مجله علمی پژوهشی «علوم و فناوری های پدافند نوین»
سال هفتم، شماره ۳، پاییز ۱۳۹۵؛ ص ۱۹۱-۱۸۳

شناخت آسیب‌های امنیتی پروتکل احراز اصالت MAP و ارائه پروتکل بهبودیافته 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 بهبودیافته پیشنهاد داده و نشان می‌دهیم این پروتکل در مقابل حملات ذکرشده امن است. بعلاوه پیچیدگی و کارایی پروتکل‌های ارایه 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 بهبودیافته 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. Shahrbabak*, 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 سال‌های اخیر، استفاده از سامانه‌های «شناسایی از طریق امواج رادیویی»[1] (RFID) رشد چشم‌گیری داشته است. این سامانه‌ها در دستگاه‌های کنترل دسترسی‌یابی [۱]، گذرنامه‌ی الکترونیکی [۲]، کنترل تردد [3]، دستگاه‌های ضد سرقت اتومبیل‌ها، تشخیص هواپیماهای جنگنده خودی از دشمن و خیلی از موارد دیگر مورداستفاده قرار می‌گیرد [۴ و ۵]. این سامانه در ارتش برای برنامه‌ریزی و تدارکات جنگی بصورت گسترده استفاده می‌شود [۲۰] و همینطور یک روش شناخته شده برای دنبال کردن و ردگیری وسایل نقلیه است [۲۱]. سامانه‌های RFID، قابلیت شناسایی خودکار اشیا، با استفاده از یک تراشه‌ی کوچک ارزان‌قیمت را فراهم می‌کنند. این تراشه‌ی کوچک به‌اصطلاح «برچسب»[2] یا «برچسب هوشمند»[3] نامیده می‌شود. داده‌های ذخیره‌شده در این برچسب‌ها توسط دستگاه‌هایی تحت عنوان «کارت‌خوان»[4] خوانده می‌شود. در حالت کلی سامانه‌های RFID از سه بخش اصلی تشکیل‌شده‌اند که شامل برچسب، کارت‌خوان و «سرویس‌دهنده‌ی نهایی»[5] است. تمام اطلاعات و کلیدهای مخفی برچسب‌ها در سرویس‌دهنده‌ی نهایی ذخیره می‌شود. کارت‌خوان مابین برچسب و سرویس‌دهنده‌ی نهایی قرار می‌گیرد و وظیفه‌ی تبادل داده بین آن‌ها را بر عهده دارد.

در سال‌های اخیر، یکی از استانداردهای محبوب برای برچسب‌های فعال استاندارد «EPC C1 G2»[6] است که توسط سازمان «EPCglobal»[7] ارائه شده است [۶ و ۷]. استاندارد EPC C1 G2 برای تشخیص خطا در داده‌های مبادله شده، از عملگرهایی نظیر CRC[8]، XOR و PRNG[9] استفاده می‌کند. تلاش محققان برای طرحی پرتکلی امن و استاندار می‌باشد که در این راستا یه و همکاران [۱۰]، یون و همکاران [۱۸]، خیایو و همکاران [۲۲] و همچنین علوی و همکاران [۲۳] پروتکل‌های احراز هویت بر روی این استاندارد را پیشنهاد کرده اند. از آنجایی که این سامانه جدید بوده و بصورت فراگیر مورد استفاده قرار می‌گیرد، در کشور ما نیز مورد استفاده قرار گرفته که تحقیقاتی در این راستا به نشر رسیده است[۱۹]. تا کنون پروتکل‌های احراز هویت زیادی مبتنی بر استاندارد EPC C1 G2 ارائه شده است [۸ - ۱۰]. با وجود این‌که همه‌ی آن‌ها سعی در فراهم آوردن امنیت و محرمانگی کاربران RFID بودند، ولی ایرادهایی بر همه‌ی آن‌ها وارد شده است. «یه» و همکارانش [۱۰] پروتکل احراز هویت متقابلی تحت عنوان SRP را برای سامانه‌های RFID پیشنهاد کردند. حبیبی و همکارانش [۱۱ و ۱۲] نشان دادند که پروتکل SRP ضعف‌هایی دارد. همچنین، «پنگ» و همکارانش [۱۳] «پروتکل احراز هویت متقابلی» (MAP) مطابق با استاندارد EPC C1 G2 برای سامانه‌های RFID ارائه کردند. آن‌ها ادعا کردند پروتکل MAP امن است و می‌تواند امنیت کاربران RFID را تأمین کند.

