند. مهاجم با روش شرح داده شده در بخش قبل شناسه EPC_s برچسب قربانی را به دست می‌آورد. در نتیجه با در اختیار داشتن EPC_s ، مهاجم می‌تواند در هر زمان دلخواهی هویت برچسب قربانی را به ترتیب زیر جعل کند.

(۱) مهاجم در دور i -ام، نشست برگزار شده بین برچسب قربانی T_j و کارت‌خوان را شنود کرده و مقادیر (C_i, M_1, CN_2, M_2) را به دست می‌آورد. سپس زمانی که کارت‌خوان و برچسب نشست جدیدی در دور $(i+1)$ -ام با یکدیگر برقرار می‌کنند. مهاجم بعد از اجرای مرحله اول و دوم پروتکل، زمانی که برچسب پیام $(C_{i+1}, M'_1, CN'_2, M'_2)$ را به کارت‌خوان ارسال می‌کند پروتکل را متوقف کرده و از رسیدن پیام به کارت‌خوان جلوگیری می‌کند.
(۲) به دلیل این‌که او مقدار EPC_s را در اختیار دارد با محاسبه زیر می‌تواند مقدار K_i و K_{i+1} را به دست آورد:

$$K_i = M_1 \oplus N_1 \oplus EPC_s \quad (۸)$$

$$K_{i+1} = K_i \oplus (N_2 \gg l/4) \quad (۹)$$

(۳) حال مهاجم یک نشست جدید را با کارت‌خوان آغاز می‌کند و در پاسخ به N'_1 ارسالی از طرف کارت‌خوان، ابتدا عدد N'_2 را تولید کرده سپس پیام‌های M'_1, CN'_2 و M'_2 را محاسبه می‌کند:

$$M'_1 = EPC_s \oplus N'_1 \oplus K_{i+1} \quad (۱۰)$$

$$CN'_2 = N'_2 \oplus PRNG(K_{i+1}) \quad (۱۱)$$

$$M'_2 = CRC(EPC_s \oplus N'_2 \oplus C_{i+1}) \oplus K_{i+1} \quad (۱۲)$$

چون که هر سه پیام با مقادیر معتبر و صحیحی محاسبه شده‌اند مورد پذیرش کارت‌خوان قرار می‌گیرد و هویت مهاجم به‌عنوان برچسب قربانی مورد تأیید قرار می‌گیرد.

۳-۳. حمله جعل هویت کارت‌خوان

فاش شدن EPC_s برای مهاجم، نه‌تنها باعث آسیب‌پذیری پروتکل در برابر حمله جعل هویت برچسب می‌شود، بلکه امکان وقوع حمله جعل هویت کارت‌خوان و ناهم‌زمان کردن طرفین نیز برای مهاجم فراهم می‌شود. روند انجام این حمله به شکل زیر است.

(۱) مهاجم دور i -ام، نشست برگزار شده بین برچسب قربانی T_j و کارت‌خوان را شنود کرده و با روش شرح داده شده در بخش‌های قبل اقدام به محاسبه مقادیر EPC_s و K_{i+1} می‌کند.
(۲) مهاجم با ارسال عدد N'_1 یک نشست جدید را با برچسب قربانی T_j آغاز می‌کند. برچسب نیز با $(C_{i+1}, M'_1, CN'_2, M'_2)$ به او پاسخ می‌دهد.

(۳) پس از دریافت پاسخ برچسب، مهاجم ابتدا N'_2 را از طریق پیام CN'_2 و K_{i+1} استخراج می‌کند و پیام M'_3 را محاسبه کرده و برای برچسب ارسال می‌کند:

$$N'_2 = CN'_2 \oplus PRNG(K_{i+1}) \quad (۱۳)$$

(۱) مهاجم ابتدا اجازه می‌دهد که کارت‌خوان و برچسب در دور i -ام یک نشست موفقیت‌آمیز را انجام دهند که در این بین مقادیر $(N_1, C_i, M_1, CN_2, M_2, M_3)$ را ذخیره می‌کند (از مرحله (۲) پروتکل). سپس مهاجم یک نشست را با برچسب قربانی (برچسب انتخابی یا T_j مورد بررسی) در دور $i+1$ آغاز می‌کند. این بار خود مهاجم عدد تصادفی N_1 را برای برچسب ارسال می‌کند و برچسب نیز با پیام $(C_{i+1}, M'_1, CN'_2, M'_2)$ به او پاسخ می‌دهد. مهاجم از این پیام مقدار M'_1 را ذخیره می‌کند و این نشست را خاتمه می‌دهد. به دلیل این‌که نشست اول به اتمام نرسیده است در نتیجه برچسب نیز به‌روزرسانی نکرده است. پس پیام‌های M_1 و M'_1 این‌گونه تشکیل شده‌اند:

