تشخیص تغییر در داده‌های برچسب رادیوشناسه با استفاده از نشانه‌گذاری مبتنی بر توابع درهم‌ساز توسط شبکه عصبی مصنوعی

ابراهیم شفیعی[1]، سید محمدرضا موسوی میرکلائی[2] و ابوالفضل فلاحتی[3]

[1]دانشجوی دکتری دانشکده مهندسی برق، دانشگاه علم و صنعت ایران

[2]استاد دانشکده مهندسی برق، دانشگاه علم و صنعت ایران، M_Mosavi@iust.ac.ir

[3]دانشیار دانشکده مهندسی برق، دانشگاه علم و صنعت ایران

چکیده

امروزه با مطرح شدن حملات دست‌کاری، حفاظت از داده‌ی برچسب رادیوشناسه (RFID) اهمیت یافته است. رویکرد نشانه‌گذاری مانع از تغییر غیرمجاز محتوای چنین برچسب‌هایی می‌گردد. به دلیل محدودیت‌های برچسب‌ها از قبیل ساختار ساده، تعداد ۹۶ بیت حافظه و دودویی بودن محتوای آن، به‌کارگیری روش‌های متعارف درهم‌سازی و نشانه‌گذاری را غیرممکن می‌کند. بدین ترتیب از رویکردهایی که قابلیت اعمال بر رشته بیت دودویی را دارند، استفاده می‌شود. این مقاله مبتنی بر الگوریتم ویژه‌ای از شبکه عصبی است که به‌منظور ایجاد چکیده پیام و کد نشانه استفاده شده است که باعث می‌گردد تعداد ۸ بیت از حافظه ارزشمند برچسب، جهت نشانه‌گذاری مورد استفاده قرار گیرد. در الگوریتم پیشنهادی تمام بیت‌های برچسب تحت نشانه‌گذاری قرار می‌گیرند و مکان بیت‌های نشانه قابل شناسایی نیست. این روش علاوه بر عدم نیاز به محرمانه بودن الگوریتم نشانه‌گذاری، مزایای دیگری از قبیل سهولت پیاده‌سازی و سرعت اجرا را داراست. مزیت دیگر رویکرد پیشنهادی، شبکه عصبی است که یک تابع یک‌طرفه محسوب می‌شود. این شبکه علاوه بر جایگشت، جانشینی و ابهام ایجاد می‌کند و می‌تواند همبستگی چکیده پیام را با کد کمینه نماید. فرآیند دو مرحله‌ای الگوریتم با استفاده از کلید محرمانه و مکان بیت‌های نشانه شبه‌تصادفی، مانع تحلیل الگوریتم می‌شود.

واژه‌های کلیدی

رادیوشناسه، امنیت RFID، نشانه‌گذاری، شبکه عصبی.

مقدمه

یکی از روش‌های شناسایی خودکار اشیاء، انسان و یا حیوانات، شناسایی از طریق امواج رادیویی[1] است که عملکرد آن شبیه بارکد می‌باشد. برچسب‌های هوشمند بیانگر سیستم‌هایی است که از امواج رادیویی به‌منظور انتقال اطلاعات مربوط به هویت یک شیء استفاده می‌کنند. پرکاربردترین برچسب RFID، برچسب EPC[2] می‌باشد. برچسب‌های کد الکترونیک تولید (EPC) شامل ریزتراشه‌های[3] الکترونیکی و آنتن‌های کوچکی هستند که انتقال اطلاعات ریزتراشه توسط آنتن مذکور انجام می‌شود [۱و۲]. در شکل ۱ ساختار برچسب RFID و یک سیستم نوعی رادیوشناسه به‌منظور پیاده‌سازی الگوریتم پیشنهادی مشاهده می‌گردد. عواقب تغییر در داده‌ی برچسب بدیهی است؛ اما برای توجه به اهمیت مطلب، به ذکر سه مثال اکتفا شده است. با تغییر در داده‌ی برچسب اولاً می‌توان کتاب را از مبادی کتابخانه بدون شناسایی و مکانیزم امانت عبور داد. ثانیاً، دست‌کاری رادیوشناسه کالا می‌تواند منجر به تغییر بهاء و نوع کالا گردد و درنهایت دست‌کاری رادیوشناسه، باعث تغییر در هویت افراد می‌گردد تا به آن‌ها اجازه‌ی ورود به یک حیطه داده شود؛ بنابراین تشخیص انواع تغییر در داده‌های برچسب RFID دارای اهمیت زیادی است [۳و۴].

این تحقیق بر روی برچسب‌های رادیوشناسه‌ی که شامل یک کد ۱۲۸ بیت قابل خواندن و نوشتن به تعداد نامحدود است، قابل

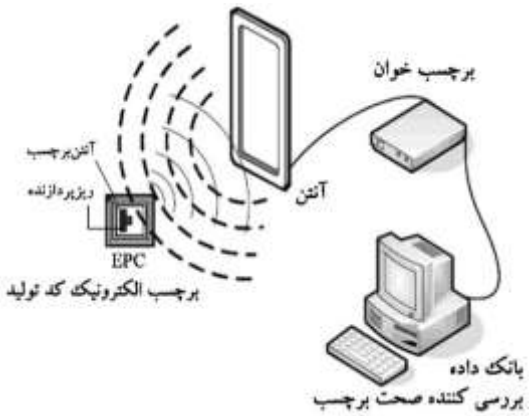
[1]RFID

[2]Electronic Product Code

[3]Micro Chip

تاریخ دریافت : ۱۳۹۴/۱۱/۰۴ تاریخ پذیرش : ۱۳۹۶/۱۰/۲۱

پیاده‌سازی می‌باشد. از این ۱۲۸ بیت، ۹۶ بیت برای کد EPC و ۳۲ بیت برای تصحیح خطای داده‌ها و فرمان کشتن[۴] به کار می‌رود. ساختار ذخیره داده‌ی «کد تولید» در چنین برچسب‌هایی بر اساس جدول ۱ است. میدان بیت‌های سرآیند[۵] بیانگر طول، نوع، ساختار، ویرایش و نسل کد EPC است. «مدیریت EPC» تولیدکننده‌ی برچسب را مشخص می‌کند. میدان[۶] «OC[۷]» تعیین کننده‌ی نوع شئ‌ای است که برچسب بر آن نصب می‌شود و شماره‌ی سریال رادیوشناسه، برای تمایز هر شئ از یک نوع، توسط تولیدکننده مورد استفاده قرار می‌گیرد [۵و۶].



شکل ۱ . نمای کلی از ساختار RFID.

جدول ۱. جدول ساختار ۹۶ بیت EPC [۶].