در این مقاله نشان می‌دهیم برخلاف ادعای طراحان پروتکل MAP، این پروتکل همچنان امن نیست و ضعف‌هایی بر این پروتکل وارد است. سه حمله‌ی کشف کلید پروتکل، جعل هویت برچسب و جعل هویت کارت خوان بر روی این پروتکل انجام می‌پذیرد. در ادامه اشاره‌ی مختصری به مفاهیم حمله‌های انجام‌شده در این مقاله می‌شود.

حمله جعل: در این نوع حمله، مهاجم سعی می‌کند به نحوی بتواند خود را به‌جای یک برچسب و یا یک کارت‌خوان مجاز جا بزند. این حمله می‌تواند هم برای جعل هویت یک برچسب مجاز و هم جعل هویت یک کارت‌خوان مجاز استفاده شود. روش‌های گوناگونی وجود دارند که یک مهاجم را قادر می‌سازند تا روی یک پروتکل حمله جعل هویت انجام دهد. عامل کلیدی که یک مهاجم را قادر به انجام حمله جعل می‌کند دست‌یابی او به اطلاعات محرمانه مربوط به برچسب و یا کارت‌خوان است که این اطلاعات می‌تواند شامل شناسه، کلیدهای مخفی و حتی برخی از اطلاعات جزئی قابل‌استخراج از پیام‌های تبادلی باشند. در حمله جعل هویت یک برچسب مجاز، مهاجم با اطلاعاتی که از شنود ارتباطات مربوط به آن برچسب به دست آورده است، قادر می‌شود که به پرسمان‌های یک کارت‌خوان مجاز به‌درستی پاسخ دهد و خود را به‌عنوان یک برچسب مجاز به کارت‌خوان اثبات کند. این کار می‌تواند با ایجاد تغییراتی در پیام‌های تبادلی در نشست‌های قبلی نیز صورت گیرد. از طرف دیگر، یک مهاجم می‌تواند به کمک روش‌های ذکرشده در فوق، هویت یک کارت‌خوان و یا یک سرویس‌دهنده مجاز را نیز جعل کند و در برگزاری یک نشست با برچسب موردنظر، پروتکل را به‌طور کامل اجرا کرده و به پایان برساند. این کار باعث می‌شود که مهاجم به اهدافی شامل دستیابی به اطلاعات محرمانه مربوط به آن برچسب و یا ناهمزمان کردن آن برچسب و سرویس‌دهنده باشد، دست یابد [۱۴].

حمله کشف کلید و شناسه: پروتکل‌های احراز هویت به کار گرفته‌شده در سامانه‌های RFID می‌توانند در برابر حملات گوناگون کشف کلید و شناسه آسیب‌پذیری داشته باشند. در این‌گونه حملات مهاجم سعی می‌کند با یافتن رخنه‌های امنیتی در پروتکل موردنظر، با روش‌های متفاوتی اقدام به کشف کلید یا کلیدهای مخفی و یا شناسه‌ی برچسب قربانی کند. یکی از رایج‌ترین روش‌های کشف کلید و شناسه، روش جستجوی جامع فضای کلید است. اگر کلیدهای مخفی به کار گرفته‌شده در پروتکل، رشته بیت‌هایی با طول کم باشند آنگاه این فرصت برای مهاجم فراهم

[1] Radio 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)

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

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

حمله ناهم‌زمانی: پروتکل‌های احراز هویت به کار گرفته‌شده 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'، $t' > t$ ردیابی کند، زیرا مقادیر استفاده‌شده برای شناسایی برچسب در زمان‌های t و t' متفاوت خواهند بود.