$$M_1 = EPC_s \oplus N_1 \oplus K_i \quad (۳)$$

$$M'_1 = EPC_s \oplus N_1 \oplus K_{i+1} \quad (۴)$$

(۲) مهاجم با XOR کردن M_1 و M'_1 ، رشته $(N_2 \gg l/4)$ را به دست می‌آورد (توجه شود که $N_2 \gg l/4$ را از درون :

$$\begin{aligned} M_1 \oplus M'_1 &= EPC_s \oplus N_1 \oplus K_i \oplus EPC_s \oplus N_1 \oplus K_{i+1} \\ &= EPC_s \oplus N_1 \oplus K_i \oplus EPC_s \oplus N_1 \oplus K_i \oplus (N_2 \gg l/4) \\ &= (N_2 \gg l/4) \end{aligned} \quad (۵)$$

(۳) سپس مهاجم با انجام XOR کردن پیام M_1 و M_3 ، کلید K_i را حذف می‌کند و رشته ۱۶ بیتی β را به دست می‌آورد:

$$\begin{aligned} M_3 \oplus M_1 \oplus N_1 &= CRC(EPC_s \oplus (N_2 \gg l/4)) \oplus K_i \oplus EPC_s \oplus N_1 \oplus K_i \oplus N_1 \\ &= CRC(EPC_s \oplus (N_2 \gg l/4)) \oplus EPC_s = \beta \end{aligned} \quad (۶)$$

(۴) فرض کنید که $L = \{l_1, l_2, \dots, l_{2^{16}}\}$ مجموعه تمام رشته بیت‌های ۱۶ بیتی باشد. چون که EPC_s یک‌رشته ۱۶ بیتی است پس $EPC_s \in L$. حال مهاجم به کمک β و با اجرای الگوریتم ۱ می‌تواند با حداکثر 2^{16} محاسبه برون‌خط به EPC_s دست پیدا کند.

Algorithm 1 (۷)

```

For  $1 \leq i \leq 2^{16}$ 
  Choose  $l_i \in L$ 
   $\alpha = CRC(l_i \oplus (N_2 \gg l/4)) \oplus l_i$ 
  if  $\alpha = \beta$  then
    return  $l_i$  as  $EPC_s$ 
End
    
```

۳-۲. محاسبه کلید نشست K_i و جعل هویت برچسب

در این قسمت نشان داده می‌شود که یک مهاجم چگونه می‌تواند با بهره‌گیری از روش شرح داده شده در بخش قبل، اقدام به محاسبه کلید نشست K_i و سپس جعل هویت برچسب قربانی

نقطه‌ضعف سوم که متوجه پروتکل MAP هست مربوط به ساختار به‌روزرسانی کلید K_i است که امکان نشت اطلاعات و به‌خطر افتادن کلیدهای مخفی برچسب قربانی می‌شود که برای رفع این مشکل نیز راه‌حل مناسبی ارائه می‌کنیم. که در پروتکل پیشنهادی بدون تغییری در طول رشته بیت EPC_s ، مشکلات ذکر شده رفع گردیده است.

پروتکل پیشنهادی دارای دو گام اولیه و گام احراز هویت است که به شرح زیر هست:

گام اولیه: این مرحله شبیه پروتکل MAP است.

گام احراز هویت: در این گام، فرایند احراز هویت بین کارت-خوان و سرویس‌دهنده‌ی نهایی صورت می‌گیرد. این فرایند از ۵ مرحله تشکیل شده است که به‌صورت زیر قابل بیان است.

(۱) در ابتدا کارت‌خوان عدد تصادفی N_1 را تولید می‌کند، سپس عدد N_1 را به برچسب ارسال می‌کند.