Header	(EM) EPC Manager	(OC) Obiect Class	(SN) Seriyal Number
۸ بیت	۲۸ بیت	۲۴ بیت	۳۶ بیت

بررسی کارهای پیشین

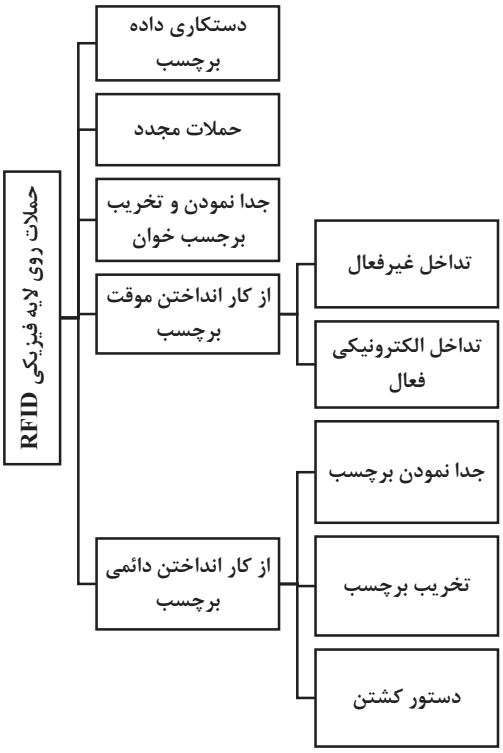
شکل ۲ حملات بر روی لایه فیزیکی RFID را نشان می‌دهد. تحقیقات فراوانی در مورد امنیت RFID صورت گرفته است. فیلدهفر و همکارانش [۷] به بررسی الگوریتم رمزنگاری AES[۸] در حوزه امنیت RFID پرداخته‌اند که قابل پیاده‌سازی بر روی RFIDهای ارزان قیمت نمی‌باشند.

در مرجع [۹]، RFID را بر مبنای احراز هویت مبتنی بر کلید عمومی و خصوصی، بررسی نموده‌اند. جولز و همکارانش [۸]، تحقیقاتی در مورد دست‌کاری برچسب‌های RFID انجام داده‌اند و راهکارهای افزایش امنیت را بررسی کرده‌اند. مرجع [۱۱]، با ارائه برچسبی بر اساس کلید عمومی و خصوصی باعث افزایش امنیت برای برچسب‌ها شده است.

لیگل [۱۰]، روش جدیدی برای پویش کردن فیلدهای RFID ارائه نموده است.

در مرجع [۱۲و۱۳]، چنگ از نشانه‌گذاری شکننده بر روی حافظه‌ی برچسب برای جلوگیری از دست‌کاری و تصادم[۹] استفاده کرده است. در این طرح نشانه در ۸ بیت از ۳۶ بیت شماره‌سریال برچسب تعبیه شده است و یک بیت دیگر نیز از شماره‌سریال به‌عنوان چک توازن[۱۰] برای صحت نشانه اختصاص یافته است؛ بنابراین در مجموع ۹ بیت برای نشانه در نظر گرفته شده که ۸ بیت اصلی آن از خروجی تابع درهم‌سازی[۱۱] است که ورودی تابع درهم‌ساز بیت‌های «مدیریت EPC» و «OC» می‌باشد.

در مرجع [۱۴] تشخیص دست‌کاری در برچسب‌های RFID با استفاده از نشانه‌گذاری شکننده مبتنی بر آشوب[۱۲] بیان می‌شود که نه تنها توانایی تشخیص دست‌کاری را دارد، بلکه می‌تواند فیلدی از برچسب که مورد دست‌کاری قرار گرفته است، شناسایی کند.



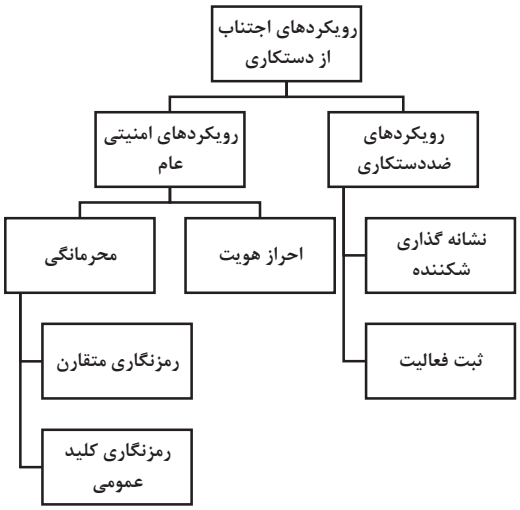
شکل ۲ . فلوچارت حمله برای اخلال در لایه فیزیکی [۶].

[۴] Kill Command
[۵] Header
[۶] Field
[۷] Object Class
[۸] Advanced Encryption Standard
[۹] Collision
[۱۰] Parity Check
[۱۱] Hash Function
[۱۲] Choas

نشانه‌گذاری [۱۳]

نشانه‌گذاری، روشی است که در آن از علامت‌گذاری‌های اضافی[۱۴] برای افزودن اســتحکام[۱۵] به داده‌ی محرمانه اســتفاده می‌شــود. ایده‌ی اصــلی در نشــانه‌گذاری آن اســت که ســیگنال یا علامتی موسوم به نشانه[۱۶] را به داده‌های میزبان بیفزاید به نحوی که علاوه بر نهان و ایمن بودن، بتوان آن را به شرط داشتن کلید رمزنگاری، به‌صورت جزئی یا کامل از کل داده موجود، بازیابی نماید. طراحی نشــانه، یک موازنه بین نامرئی بودن[۱۷] و اســتحکام اســت. هرچه نشــانه با قابلیت‌های بیشــتری باشــد، پنهان کردن آن دشــوارتر می‌گردد. روش‌های مختص به نشــانه‌گذاری متن، به پنهان‌ســازی اطلاعات در مفهوم[۱۸] و قالب[۱۹] تقســیم‌بندی می‌شــوند [۱۵]. به دلیــل محــدودیــت داده‌های RFID از روش نشــانه‌گذاری قالب اســتفاده می‌شــود؛ یعنی بخشــی از فضــای حافظه‌ی برچسب، به ذخیره‌ی نشــانه اختصاص می‌یابد که از انجام محاسبات خاصی بر سایر اجزای این حافظه به دست می‌آید [۱۶].

در شکل ۳ لزوم انتخاب نشانه‌گذاری برای آشکارسازی دست‌کاری را نشــان می‌دهد. آشــکارســازی دست‌کاری به‌منظور حفظ صحت داده‌هــای[۲۰] برچســب از تغییرات احتمــالی عمــدی و غیرعمدی می‌باشــد. از انواع نشــانه‌گذاری می‌توان نشــانه‌گذاری مستحکم[۲۱]، نشــانه‌گذاری شــکننده و نشــانه‌گذاری نیمه شــکننده[۲۲] را نام برد [۱۷].

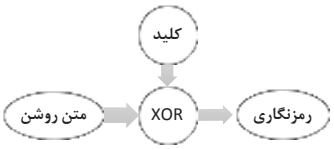


شکل ۳ . فلوچارت رویکردهای اجتناب از دست‌کاری [۶].

رویکرد پیشنهادی

این مقاله به رفع آسیب‌پذیری لایه فیزیکی RFID پرداخته است. لازم به ذکر است که عامل تعداد کم بیت‌های حافظه (۹۶ بیت) و دودویــی بــودن محتوی آن، بــه‌کــارگیری روش‌هــای متعــارف چکیده‌ســازی پیام و نشــانه‌گذاری را غیرممکن می‌کند. از این رو بــایــد از رویکردهــایی که قابل اعمال بر داده‌های دودویی با حجم کم می‌باشــند، اســتفاده نمود. روش نشــانه‌گذاری مبتنی بر شبکه عصبی به‌منظور تشخیص دست‌کاری پیشنهاد گردیده که یک گام نوین در امنیت RFID می‌باشد.