بااین‌حال همین ویژگی باعث به وجود آمدن آسیب‌پذیری‌های دیگری برای پروتکل‌های موردبحث می‌شود. در واقع اگر برچسب و سرویس‌دهنده به هر دلیلی نتوانند مقادیر یکسانی را به‌روزرسانی کنند این امر موجب نشود تا مقادیر مخفی ذخیره‌شده در برچسب و سرویس‌دهنده متفاوت باشد و درنتیجه در نشست‌های بعدی هنگامی‌که 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 تائید هویت آن 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 پرداخته‌شده است و مقایسه‌ای از تحلیل امنیتی پروتکل پیشنهادشده 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 داده‌شده است. در پروتکل MAP کانال ارتباطی بین کارت‌خوان و سرویس‌دهنده‌ی نهایی امن در نظر گرفته‌شده 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

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

شکل۱. پروتکل MAP [۱۹].

جدول ۱. نمادهای استفاده شده در پروتکل MAP

نماد	توضیحات
$\mathbf{EPC}_s$	یک مقدار ۱۶بیتی که از Xor کد استاندارد EPC ایجادشده
$\boldsymbol{K}_\mathbf{i}$	کلید خصوصی ذخیره‌شده در برچسب و پایگاه داده برای احراز هویت
$\boldsymbol{P}_\mathbf{i}$	کلید دست‌یابی ذخیره‌شده در برچسب و پایگاه داده
$\mathbf{C}_\mathbf{i}$	ایندکس ذخیره‌شده در برچسب برای جستجوی آن در پایگاه داده
RID	شناسه کارتخوان
$\boldsymbol{P}_{\boldsymbol{old}}$	کلید دست‌یابی قبلی ذخیره‌شده در پایگاه داده
$\boldsymbol{K}_\mathbf{old}$	کلید خصوصی قبلی ذخیره‌شده در پایگاه داده
$\boldsymbol{K}_\mathbf{new}$	کلید خصوصی جدید ذخیره‌شده در پایگاه داده
$\mathbf{C}_\mathbf{old}$	ایندکس قبلی پایگاه داده ذخیره‌شده در پایگاه داده
$\mathbf{C}_\mathbf{new}$	ایندکس جدید پایگاه داده ذخیره‌شده در پایگاه داده
$\mathbf{D}_\mathbf{i}$	ریز اطلاعات ذخیره‌شده از برچسب در پایگاه داده
h	تابع چکیده‌ساز

پروتکل MAP از دو گام اصلی تشکیل شده است که به‌صورت 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 سرویس‌دهنده‌ی نهایی بارگذاری شده‌اند، با مقادیر اولیه‌ی تصادفی K_0، P_0، C_0 که توسط تولیدکننده تولید شده‌اند، مقداردهی می‌شوند. به‌این‌ترتیب 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_0مقداردهی 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 می‌کند. در ادامه با استفاده از این مسئله که $K_i = K_{old}$ یا $K_i = K_{new}$، سرویس‌دهنده نهایی مقدار $N_2 = CN_2 \oplus PRING(K_i)$ 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 $M_3 = CRC(EPC_s \oplus (N_2 \gg l/4)) \oplus K_i$ را محاسبه می‌کند و مقادیر (D_i, M_3) را به کارت‌خوان ارسال می‌کند.