(۲) در این مرحله، برچسب در ابتدا عدد تصادفی N_2 را تولید می‌کند. سپس با استفاده از عدد N_2 مقادیر زیر را محاسبه کرده و به کارت‌خوان ارسال می‌کند:

$$\begin{aligned} M_1 &= PRNG(EPC_s \oplus N_1) \oplus PRNG(N_2) \oplus K_i, \\ CN_2 &= N_2 \oplus PRNG(K_i) \\ M_2 &= CRC(EPC_s \oplus N_2 \oplus C_i) \oplus K_i \end{aligned} \quad (21)$$

(۳) در این مرحله، کارت‌خوان داده‌های ارسال شده از برچسب را دریافت می‌کند و مقادیر (M_1, CN_2, M_2, N_1) را به سرویس‌دهنده‌ی نهایی ارسال می‌کند.

(۴) سرویس‌دهنده‌ی نهایی بعد از دریافت داده از کارت‌خوان، عملیات زیر را انجام می‌دهد:

- سرویس‌دهنده‌ی نهایی با استفاده از K_{old} و K_{new} ذخیره شده در پایگاه داده خود، عدد تصادفی N_2 را محاسبه می‌کند:

$$\begin{aligned} N_2^{old} &= CN_2 \oplus PRNG(K_{old}) \\ N_2^{new} &= CN_2 \oplus PRNG(K_{new}) \end{aligned}$$

- در ادامه با استفاده از مقادیر محاسبه شده N_2^{old} یا N_2^{new} ، سرویس‌دهنده نهایی مقادیر I_{old} و I_{new} را به‌صورت زیر محاسبه می‌کند:

$$\begin{aligned} I_{old} &= PRNG(EPC_s \oplus N_1) \oplus PRNG(N_2^{old}) \oplus K_{old} \\ I_{new} &= PRNG(EPC_s \oplus N_1) \oplus PRNG(N_2^{new}) \oplus K_{new} \end{aligned}$$

- سپس بررسی می‌کند که آیا مقدار $I_{old} \stackrel{?}{=} M_1$ یا $I_{old} \stackrel{?}{=} M_2$ است یا نه؟ و از این مقایسه مقدار $X = old$ یا $X = new$ را تعیین می‌کند.

$$M'_3 = CRC(EPC_s \oplus (N'_2 \gg l/4)) \oplus K_{i+1} \quad (14)$$

(۴) برچسب در ابتدا برقرار بودن رابطه $M'_3 \oplus K_{i+1} = CRC(EPC_s \oplus (N'_2 \gg l/4))$ را بررسی می‌کند و سپس هویت مهاجم را به‌عنوان کارت‌خوان مجاز تأیید کرده و مقادیر خود را به‌روزرسانی می‌کند:

$$C_{i+1} \leftarrow PRNG(N'_1 \oplus N'_2) \oplus K_{i+1} \quad (15)$$

$$K_{i+1} \leftarrow K_{i+1} \oplus (N'_2 \gg l/4) \quad (16)$$

۳-۴. حمله ناهم‌زمانی

بعد از جعل هویت کارت‌خوان مشاهده می‌شود که اکنون مقادیری به‌صورت مخفی بر روی برچسب به‌روزرسانی و ذخیره شده است که کارت‌خوان هیچ اطلاعی از آن‌ها ندارند و کلیدهای مخفی برچسب به‌صورت زیر است:

$$C_i \leftarrow PRNG(N'_1 \oplus N'_2) \oplus K_i \quad (17)$$

$$K_i \leftarrow K_i \oplus (N'_2 \gg l/4) \quad (18)$$

درحالی‌که، کلیدهای مخفی در سرویس‌دهنده نهایی به‌صورت زیر می‌باشند:

$$C_i \leftarrow PRNG(N_1 \oplus N_2) \oplus K_i \quad (19)$$

$$K_i \leftarrow K_i \oplus (N_2 \gg l/4) \quad (20)$$

در نتیجه طرفین ناهم‌زمان شده‌اند و هرگاه که برچسب قصد داشته باشد تا خود را به کارت‌خوان احراز هویت کند، کارت‌خوان هویت او را تأیید نخواهد کرد.

۴. پروتکل پیشنهادی بهبودیافته MAP

گرچه طراحان پروتکل MAP سعی کرده‌اند تا یک پروتکل امن و با ساختار مناسب طراحی کنند، با این حال نشان دادیم که به دلیل برخی رخنه‌های امنیتی، این پروتکل در برابر حملات مختلف آسیب‌پذیر است. برای برطرف کردن این ضعف‌ها و امن کردن پروتکل در برابر حملات انجام شده، ابتدا به ضعف‌های MAP اشاره نموده و سپس راهکارهای اصلاحی خود را پیشنهاد می‌کنیم.