کلود شــانون[۲۳] اثبات کرد در رمزنگاری ورنام[۲۴]، اگر کلید تصادفی باشــد و از یک کلید دو بار اســتفاده نشــود، این رمزنگاری XOR «بی‌قید و شــرط» امن خواهد بود (مطابق شــکل ۴). میزان توان محاسباتی رمزشکن هیچ تأثیری بر شکستن رمز ندارد [۱۸].



شکل ۴ . رمزنگاری ورنام.

در روش پیشــنهادی از شــبکه عصــبی اســتفاده شده است که هر نرون آن نقش رمزنگاری قفل یکبار مصــرف[۲۵] را دارد. خصوصیت برگشــت‌ناپذیری باعث کاربرد مفید آن برای طراحی الگوریتم‌های رمزنگاری شده است [۱۹].

شــبکه عصبی نگاشتی از بیت‌های ورودی به خروجی می‌باشد که بدون داشــتن وزن‌ها و بایاس‌ها این نگاشــت امکان‌پذیر نیســت. وزن‌ها و بایاس‌ها نقش کلید رمزنگاری را دارند [۲۰].

در الگوریتم پیشــنهادی به‌کارگیری ســاختار توزیع‌شــده و موازی شــــبکــه عصــــبی ســــرعت پردازش را افزایش می‌دهد و ویژگی درهم‌پیچیدگی و پخش‌کنندگی شــبکه عصــبی موجب برهم زدن الگوی کد می‌شــود. این الگوریتم جهت تشــخیص دســت‌کاری در سیستم شناسایی برچسب پیاده‌سازی می‌گردد.

نشــانه‌گذاری در هنگام برنامه‌ریزی برچســب صــورت می‌گیرد و پردازشــی بر روی برچسب ایجاد نمی‌کند. پروتکل‌های سبک وزن RFID باید عملیات ســاده داشــته باشــند. بدین منظور از عملیات ساده زیر برای پیاده‌سازی الگوریتم بهره برده شده است:

---

[۱۳] Watermarking

[۱۴] Additional Notation

[۱۵] Robustness

[۱۶] Watermark

[۱۷] Imperceptibility

[۱۸] Semantic

[۱۹] Format

[۲۰] Data Integrity

[۲۱] Roubust Watermarking

[۲۲] Semi-Fragile Watermarking

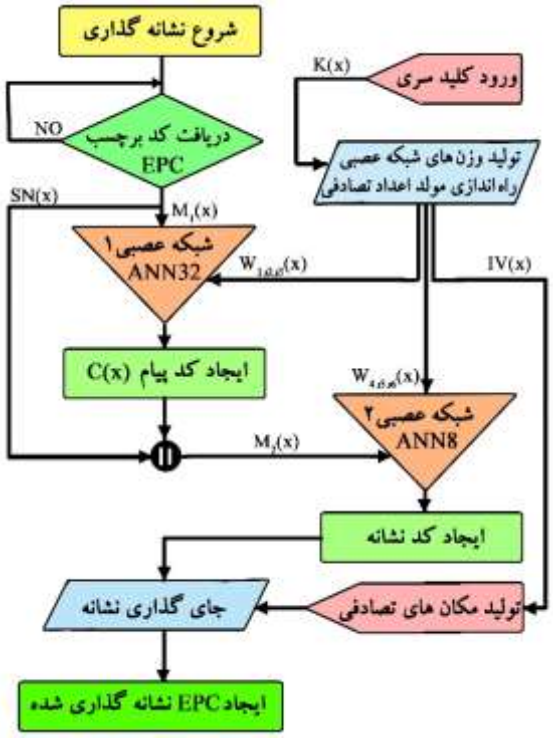
[۲۳] Claude Shannon

[۲۴] Vernam

[۲۵] One Time Pad

- عملیات ساده جمع پیمانه‌ای، XOR، AND، OR، ROT
- استفاده از مولدهای شبه تصادفی

پیش از تفضیل الگوریتم پیشنهادی، باید توجه داشت هر روش نشانه‌گذاری از دو بخش «طراحی کد نشانه» و «طراحی روش تعبیه کد در داده» تشکیل شده است. الگوریتم پیشنهادی پس از یک فرآیند چهار مرحله‌ای، هشت بیت نشانه را تولید می‌کند و در مکان بیتی نامشخص در میدان شماره‌سریال داده‌ی برچسب قرار می‌دهد. فرآیند الگوریتم مطابق شکل ۵ می‌باشد. این الگوریتم با داشتن کد EPC تولیدکننده کالا و کلید سری شروع به تولید نشانه می‌کند.



شکل ۵ . فلوچارت رویکرد پیشنهادی نشانه‌گذاری.

جزئیات چگونگی تولید نشانه در شکل ۷ آمده است.

مرحله اول: پس از دریافت دنباله بیت‌های کد EPC یک رشته بیتی به نام $M_1(x)$ از EPC ایجاد می‌گردد. $M_1(x)$ ورودی شبکه عصبی اول می‌باشد. بلوک تولید کننده‌ی وزن‌های شبکه عصبی با استفاده از کلید سری، اوزان $W_{1,2,3}(x)$ را ایجاد می‌کند. با اعمال $M_1(x)$ و $W_{1,2,3}(x)$ به شبکه عصبی اول، کد $C(x)$ ایجاد می‌گردد. چگونگی تولید رشته‌های بیتی در ادامه توضیح داده می‌شود.

مرحله دوم: در این مرحله با عمل الحاق بیتی شماره‌سریال EPC و $C(x)$ ایجاد شده از مرحله قبل، پیام $M_2(x)$ ایجاد می‌شود و با در نظر گرفتن اوزان $W_{4,5,6}(x)$ و پیام $M_2(x)$ شبکه عصبی دوم، بیت‌های نشانه را ایجاد می‌کند.

مرحله سوم: شامل عمل جایگذاری شبه تصادفی نشانه در میدان شماره‌سریال می‌باشد. مولد مکان‌های تصادفی[۲۶] با بردار آغازین[۲۷] ($IV(x)$) راه‌اندازی می‌شود و هشت جایگاه تصادفی برای بیت‌های نشانه، تخصیص داده می‌شود.

مرحله چهارم: ۲۸ بیت چکیده پیام ایجاد شده را با شماره سریال XOR می‌کنیم. این کار برای جلوگیری از حملات متن اصلی منتخب[۲۸] صورت می‌گیرد (یعنی حالتی که مهاجم کد EPC را در اختیار داشته باشد). در نهایت کد نشانه در جایگاه تصادفی قرار داده و بدین ترتیب برچسب نشانه‌گذاری شده ایجاد می‌گردد.

لازم به ذکر است که سازنده کالا کد شماره سریال را به‌صورت کد ۹۶ بیتی ایجاد می‌کند که ۸ بیت آن جهت نشانه‌گذاری ذخیره شده است.