ب) سپس سرویس‌دهنده نهایی مقادیر خود را به‌صورت زیر به‌روزرسانی می‌کند:

$$C_{iold} \leftarrow C_i, C_{inew} \leftarrow PENG(N_1 \oplus N_2)$$
$$K_{iold} \leftarrow K_i, K_{inew} \leftarrow K_i \oplus (N_2 \gg l/4) \quad (۱)$$

اگر آن‌ها برابر بودند، برچسب با موفقیت سرویس‌دهنده نهایی را احراز هویت می‌کند و پارامترهای زیر را به‌صورت زیر به‌روزرسانی می‌کند:

$$C_i \leftarrow PENG(N_1 \oplus N_2)$$
$$K_i \leftarrow K_i \oplus (N_2 \gg l/4) \quad (۲)$$

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 به‌گونه‌ای طراحی کنند که شناسه برچسب به‌صورت آشکار ارسال نشود و درنتیجه 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 به‌صورت آشکار ارسال می‌شوند شناسه برچسب را محاسبه کرد. روند انجام

حمله به ترتیب زیر است.

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 تشکیل‌شده‌اند:

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

$$M_1' = EPC_s \oplus N_1 \oplus K_{i+1} \quad (۴)$$

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 :

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

۳) سپس مهاجم با انجام XOR کردن پیام M_1 وM_3، کلید K_i را حذف می‌کند و رشته ۱۶ بیتی β را به دست می‌آورد:

$$M_3 \oplus M_1 \oplus N_1 = CRC\big(EPC_s \oplus (N_2 \gg l/4)\big) \oplus K_i \oplus EPC_s \oplus N_1 \oplus K_i \oplus N_1 = CRC\big(EPC_s \oplus (N_2 \gg l/4)\big) \oplus EPC_s = \beta \quad (۶)$$

۴) فرض کنید که $L = \{l_1, l_2, \dots, l_{2^{16}}\}$ مجموعه تمام رشته بیت‌های ۱۶ بیتی باشد. چون‌که 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 ≤ i ≤ 2^16
    Choose l_i ∈ L
α = CRC(l_i ⊕ (N_2 ≫ l/4)) ⊕ l_i
  if α = β 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 داده‌شده در بخش قبل، اقدام به محاسبه کلید نشست K_i و سپس جعل هویت برچسب قربانی کند. مهاجم با روش شرح داده‌شده 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 برگزارشده 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 به‌دست 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:

$$M_1' = EPC_s \oplus N_1' \oplus K_{i+1} \quad (۱۰)$$

$$CN_2' = N_2' \oplus PRNG(K_{i+1}) \quad (۱۱)$$

$$M_2' = CRC(EPC_s \oplus N_2' \oplus C_{i+1}) \oplus K_{i+1} \quad (۱۲)$$

چون‌که هر سه پیام با مقادیر معتبر و صحیحی محاسبه شده‌اند مورد پذیرش کارت‌خوان قرار می‌گیرد و هویت مهاجم به‌عنوان برچسب قربانی مورد تائید 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 برگزارشده بین برچسب قربانی T_j و کارت‌خوان را شنود کرده و با روش شرح داده‌شده 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:

$$N_2' = CN_2' \oplus PRNG(K_{i+1}) \quad (۱۳)$$

$$M_3' = CRC(EPC_s \oplus (N_2' \gg l/4)) \oplus K_{i+1} \quad (۱۴)$$

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 تائید کرده و مقادیر خود را به‌روزرسانی می‌کند:

$$C_{i+1} \leftarrow PRNG(N_1' \oplus N_2') \oplus K_{i+1} \quad (۱۵)$$

$$K_{i+1} \leftarrow K_{i+1} \oplus (N_2' \gg l/4) \quad (۱۶)$$

۳-۴. حمله ناهم‌زمانی

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

$$C_i \leftarrow PRNG(N_1' \oplus N_2') \oplus K_i \quad (۱۷)$$

$$K_i \leftarrow K_i \oplus (N_2' \gg l/4) \quad (۱۸)$$

درحالی‌که، کلیدهای مخفی در سرویس‌دهنده نهایی به‌صورت زیر می‌باشند:

$$C_i \leftarrow PENG(N_1 \oplus N_2) \oplus K_i \quad (۱۹)$$

$$K_i \leftarrow K_i \oplus (N_2 \gg l/4) \quad (۲۰)$$

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

۴. پروتکل پیشنهادی بهبودیافته MAP

گرچه طراحان پروتکل MAP سعی کرده‌اند تا یک پروتکل امن و با ساختار مناسب طراحی کنند، بااین‌حال 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 اصلاحای خود را پیشنهاد می‌کنیم.