اولین نکته‌ای که در رابطه با پروتکل MAP قابل ذکر است کوتاه بودن طول رشته بیت EPC_s است که دارای ۱۶ بیت است. به دلیل این‌که این رشته بیت در همه نشست‌ها مقدار ثابتی دارد بایستی در طراحی پیام‌هایی که شامل EPC_s هستند دقت زیادی شود در غیر این صورت امکان افشای آن بسیار زیاد خواهد بود که ما نیز این موضوع را نشان دادیم، در نتیجه سعی می‌کنیم تا این ضعف را برطرف کنیم.

نقطه ضعف دوم در نحوه تولید پیام توسط برچسب و همچنین نحوه احراز هویت در سرویس‌دهنده نهایی است.

و پارامتر زیر را به صورت زیر به روزرسانی می‌کند:

$$C_i \leftarrow PENG(N_1 \oplus N_2) \quad K_i = h(K_i \oplus (N_2 \gg l/4)) \quad (۲۲)$$

در غیراین صورت برچسب مرحله را متوقف می‌کند و پروتکل ساقط می‌شود. ساختار کلی پروتکل پیشنهادی در شکل (۲) در زیر آورده شده است. در پروتکل بهبودیافته از تابع چکیده ساز استفاده شده که به شرح زیر است.

تابع چکیده ساز: هدف اصلی از به کارگیری تابع

چکیده ساز، تولید رشته بیتی است که به هیچ عنوان قابل پیش‌بینی نباشد و در نتیجه به خواص تصادفی بودن نزدیک باشد. خاصیت تابع چکیده ساز این است که برگشت پذیر نیست.

۴-۱. تحلیل امنیتی پروتکل بهبودیافته

همان طور که مشاهده شد در پروتکل MAP به دلیل وجود نقاط ضعف در طراحی پروتکل از قبیل نحوه تولید پیام توسط برچسب، نحوه احراز هویت در سرویس دهنده نهایی و روش به روزرسانی مقادیر مخفی، مهاجم قادر به دست آوردن کلید مخفی و اعمال حمله‌های مختلفی از قبیل حمله جعل برچسب، حمله جعل کارت خوان و حمله نا هم زمانی است. در پروتکل پیشنهادی همه‌ی ضعف‌های اشاره شده برطرف شده که در ادامه تحلیل‌های امنیتی آن‌ها آورده شده است.

• بعد از تعیین مقدار $X = old$ یا $X = new$ سرویس دهنده نهایی برای احراز هویت برچسب مقدار $CRC(EPC_S \oplus N_2^X \oplus C_X) \oplus K_X$ را محاسبه کرده و با پیام M_2 مقایسه می‌کند و در صورت برابری آن‌ها، سرویس دهنده نهایی، برچسب مورد نظر را تایید هویت می‌نماید. در غیر این صورت، کارت خوان پیام خطا از سرویس دهنده نهایی دریافت می‌کند و پروتکل ساقط می‌شود.

(۵) بعد از احراز هویت موفق سرویس دهنده نهایی پیام و مقادیر (D_i, M_3) را به کارت خوان ارسال می‌کند. سپس سرویس دهنده نهایی مقادیر خود را به صورت زیر به روزرسانی می‌کند:

• اگر $X = new$ آنگاه:

$$C_{iold} \leftarrow C_i, C_{inew} \leftarrow PENG(N_1 \oplus N_2^{new})$$

$$K_{iold} \leftarrow K_{new}, K_{inew} \leftarrow h(K_{new} \oplus (N_2^{new} \gg l/4))$$

(۶) در این مرحله، کارت خوان مقدار (D_i, M_3) را دریافت می‌کند و سپس M_3 را به برچسب ارسال می‌کند.