در ادامه به شرح جزئیات مراحل ذکرشده، مطابق شکل ۷ پرداخته می‌شود. با استفاده از داده برچسب که بر اساس جدول ۱ است، چندجمله‌ای $M_1(x)$ مطابق الحاق بیتی زیر به دست می‌آید:

$$M_1(x)=\text{Header} \parallel \text{EPC Manager} \parallel \text{Object Class} \qquad (۱)$$

در جانشینی بیتی ذکر شده با در اختیار داشتن ۸ بیت سرآیند[۲۹]، مدیریت EPC با ۲۸ بیت و کلاس شئ با ۲۴ بیت، رشته $M_1(x)$ با طول ۶۰ بیت تشکیل می‌گردد. لازم به ذکر است، عملگر $\parallel$ در رابطه (۱) بیانگر عملگر الحاق رشته بیتی می‌باشد.

چگونگی تولید $W_1(x)$ و $W_2(x)$ در بخش شبکه عصبی توضیح داده می‌شود. در مرحله بعد از اعمال چندجمله‌ای $M_1(x)$ ایجادشده به شبکه عصبی اول کد پیام $C(x)$ ایجاد می‌گردد. از الحاق بیتی $C(x)$ ایجادشده از شبکه عصبی اول و شماره‌سریال برچسب، $M_2(x)$ ایجاد می‌گردد. $M_2(x)$ با استفاده از عمل جانشینی زیر به دست می‌آید:

$$M_2(x) = C(x) \parallel \text{Serial Number (28 bits)} \qquad (۲)$$

پس از اعمال $M_2(x)$ به شبکه دوم، هشت بیت خروجی شبکه عصبی دوم به‌عنوان نشانه در نظر گرفته می‌شود.

در این مرحله مکان بیت‌های نشانه با استفاده از مولد اعداد شبه تصادفی در میدان شماره‌سریال مشخص می‌شود. در واقع هدف، به دست آوردن مکان‌هایی است که بیت‌های نشانه در آن مخفی می‌گردند. با استفاده از رابطه (۳) هشت مکان یا عدد تصادفی بین صفر تا ۳۶ تولید می‌شود [۲۱].

[۲۶] Pseudo Random Number Generator

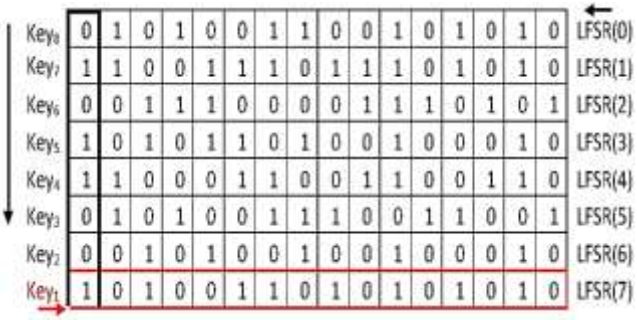
[۲۷] Initialization Vector

[۲۸] Chosen Plaintext Attack

[۲۹] Header

$$X_i = \text{key}_i^2 mod\ N \qquad (۳)$$

عملکرد این تابع شبیه مولد عدد شبه تصادفی BBS[۳۰] می‌باشد. X_i مکان تصادفی، Key_i معادل کد دهدهی کلید فرعی است و N برابر ۳۶ می‌باشد. ۸ عدد تصادفی تولید شده توسط مولد شبه تصادفی، معادل مکان‌های نشانه در ۳۶ بیت شماره‌سریال می‌باشند. مقادیر تصادفی را تنها کسی که از مقدار K(x) اطلاع دارد، می‌تواند تولید کند. Key_i مانند کلید فرعی هستند که از کلید اصلی مشتق شده‌اند. کلید اصلی را به‌صورت ستونی در یک جدول ۸ در ۱۶ جایگذاری می‌شود. هر سطر به اندازه شماره سطر آن منهای یک، شیفت چپ چرخشی[۳۱] داده می‌شود. ۸ سطر جدول همان ۸ کلید (Key_i) مولد مکان تصادفی می‌باشند که از باقیمانده مجذور آن به پیمانه ۳۶ مکان‌های تصادفی به دست می‌آید. برای درک بهتر به شکل ۶ توجه کنید. نکته‌ی قابل توجه این است که ۸ عدد تولید شده، باید متفاوت باشند. در صورتی که X_iای با مقادیر قبلی تولید شده مساوی باشد، به key_i آن یک واحد اضافه می‌شود و مجذور می‌گردد و حاصل دوباره به پیمانه ۳۶ برده می‌شود. این روند ادامه داده می‌شود تا عدد به دست آمده با اعداد قبلی متفاوت باشد؛ به‌عبارت دیگر پس از هر بار تولید یک عدد، باید با تمام اعداد تولید ش÷ده‌ی قبلی مقایسه گردد.



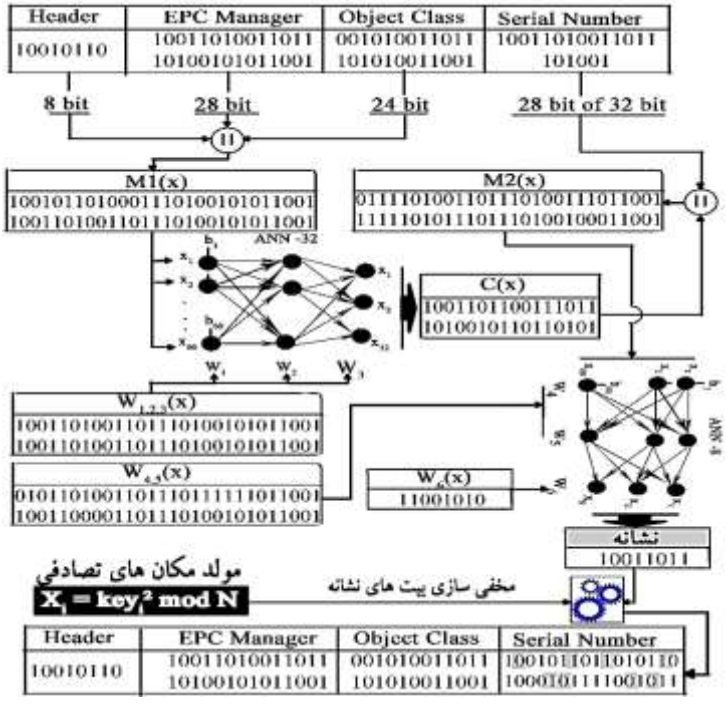
شکل ۶ . فرآیند تولید ۸ کلید مکان‌های تصادفی.

تشخیص دست‌کاری

تشخیص دست‌کاری در دو گام صورت می‌گیرد.

گام اول: استخراج نشانه

ابتدا $M_1(x)$ طبق رابطه (۱) تولید نموده و با استفاده از الگوریتم ANN32 مقدار رشته بیتی C(x) تولید می‌گردد؛ سپس با استفاده از رابطه (۳) و داشتن کلید، مکان بیت‌های نشانه تعیین می‌شود و به این ترتیب نشانه استخراج می‌گردد. حال باید نشانه‌ای که از کد EPC تولید شده با نشانه‌ی استخراج شده مقایسه شود.



شکل ۷ . فرآیند تولید نشانه (علامت ① مبین عملگر الحاق رشته بیتی است).