اولین نکته‌ای که در رابطه با پروتکل MAP قابل‌ذکر 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.

نقطه‌ضعف سوم که متوجه پروتکل MAP هست مربوط به ساختار به‌روزرسانی کلید K_i است که امکان نشت اطلاعات و به خطر 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 ذکرشده 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 تشکیل‌شده است که به‌صورت زیر قابل‌بیان 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 می-کند. سپس با استفاده از عدد N_2 مقادیر زیر را محاسبه کرده و به کارت‌خوان ارسال می‌کند.:

$$M_1 = PRNG(EPC_S \oplus N_1) \oplus PRNG(N_2) \oplus K_i,$$
$$CN_2 = N_2 \oplus PRNG(K_i)$$
$$M_2 = CRC(EPC_S \oplus N_2 \oplus C_i) \oplus K_i \quad (۲۱)$$

۳) در این مرحله، کارت‌خوان داده‌های ارسال‌شده از برچسب را دریافت می‌کند و مقادیر (M_1, CN_2, M_2, N_1) را به سرویس-دهنده‌ی 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} ذخیره‌شده در پایگاه داده خود، عدد تصادفی N_2 را محاسبه می‌کند:

$$N_2^{old} = CN_2 \oplus PRING(K_{old})$$
$$N_2^{new} = CN_2 \oplus PRING(K_{new})$$

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

$$I_{old} = PRNG(EPC_S \oplus N_1) \oplus PRNG(N_2^{old}) \oplus K_{old}$$
$$I_{new} = PRNG(EPC_S \oplus N_1) \oplus PRNG(N_2^{new}) \oplus K_{new}$$

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

• بعد از تعیین مقدار $X = old$ یا $X = new$، سرویس‌دهنده 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، سرویس‌دهنده نهایی، برچسب موردنظر را تایید هویت می‌نماید. در غیر این صورت، کارت‌خوان پیام خطا از سرویس‌دهنده نهایی دریافت می‌کند و پروتکل ساقط می‌شود.

۵) بعد از احراز هویت موفق سرویس‌دهنده نهایی پیام $M_3 = CRC(EPC_s \oplus (N_2^X \gg l/4)) \oplus K_X$ را محاسبه نموده و مقادیر (D_i, M_3) را به کارت‌خوان ارسال می‌کند. سپس سرویس‌دهنده نهایی مقادیر خود را به‌صورت زیر به‌روزرسانی 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، کارت‌خوان مقدار (D_i, M_3) را دریافت می‌کند و سپس M_3 را به برچسب ارسال می‌کند.

۷) برچسب مقدار $CRC(EPC_s \oplus (N_2 \gg l/4)) \oplus K_i$ را 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 سرویس‌دهنده نهایی را احراز هویت می‌کند و پارامتر زیر را به‌صورت زیر به‌روزرسانی 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 به‌کارگیری تابع چکیده‌ساز، تولید رشته بیتی است که به هیچ عنوان قابل پیش‌بینی نباشد و در نتیجه به خواص تصادفی بودن نزدیک باشد. خاصیت تابع چکیده‌ساز این است که برگشت‌پذیر نیست.