(۷) برچسب مقدار $CRC(EPC_S \oplus (N_2 \gg l/4)) \oplus K_i$ محاسبه و بررسی می‌کند که آیا مقدار $M_3 \oplus K_i$ برابر $CRC(EPC_S \oplus (N_2 \gg l/4))$ است یا نه؟ اگر آن‌ها برابر بودند، برچسب با موفقیت سرویس دهنده نهایی را احراز هویت می‌کند

Database ($K_{old}, C_{old}, K_{new}, C_{new}, EPC_S, D_i$)	Reader		Tag (K_i, C_i, EPC_S)
For each $\{EPC_S, C_{old}, C_{new}, K_{old}, K_{new}\}$ $N_2^{old} = CN_2 \oplus PRNG(K_{old})$ $N_2^{new} = CN_2 \oplus PRNG(K_{new})$ Then computes values below: $I_{old} = PRNG(EPC_S \oplus N_1) \oplus PRNG(N_2^{old}) \oplus K_{old}$ $I_{new} = PRNG(EPC_S \oplus N_1) \oplus PRNG(N_2^{new}) \oplus K_{new}$ Matches $M_1 \stackrel{?}{=} I_X$ If $M_1 = I_{new}$ $X = new$ Else if I_{old} $X = old$ End Verify $M_2 \stackrel{?}{=} CRC(EPC_S \oplus N_2^X \oplus C_X) \oplus K_X$ Then computes values below: $M_3 = CRC(EPC_S \oplus (N_2^X \gg l/4)) \oplus K_X$ If $X = new$ $C_{old} \leftarrow C_{new} \leftarrow PENG(N_1 \oplus N_2^{new})$ $K_{old} \leftarrow K_{new} \leftarrow h(K_{new} \oplus (N_2^{new} \gg l/4))$ Else Do not update its secret value End	(1) $N_1 \rightarrow$	(2) $(M_1, CN_2, M_2) \leftarrow$	Generates random number N_2 $M_1 = PRNG(EPC_S \oplus N_1) \oplus PRNG(N_2) \oplus K_i$ $CN_2 = N_2 \oplus PRNG(K_i)$ $M_2 = CRC(EPC_S \oplus N_2 \oplus C_i) \oplus K_i$
	(3) $(M_1, CN_2, M_2, N_1) \leftarrow$		
	(4) $(M_2, D_i) \rightarrow$	(5) $M_3 \rightarrow$	Verify $M_3 \oplus K_i \stackrel{?}{=} CRC(EPC_S \oplus (N_2^X \gg l/4))$ $C_i \leftarrow PRNG(N_1 \oplus N_2)$ $K_i = h(K_i \oplus (N_2 \gg l/4))$
	$D \leftarrow R$ Channel (Secure)		$R \rightarrow T$ Channel (Insecure)

شکل ۲. پروتکل بهبودیافته

حمله کشف کلید جلوگیری کرده‌ایم و آن را با تابع چکیده ساز امن ساختیم در نتیجه این حمله امکان پذیر نیست.

حمله جعل: در پروتکل بهبودیافته با اعمال تغییراتی در نحوه تولید پیام توسط برچسب به صورت

$$M_1 = PRNG(EPC_S \oplus N_1) \oplus PRNG(N_2) \oplus K_i$$

و همچنین نحوه احراز هویت در سرویس دهنده نهایی، مهاجم قادر به جعل هویت برچسب و همچنین جعل هویت کارت خوان نیست.

علاوه بر این، به دلیل شباهت پروتکل بهبودیافته با پروتکل MAP، این پروتکل در مقابل حملاتی نظیر ناهم‌زمانی و ردیابی امن است. در جدول (۱) مقایسه‌ای از تحلیل عملکردی برخی پروتکل‌های مشابه با پروتکل بهبود یافته نشان داده شده است. همچنین در جدول (۲) تحلیل امنیتی و محرمانگی پروتکل بهبودیافته با تعدادی پروتکل‌های مشابه نشان داده شده است. مشاهده می‌شود که برخلاف سایر پروتکل‌ها امنیت و محرمانگی پروتکل بهبودیافته در برابر حملات موجود برقرار است.

حمله کشف کلیدهای مخفی: رخنه امنیتی که در پروتکل

MAP منجر به اعمال حمله کشف کلید گردید ناشی از نحوه به روزرسانی کلید مخفی بود که در پروتکل پیشنهادی برای رفع این مشکل، به روزرسانی مقادیر مخفی به صورت $K_i = h(K_i \oplus (N_2 \gg l/4))$ تغییر یافته است. ما با استفاده کردن از یک تابع چکیده‌ساز در ساختار به‌روزرسانی کلید K_i ، نقطه ضعف مورد نظر در پروتکل MAP را برطرف می‌کنیم. در واقع با تغییر نحوه به‌روزرسانی کلید مخفی، وابستگی بین پیام ارسالی از برچسب به کارت‌خوان و پیام ارسالی $(N_2 \gg l/4)$ از سرویس دهنده نهایی به کارت‌خوان و کلید مخفی K_{i+1} برطرف خواهد شد و مهاجم قادر به استفاده از این پیام‌ها برای به‌دست آوردن عدد تصادفی N_2 و کلید مخفی K_i نخواهد بود. در نتیجه مهاجم قادر به حمله کشف کلیدهای مخفی نیست.