[۳۰] Blum, Blum, and Shub

[۳۱] Circular Left Shift

گام دوم: تولید نشانه

۸ بیت نشانه استخراج شده گام اول در حافظه‌ای ذخیره می‌گردد. حال از این کد EPC، بیت‌های نشـانه محاسـبه می‌شـوند. بدین منظور از الگوریتم تولید نشـانه توضـیح داده شـده در بخش قبل، برای ایجاد نشانه، اجرا می‌گردد. ۲۸ بیت از ۳۲ شماره‌سریال پس از جداسازی نشانه با چکیده پیام XOR شده و بعد به C(x) الحاق می‌گردد. سـپس $M_2(x)$ تولیـد و به الگوریتم دوم (ANN8) وارد می‌شـود و بیت‌های نشـانه، محاسـبه می‌گردند. اکنون بیت‌های نشانه محاسبه‌شده از کد EPC، با بیت‌های نشانه‌ی استخراج شده در گام اول، مقایسـه می‌گردند. در صـورت عدم تسـاوی آن‌ها، دسـت‌کاری تشـخیص داده می‌شـود. نمونه‌ای از عملیات شـرح داده شـده در شـکل ۸ آورده شـده اسـت. همان‌طور که در شکل مشاهده می‌شود، تعدادی از بیت‌ها براثر دست‌کاری یا خطا تغییر نموده اسـت. در گام آخر بیت‌های نشـانه‌ی اسـتخراج شـده با نشانه‌ی محاسبه شده، مقایسه می گردد و در صورت عدم تطبیق، دست‌کاری برچسب مورد تأیید است.

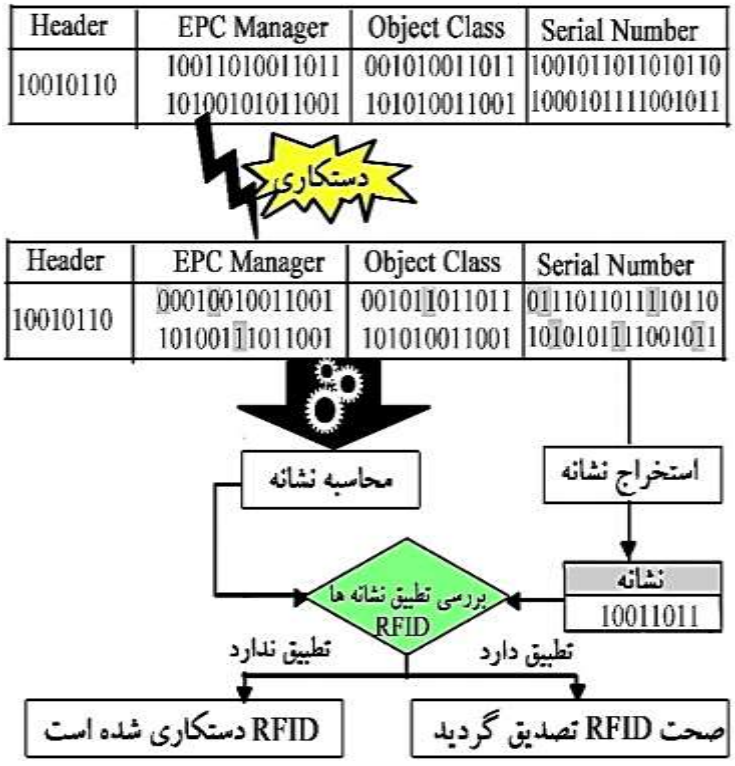
شبکه عصبی

مزیت رویکرد پیشـنهاد شـده، علاوه بر موارد یادشـده استفاده از شـبکه عصـبی اسـت. سـاختار شـبکه عصـبی، جایگشـت[۳۲] و جانشینی[۳۳] ایجاد می‌کند و به‌گونه‌ای تعلیم می‌بیند که همبستگی چکیده پیام را با کد حداقل نماید. طبق آزمون‌های گرفته‌شـده، شبکه عصبی یک تابع به‌طور کامل یک‌طرفه است.

فرآیند دو مرحله‌ای با استفاده از کلیدهای فرعی احتمال تشخیص نحوه‌ی محاسـبه‌ی نشانه را بسیار دشوار و دور از دسترس کرده و بر اسـتحکـام روش می‌افزایـد. روش تولیـد اوزان بـه‌طوری طراحی شده که ابهام و پیچیدگی را باهم داشته باشد [۲۰].

عناصـر پردازشـی در شـبکه عصبی مصنوعی طراحی شده، دارای تابع فعال‌سـاز پله[۳۴] می‌باشند که می‌تواند به‌صورت بلادرنگ عمل کنند و زمانی از پردازش را به خود اختصاص ندهد.

x_i مبین بیت ورودی شـبکه اسـت که دارای مقادیر صفر و یک می‌باشـد. x_i ها بیت‌هایی از پیام‌های $M_1(x)$ و $M_2(x)$ می‌باشـند. تمام وزن‌ها (W_{ij}) ورودی را به خروجی نگاشت می‌کنند.



شکل ۸ . فلوچارت حمله برای اخلال در انطباق، فرآیند تشخیص تغییر.

[۳۲] Permuted Choice One

[۳۳] Substitution Choice One

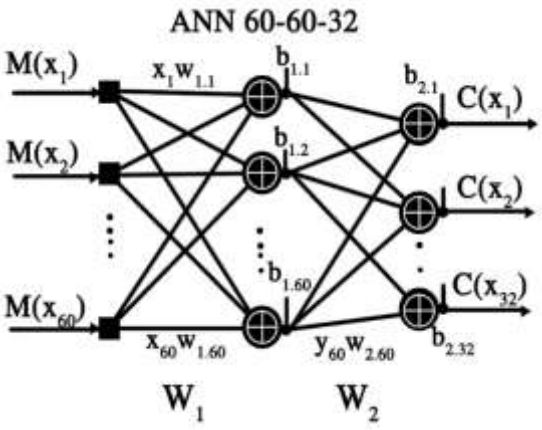
[۳۴] Step Activation Function

خروجی هر نرون از حاصل‌ضرب هر ورودی با وزن آن لینک ایجاد می‌شود که در پایان از یک تابع پله عبور می‌کند. ورودی هر تابع فعال‌ساز متغیر σ_{ji} از رابطه (۴) به دست می‌آید.

$$\sigma_{ji} = \sum_{i=1}^{N} W_{ji}, x_i \qquad (۴)$$

در رابطه (۴)، i و j به ترتیب بیانگر شماره نرون لایه و شماره لایه می‌باشند. در شکل ۹ ساختار شبکه عصبی نشان داده شده است. خروجی هر نرون τ بدین صورت محاسبه می‌گردد.

$$\tau_{ji} = step(\sigma_{ji}) \qquad (۵)$$



شکل ۹ . ساختار شبکه عصبی پیشنهادی.

در رابطه (۵)، Step بیانگر تابع پله می‌باشد که خروجی آن برای مقادیر بزرگتر مساوی از صفر، برابر یک و در غیر این صورت صفر می‌باشد. ساختار شبکه عصبی ANN-32 و ANN-8 به ترتیب ۳۲-۶۰-۶۰ و ۸-۶۰-۶۰ می‌باشند؛ یعنی ۶۰ نرون در لایه ورودی و مخفی دارد و به تناسب هر مرحله الگوریتم پیشنهادی، شبکه عصبی اول، ۳۲ و در شبکه عصبی دوم، ۸ نرون در لایه خروجی وجود دارد (مطابق ساختار شکل ۷).