۴-۱. تحلیل امنیتی پروتکل بهبودیافته

همان‌طور که مشاهده شد در پروتکل MAP به دلیل وجود نقاط ضعف در طراحی پروتکل از قبیل نحوه‌ی تولید پیام توسط برچسب، نحوه‌ی احراز هویت در سرویس‌دهنده نهایی و روش به‌روزرسانی مقادیر مخفی، مهاجم قادر به دست آوردن کلید مخفی و اعمال حمله‌های مختلفی از قبیل حمله جعل برچسب، حمله جعل کارت‌خوان و حمله ناهم زمانی است. در پروتکل پیشنهادی همه‌ی ضعف‌های اشاره شده برطرف شده که در ادامه تحلیل‌های امنیتی آن‌ها آورده شده است.

Database $(K_{old}, C_{old}, K_{new}, C_{new}, EPC_s, D_i)$	Reader		Tag (K_i, C_i, EPC_s)
For each $\{EPC_s, C_{old}, C_{new}, K_{old}, K_{new}\}$ $N_2^{old} = CN_2 \oplus PRING(K_{old})$ $N_2^{new} = CN_2 \oplus PRING(K_{new})$ 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* *D*o not update its secret value *End*		**(1)** $N_1 \rightarrow$	Generates random number N_2 $M_1 = PRNG(EPC_S \oplus N_1) \oplus PRNG(N_2) \oplus K_i$. $CN_2 = N_2 \oplus PRNG(K_i)$ $M_2 = CRC(EPC_S \oplus N_2 \oplus C_i) \oplus K_i$
	(2) $(M_1, CN_2, M_2) \leftarrow$		
	(3) $(M_1, CN_2, M_2, N_1) \leftarrow$		
	(4) $(M_2, D_i) \rightarrow$		$Verify$ $M_3 \oplus K_i \stackrel{?}{=} CRC(EPC_s \oplus (N_2^X \gg l/4))$ $C_i \leftarrow PRNG(N_1 \oplus N_2)$ $K_i = h(K_i \oplus (N_2 \gg l/4))$
		(5) $M_3 \rightarrow$	
$D \leftarrow R$ Channel (Secure)		$R \rightarrow T$ Channel (Insecure)	

شکل ۲. پروتکل بهبودیافته

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، به‌روزرسانی مقادیر مخفی به‌صورت $K_i = h(K_i \oplus (N_2 \gg l/4))$ 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 $M_3 = CRC(EPC_s \oplus (N_2 \gg l/4)) \oplus K_i$ از سرویس‌دهنده 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 حمله کشف کلید جلوگیری کرده‌ایم و آن را با تابع چکیده ساز امن ساختیم در نتیجه این حمله امکان‌پذیر نیست.

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

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

و همچنین نحوه احراز هویت در سرویس‌دهنده نهایی، مهاجم قادر به جعل هویت برچسب و همچنین جعل هویت کارت‌خوان 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.

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	$0H + 6 \times PRNG$	$0H + 6 \times PRNG$	$2 \times \mathrm{CRC} + 2 \times PRNG$	$1H + 2 \times \mathrm{CRC} + 3 \times PRNG$
هزینه محاسبات در سرور و کارت-خوان	$2H + 10 \times PRNG$	$4H + 10 \times PRNG$	$4 \times \mathrm{CRC} + 2 \times PRNG$	$1H + 2 \times \mathrm{CRC} + 4 \times PRNG$

جدول۲. تحلیل مقایسه‌ای از تحلیل امنیتی پروتکل‌ها امن⊙: ناامن: ×

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

۵. نتیجه‌گیری

در این مقاله، ابتدا اشاره‌ای به مفاهیم حمله‌های کشف کلید شناسه، جعل هویت برچسب، جعل هویت کارت‌خوان و حمله‌ی ناهم زمانی 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 ارائه‌شده بود. نشان داده شد که برخلاف ادعای امن بودن این 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

مشابه موجود مورد مقایسه قرار گرفت و نشـان داده شـد کـه در برابر حملات موجود امن است.

۶. مراجع

[1] Australia, E.-C. “Access Control, Sensor Control, and Trans-Ponders”; 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.; Thoroe, 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.

[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 Shahrbabak, 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-1Generation-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 Conferance 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.