حمله ناهم‌زمانی: این حمله به دلیل اعمال حمله کشف

کلید به‌وجود می‌آید و از آنجایی که ما در پروتکل بهبود یافته از

جدول ۱. تحلیل عملکردی پروتکل‌ها. H: تابع چکیده‌ساز PRNG: تابع شبه عدد تصادفی CRC: تابع CRC

عملکرد	پروتکل یه و همکارانش [۱۰]	پروتکل یون و همکارانش [۱۱]	پروتکل SPRS [۱۴]	پروتکل پیشنهادی
هزینه محاسبات در برچسب	$0H + 6 \times PRNG$	$0H + 6 \times PRNG$	$2 \times CRC + 2 \times PRNG$	$1H + 2 \times CRC + 3 \times PRNG$
هزینه محاسبات در سرور و کارت-خوان	$2H + 10 \times PRNG$	$4H + 10 \times PRNG$	$4 \times CRC + 2 \times PRNG$	$1H + 2 \times CRC + 4 \times PRNG$

جدول ۲. تحلیل مقایسه‌ای از تحلیل امنیتی پروتکل‌ها امن (⊙): نامن: ×

حمله‌ها	پروتکل یه و همکارانش [۱۰]	پروتکل یون و همکارانش [۱۱]	پروتکل پنگ و همکارانش [۱۹]	پروتکل بهبود داده شده
حمله کشف مقادیر مخفی	×	×	×	⊙
حمله تکرار	⊙	⊙	⊙	⊙
حمله جعل برچسب	×	×	×	⊙
حمله جعل کارت‌خوان	⊙	×	⊙	⊙
حمله ناهم‌زمانی	×	⊙	×	⊙

پروتکل توسط طراحان آن، این پروتکل در مقابل حمله‌هایی نظیر کشف کلید شناسه، جعل هویت برچسب، جعل هویت کارت‌خوان و ناهم‌زمانی امن نیست و این حمله‌ها می‌تواند روی این پروتکل انجام شود. همچنین، جهت تأمین امنیت و محرمانگی کاربران، یک نسخه‌ی بهبودیافته از پروتکل پنگ و همکارش ارائه شد. درنهایت جهت ارزیابی بیشتر، امنیت و محرمانگی پروتکل بهبودیافته مورد ارزیابی قرار گرفت و با برخی از پروتکل‌های

۵. نتیجه‌گیری

در این مقاله، ابتدا اشاره‌ای به مفاهیم حمله‌های کشف کلید شناسه، جعل هویت برچسب، جعل هویت کارت‌خوان و حمله‌ی ناهم‌زمانی شد. در ادامه به تحلیل امنیتی یک پروتکل احراز هویت سامانه‌های RFID مبتنی بر استاندارد EPC C-1 G-2 پرداخته شد. این پروتکل توسط پنگ و همکارش در سال ۲۰۱۳ ارائه شده بود. نشان داده شد که برخلاف ادعای امن بودن این