مولد اولیه وزن‌ها

همان طور که می‌دانید هدف از آموزش شبکه عصبی تعیین مقدار بهینه وزن‌ها است. در این تحقیق، هدف از آموزش کاهش همبستگی ورودی و خروجی شبکه عصبی می‌باشد. وزن‌های اولیه شبکه عصبی مانند کلید فرعی هستند که از کلید اصلی K(x) استخراج می‌شوند. بیت‌های رشته کلید اصلی به‌صورت (۶) نام‌گذاری می‌شوند.

$$K(x)=\{x_1, x_2, x_3, \ldots, x_{127}, x_{128}\} \qquad (۶)$$

K(x) یک رشته با طول ۱۲۸ بیتی است که توسط سازنده برچسب به‌عنوان کلید نشانه‌گذاری به الگوریتم داده می‌شود. به‌منظور ایجاد وزن‌ها، ابتدا رشته‌های بیتی A، B، D و E طبق روابط زیر تشکیل می‌گردند:

$$A(x_\alpha) = K(x_\alpha), \alpha = 1,2,3,\ldots,60 \qquad (۷)$$

$$B(x_\alpha) = -K(x_\alpha), \alpha = 61,62,63,\ldots,120 \qquad (۸)$$

$$Z = Decimal(x_1 x_2 x_3 \ldots . x_{60}),\ x_i \in K(x) \qquad (۹)$$

$$S_1 = Z^2 mod\ 28 \qquad (۱۰)$$

$$S_{i+1} = S_i^2 mod\ 28 \qquad (۱۱)$$

$$D(x_\alpha) = SN(s_\alpha), \alpha = 1,2,3,\ldots,60 \qquad (۱۲)$$

$$E(x) = -SN(s_\alpha), \alpha = 61,62,63,\ldots,120 \qquad (۱۳)$$

با استفاده از A، B، D و E تشکیل شده وزن‌های شبکه عصبی، طبق روابط (۱۴) تا (۲۰) تولید می‌شوند.

$$W_1(x) = AB\ v\ not(A)\ D \qquad (۱۴)$$

$$W_2(x) = AB\ v\ DE \qquad (۱۵)$$

$$W_3(x) = E(x_\alpha) + K(s_\alpha) >> 5, \alpha = 1,\ldots,32 \qquad (۱۶)$$

$$W_4(x) = BE\ v\ D\ not(E) \qquad (۱۷)$$

$$W_5(x_i) = A \oplus B \oplus D \oplus E \qquad (۱۸)$$

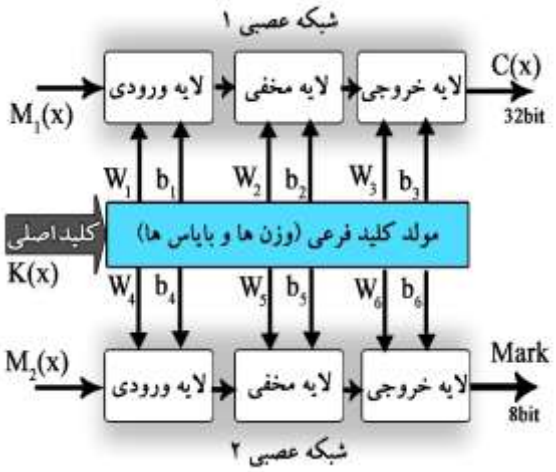
$$W_6(x_i) = B(x_j) + K(x_i) >> 5$$
$$j \in [1,8], i \in [121,128] \qquad (۱۹)$$

در رابطه‌های بیان شده از NOT(A) به‌عنوان عملگر نقیض بیتی، رابطه A v B به معنی عملگر فصل (OR) دو جمله و رابطه AB نشان دهنده عملگر عطف (AND) دو جمله A و B می‌باشند. عملگر+ نشان دهنده جمع نظیر به نظیر است. علامت‌های ⊕ و «>>» به ترتیب مبین XOR و شیفت چرخشی است. مشاهده می‌شود که به‌منظور استحکام الگوریتم از مولد تصادفی، جایگشت، جانشینی، شیفت چرخشی و توابع ابهام[۳۵] استفاده شده که موجب افزایش امنیت الگوریتم می‌گردد. تحلیل رابطه (۱۴) بدین صورت است که اگر A آنگاه B، در غیر این صورت D می‌باشد [۲۲].

تابع وزن $W_1(x)$ می‌توانست طوری تعریف شود که به جای استفاده از υ از + استفاده کند، زیرا AB و NOT(A) هرگز یک‌هایی در موقعیت بیتی یکسان نخواهد داشت. توابع $W_4(x)$ شبیه تابع $W_1(x)$ هستند، به‌طوری که آن‌ها در «توازی بیتی» کار

[۳۵] Ambiguity Function

می‌کنند تا خروجی‌شان را از بیت‌های A، B، D و E تولید نمایند. در چنین روشـــی اگر بیت‌های متناظر A، B، D و E مســـتقل و غیرمرتبط باشـــند، در این صـــورت هر بیت از وزن‌ها مســـتقل و غیرمرتبط خواهند بود. در نتیجه ارتباط بیتی نمی‌توان پیدا نمود. در این روابط ابتدا علامت هر بیت محاســـبه و در نهایت به نتیجه اعمال می‌شود. پس از ایجاد وزن‌ها تولیدی به‌صورت شکل ۱۰ به شبکه اعمال می‌شوند [۲۲].



شکل ۱۰ . مولد وزن‌های شبکه عصبی پیشنهادی.

آموزش شبکه عصبی

آموزش فرآیندی است که درنهایت منجر به یادگیری می‌شود. یادگیری شبکه زمانی انجام می‌گردد که وزن‌های ارتباطی بین لایه‌ها چنان تغییر کند که همبستگی ورودی و خروجی حداقل گردد. با دستیابی به این شرط، فرآیند یادگیری محقق شده است. این وزن‌ها بیانگر حافظه و دانش شبکه هستند. طبق قسمت قبل، وزن‌ها مقداردهی اولیه و داده‌های ورودی به آن‌ها اعمال می‌شود. همبستگی خروجی به دست آمده با ورودی محاسبه می‌گردد و اگر شرایط توقف حاکم نبود با استفاده از رابطه (۲۰) و یا (۲۱) وزن‌ها از خروجی به ورودی تغییر داده می‌شود. از یکی از این دو رابطه برای بروز رسانی وزن‌ها در فرآیند آموزش استفاده می‌شود. اگر خروجی دو نرون متصل به هم یکسان باشد ($\tau^a = \tau^b$)، وزن آن‌ها بروز می‌گردد [۲۰].

۱. رابطه اول بروز رسانی وزن نرون‌ها:

$$W_i^+ = W_i - \sigma_i x_i \theta(\tau_{ji}\tau_{(j+1)i})\theta(\tau^a\tau^b) \qquad (۲۰)$$

۲. رابطه دوم بروز رسانی وزن نرون‌ها (قدم زدن تصادفی)[۳۶]:

$$W_i^+ = W_i + x_i \theta(\tau_{ji}\tau_{(j+1)i})\theta(\tau^a\tau^b) \qquad (۲۱)$$

در روابط بالا، θ تابع ویژه‌ای می‌باشد که $\theta(a,b)=0$ اگر $a \neq b$ باشد، در غیر این صورت θ برابر یک می‌شود.

شرایط توقف آموزش از این قرار است:

- کاهش همبستگی از حد آستانه تعیین شده.
- عـدم تغییرات قـابـل توجه همبســـتگی در طی چندین دوره آموزش.
- تعداد دوره‌های آموزش از تعداد دوره‌های تعیین شـده بیشـتر گردند.
- افزایش همبستگی در چندین دوره متوالی.
- دو نرون متصل به هم با خروجی یکسان وجود نداشته باشد.

پارامترهای شرایط توقف توسط کاربر قابل تنظیم است.

تحلیل و مقایسه

این پژوهش به حفاظت از تغییر داده‌های برچسـب EPC پرداخته که پرکاربردترین برچســب RFID می‌باشـد و نیز تمام بیت‌ها را به‌منظور اطلاع از دسـت‌کاری بررسـی می‌کند. این در حالی است که رویکرد پیشنهادی نسبت به روش‌های پیشین، تعداد بیت‌های کمتر یا مسـاوی به نشـانه اختصاص می‌دهد و محل نشانه‌گذاری نیز پنهان می‌باشـد. این رویکرد علاوه بر عدم نیاز به محرمانگی الگوریتم نشانه‌گذاری، مزایای دیگری از قبیل سهولت پیاده‌سازی و سـرعـت عملکرد را نیز داراسـت. امروزه در کارهای تجارتی محرمانه نگاه داشتن الگوریتم کار دشواری می‌باشد.

شـبکـه‌های عصـبی الگوریتم پیشـنهادی باعث می‌گردد که همبســتگی چکیده پیام با کد حداقل شــود و طبق آزمون‌های گرفته شـده از الگوریتم، شـبکه‌های عصـبی از توابع یک‌طرفه هســتند، به‌طوری که با داشــتن وزن‌ها و خروجی نمی‌توان به ورودی یکتایی دست یافت.

الگوریتم پیشـــنهـادی می‌توانـد در برچســـب خوان به‌صـــورت نرم‌افزاری، پیاده‌سازی شود و نیاز به سخت‌افزار اضافه نیز نخواهد داشـت. در صـورت نیاز به پیاده‌سازی سخت‌افزاری، این الگوریتم قابل پیاده‌سـازی بر روی یک میکروکنترلر می‌باشد، زیرا الگوریتم پیشـنهادی یک الگوریتم سـبک است. تعداد عملگرهای آن بسیار کمتر از روش‌های سنتی می‌باشد.

در مرجع [۱۳] در مجموع ۹ بیت برای نشـانه در نظر گرفته شده که ۸ بیت اصلی آن از خروجی تابع درهم‌سازی است. ورودی تابع

[۳۶] Random Walk

درهم‌ساز بیت‌های «مدیریت EPC» و «OC» می‌باشد. تابع درهم‌ساز بر روی سرآیند و شماره‌سریال اعمال نمی‌شود و نویسنده معتقد است که دست‌کاری بر روی شماره‌سریال خطرناک نیست. دست‌کاری شماره‌سریال کالا را بی‌ارزش می‌کند و می‌تواند باعث اختلال در کار انبارداری شود. می‌توان با تغییر شماره سریال کالای معیوب را با کالای سالم از همان نوع تعویض نمود. از اشکالات لاینفک این نوع رویکردها آن است که نشانه‌گذاری با استفاده از تابع محرمانه انجام می‌شود و نیز امنیت این طرح به مخفی ماندن تابع درهم‌سازی بستگی دارد. این فرض اصل کرکهف [۲۳] را نقض می‌کند که امنیت نباید وابسته به مخفی نگه‌داشتن الگوریتم باشد. از آنجایی که تابع الگوریتم بین افراد درگیر در تجارت RFID به اشتراک گذاشته می‌شود، بحث ایجاد اعتماد بسیار مهم می‌گردد؛ بنابراین در برابر حملات داخلی بسیار آسیب‌پذیر می‌شود.

مشکل رویکردهایی که موقعیت بیت‌های نشانه مشخص است باعث می‌شود روش تولید نشانه قابل تحلیل باشد. در نتیجه مهاجم می‌تواند به آسانی بیت‌های نشانه را متناسب با داده تغییر یافته برچسب تغییر دهد.

مرجع [۱۴] مبتنی بر آشوب، علاوه بر تشخیص دستکاری قابلیت تصحیح نیز دارد. استفاده از تابع درهم‌ساز و ایجاد هرج و مرج در توالی بیت‌ها، مهاجم نمی‌تواند به سادگی نشانه را استخراج کند و آن را دست‌کاری نماید. بدین منظور ۱۲ بیت به نشانه اختصاص داده شده که به‌طور نامنظم در شماره سریال تعبیه شده است، ولی نشانه تعداد زیادی از بیت‌ها ارزشمند EPC را به خود اختصاص داده است.

به علت استفاده از شبکه عصبی ANN-32، در کل 2^{120} حالت پیچیدگی در هرکدام از لایه ورودی اول و لایه مخفی و 2^{32} حالت پیچیدگی در خروجی ایجاد می‌گردد.

شبکه عصبی ANN-8 مانند شبکه اول است، فقط در لایه خروجی ۸ نرون دارد که 2^{8} پیچیدگی ایجاد می‌کنند. هنگامی‌که لایه‌ای از ANN، ۶۰ ورودی داشته باشد برای هر نرون باینری 2^{60} حالت به وجود می‌آید.

مزیت آشکار رویکرد ارائه شده آن است که مکان بیت‌ها نشانه قابل پیش‌بینی نیست.

ایراد کدهای CRC، ماهیت خطی آن‌ها است که آن را آسیب‌پذیر می‌نماید [۲۴]. مزیت شبکه‌های عصبی این است که ساختارهای موازی را به‌خوبی پشتیبانی می‌کنند و نیز می‌توان آن‌ها را به‌صورت نرم‌افزاری پیاده‌سازی نمود. البته باید نسبت به محرمانه ماندن کلید اصلی K(x) احتیاط شود. درنهایت این مقاله با استفاده از ساختار موازی شبکه عصبی در کاهش حجم محاسبات کاهش چشم‌گیری داشته است. در ادامه در جدول ۲ روش‌های پیشین با الگوریتم پیشنهادی مقایسه شده است.

جدول ۲. مقایسه‌ای بین روش‌های پیشین و الگوریتم پیشنهادی.