- مشابه موجود مورد مقایسه قرار گرفت و نشان داده شد که در برابر حملات موجود امن است.
- ۶. مراجع**
- [13] Alavi, S.M.; Baghery, K.; Abdolmaleki, B. "Security and Privacy Flaws in a Recent Authentication Protocol for EPC C1 G2 RFID Tags"; *Advances in Computer Science* 2014, 44-52.
- [14] Mardani Shahrabak, M.; Abdolmaleki, B.; Baghery, K. "Weaknesses of SPRS Authentication Protocol and Present a Developed Protocol for RFID Systems"; *Journal of Advanced Defence Science and Technology* 2016, 14-33 (In Persian).
- [15] Chien, H.; Chen, C. "Mutual Authentication Protocol for RFID Conforming to EPC Class-1 Generation-2 Standards"; *Computer Standards & Interfaces* 2007, 29, 254-259.
- [16] Han, D.; Kwon, D. "Vulnerability of an RFID Authentication Protocol Conforming to EPC Class-1 Generation-2 Standards"; *Computer Standards & Interfaces* 2009, 31, 648-652.
- [17] Habibi, M. H.; Gardeshi, M.; Alaghband, M. R. "Practical Attacks on a RFID Authentication Protocol Conforming to EPC C-1 G-2 Standard"; *International Journal of UbiComp* 2011, 1-13.
- [18] Habibi, M. H.; Alaghband, M. R.; Aref, M. R. "Attacks on a Lightweight Mutual Authentication Protocol under EPC C-1 G-2 Standard"; *Information Security Theory and Practice, Lecture Notes in Computer Science* 2011, 254-263.
- [19] Pang, L.; He, L.; Pei, Q.; Wang, Y. "Secure and Efficient Mutual Authentication Protocol for RFID Conforming to the EPC C-1 G-2 Standard"; *Wireless Communication and Networking Conference* 2013, 1870-1875.
- [20] Bolan, C. "Spoofing Attack against an EPC Class One RFID"; *7th Australian Information Security Management Conference, Western Australia System*, 2009.
- [21] Hernández Castro, J. C.; Peris-Lopez, P.; Phan, R. C.-W.; Estévez-Tapiador, J. M. "Cryptanalysis of the David-Prasad RFID Ultra-Light Weight Authentication Protocol"; *Dynamics in Logistics* 2016, 291-301.
- [22] Habibi, M. H.; Gardeshi, M.; Alaghband, M. R. "Cryptanalysis of Two Mutual Authentication Protocols for Low-Cost RFID"; *International Journal of Distributed and Parallel Systems* 2011, 2, 103-114.
- [23] Kim, H. "Desynchronization Attack on Hash-Based RFID Mutual Authentication Protocol"; *J. Secur. Eng.* 2012, 9, 357-366.
- [1] Australia, E.-C. "Access Control, Sensor Control, and Transponders"; Available at: <http://www.rfid.com.au/rfiduhf.htm>, 2008.
- [2] Hoepman, J. H.; Hubbers, E.; Jacobs, B.; Oostdijk, M.; Scherer, R. W. "Crossing Borders: Security and Privacy Issues of the European E-Passport"; *IWSEC 2006, LNCS 4266 Springer-Verlag Berlin Heidelberg*, 2006, 152-167.
- [3] "Transport for London, Oyster"; Available at: <http://www.tfl.gov.uk/tickets/27298.aspx>. [Accessed 01 02 2014].
- [4] Wyld, D. C. "24-Karat Protection: RFID and Retail Jewelry Marketing"; *Int. J. UbiComp* 2010, 1, 1-14.
- [5] Khedo, K.; Sathan, D.; Elaheebocus, R.; Subramanian, R. K.; Rughooputh, S. D. V. "Overlapping Zone Partitioning Localization Technique for RFID"; *Int. J. UbiComp* 2010, 1, 20-32.
- [6] Erick, C. J.; Chung, C. A. "RFID and Auto-ID in Planning and Logistics: A Practical Guide for Military UID Applications"; *CRC Press*, 2016.
- [7] Florian, P.; Müller, M.; Silveira, M.; Thoro, L.; Schmidt, M.; Schenk, M. "Applying Product-Integrated RFID Transponders for Tracking Vehicles Across the Automotive Life Cycle"; *Dynamics in Logistics* 2016, 291-301.
- [8] "EPCglobal Inc."; Available at: <http://www.epcglobalinc.org>. [Accessed 02 01 2014].
- [9] "EPCglobal Inc., EPCTM Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocols for Communications at 860 MHz – 960 MHz Version 1.1.0";
- [10] Yeh, T. C.; Wang, Y. J.; Kuo, T. C.; Wang, S. S. "Securing RFID Systems Conforming to EPC Class-1 Generation-2 Standards"; *Expert Systems with Applications* 2010, 37, 7678-7683.
- [11] Yoon, E.-J. "Improvement of the Securing RFID Systems Conforming to EPC Class-1 Generation-2 Standards"; *Expert Syst. Appl.* 2012, 39, 1589-1594.
- [12] Xiao, F.; Zhou, Y.; Zhou, J.; Zhu, H.; Niu, X. "Security Protocol for RFID System Conforming to EPC-C1G2 Standard"; *J. Comput.* 2013, 605-612.