روش‌های تشخیص دست‌کاری EPC	مشخصه موردبررسی	تجهیزات لازم	مزایا	محدودیت‌ها
آشوب	کد EPC	سخت‌افزار با سرعت پردازش بالا	قابلیت اطمینان بالا و تصحیح کد	حجم بالای پردازش و تعداد زیاد بیت‌های مصرفی و گران بودن تجهیزات مورد نیاز الگوریتم
CRC	توازن	ارتقای نرم‌افزاری	پیاده‌سازی آسان	مکان بیت‌های نشانه قابل پیش‌بینی و وابستگی آن به انتخاب مناسب G(x)ها
رمزنگاری AES	کل کد EPC	ارتقای نرم‌افزاری و سخت‌افزاری	قابلیت اطمینان بالا	قابل پیاده‌سازی بر روی RFIDهای ارزان‌قیمت نمی‌باشند و نیازمند کلید است.
تراشه‌های نوین RFID	داده درون برچسب	ارتقای تراشه RFID	قابلیت اطمینان بالا	نیاز به تغییر سخت افزار برچسب و هزینه بالا برچسب
کلید عمومی و خصوصی	کد EPC با کلید در بانک داده	ارتقای نرم‌افزاری	قابلیت اطمینان بالا و قابلیت احراز هویت	نیاز به توزیع کلید، هزینه بالا و عدم کارآیی در مکان‌های توزیع شده با برچسب‌های زیاد
توابع درهم‌ساز	چکیده پیام کد EPC	ارتقای نرم‌افزاری و یا سخت‌افزاری	تشخیص آسان	مکان بیت‌های نشانه مشخص بوده و محرمانه بودن الگوریتم توابع چکیده‌ساز پیام
نشانه‌گذاری شکننده	نشانه کد EPC	ارتقای نرم‌افزاری	تشخیص آسان	عدم بررسی شماره سریال و اختصاص تعداد بیت بیشتری به نشانه و توازن
ثبت فعالیت‌ها	بررسی وقایع ثبت شده	ارتقای نرم‌افزاری و سخت افزاری	قابلیت اطمینان بالا	نیاز به اختصاص دادن حافظه به ثبت وقایع و نیازمند پروتکل خاص ثبت وقایع است.
شبکه عصبی	نشانه کد EPC	ارتقای نرم‌افزاری	تحت پوشش قرار دادن شماره سریال و اختصاص بیت کمتر	نیاز به کلید

حمله به الگوریتم پیشنهادی

علم «تحلیل رمز[۳۷]» به مطالعه روش‌های اصولی می‌پردازد که بر اساس آن‌ها می‌توان بدون در اختیار داشتن کلید رمز، داده رمزنگاری شده را از رمز خارج کرد یا کلید رمز را به دست آورد.

- ✓ تحلیل خطی نشانه: با فرض در اختیار داشتن نشانه و معادل کد تولید نشانه‌گذاری نشده آن انجام می‌گیرد. در این روش سعی می‌شود که بین زیر مجموعه‌ای از بیت‌های کد EPC، نشانه و کلید اصلی (یکی از کلیدهای فرعی) رابطه خطی پیدا شود. به دلیل تصادفی بودن مکان‌های نشانه و خاصیت غیرخطی شبکه عصبی این تحلیل امکان‌پذیر نیست. از داشتن باقیمانده یک تقسیم نمی‌توان به مقسم دست‌یافت (mod).
- ✓ تحلیل تفاضلی[۳۸] سیستم نشانه‌گذاری: در یک عبارت کلی، روش تحلیل تفاضلی بر این اصل استوار است که تغییرات بین دو کد EPC (حتی به اندازه یک بیت) چگونه بر روی نشانه خروجی تأثیر می‌گذارد و این دو نشانه چه اندازه با یکدیگر اختلاف دارند. عموماً میزان اختلاف فاصله همینگ[۳۹] دو الگوی بیتی در نظر گرفته می‌شود. به‌کارگیری ساختار توزیع‌شده و ویژگی درهم‌پیچیدگی و پخش‌کنندگی شبکه عصبی موجب برهم زدن الگوی کد می‌شود و تغییر یک بیت کد EPC در ورودی باعث دگرگونی کل کد نشانه می‌گردد. شبکه عصبی به‌طوری عمل می‌کند که کمترین همبستگی بین ورودی و خروجی ایجاد گردد.
- ✓ حمله روز تولد: در مبحث آمار و احتمالات مسئله مشهور «روز تولد» وجود دارد که می‌پرسد در یک گردهمایی به‌طور متوسط چند نفر باید حضور داشته باشند تا جشن تولد حداقل ۲ نفر آن‌ها در یک روز باشد. حال با توجه به این مسئله چه تعداد برچسب‌های RFID باید وجود داشته باشد تا نشانه آن‌ها برابر گردد. این تعداد از رابطه زیر به دست می‌آید:

$$n \approx \sqrt{2\ln(\frac{1}{1-p})}\sqrt{N} \qquad (۲۲)$$

در رابطه فوق، p مبین احتمال برابر شدن دو نشانه است. n و N به ترتیب بیانگر تعداد برچسب‌ها و N تعداد بیت‌های برچسب می‌باشد. این حمله هنگامی عملی است که محل بیت‌های نشانه معیین باشد. محل بیت‌های نشانه به‌صورت تصادفی به‌وسیله کلید اصلی تعیین می‌گردد.

نتیجه‌گیری

در این مقاله، ابتدا چالش‌های امنیتی RFID مورد بحث قرار گرفت و به ارائه راهکاری عملی جهت امنیت RFID پرداخته شد. این راهکار براساس نشانه‌گذاری با استفاده از ابزار شبکه عصبی ارائه گردیده و عیوب تشخیص دست‌کاری برچسب در «خواننده برچسب» رفع شده است. به‌طور کلی این الگوریتم با داشتن عملیات دو مرحله‌ای به علت حالت زیاد ایجاد شده در خروجی شبکه، شکست الگوریتم، دور از دسترس می‌نماید.

این تحقیق توانسته است تنها ۸ بیت را به نشانه اختصاص دهد و نیز تمام بیت‌ها EPC از دست‌کاری حفاظت کند و به علت نامشخص بودن مکان ذخیره بیت‌های نشانه، تشخیص وجود نشانه ممکن نخواهد بود. مزیت این روش تصادفی بودن مکان کد نشانه و خواص ذکر شده شبکه عصبی می‌باشد. در نهایت با انتقال پردازش به خواننده برچسب، فضای ذخیره و پردازش در برچسب را آزاد نگه دارد. لازم به ذکر است که طرح پیشنهادی می‌تواند ویژگی‌های تمامیت و بهمنی را برآورده سازد و کارآیی مطلوبی ارائه دهد.

مراجع

[1] A. Rani and K. Sheoran, "ECC Cryptography for RFID Communication", International Journal for Technological Research in Engineering, Vol. 2, pp. 2639-2641, 2015.

[2] K. Srivastava, A. K. Awasthi, S. D. Kaul and R. C. Mittal, "A Hash based Mutual RFID Tag Authentication Protocol in Telecare Medicine Information System", Journal of Meddical Systems, Vol. 39, pp. 153-160, January 2015.

[3] K.-H. Yeh, N.-W. Lo, K.-Y. Tsai, Y. Li and E. Winata, "A Novel RFID Tag Identification Protocol: Adaptive n-Resolution and k-Collision Arbitration", Wireless Personal Communications, Vol. 77, pp. 1775-1800, Aug. 2014.

[4] C. Floerkemeier and M. Lampe, "RFID Middleware Desigen Addressing Application Requirements and RFID Constraints", International Conference on Smart Object and Ambient Intelligence, France, pp. 219-224, 2005.

[5] D. K. Klair, K. W. Chin and R. Raad, "On the Suitability of Framed Slotted Aloha based RFID Anti-Collision Protocols for Use in RFID-Enhanced WSNs", In Proceedings of 16th International Conference on Computer Communications and Networks, pp. 583-590, 2007.

[6] F. Gandino, B. Montrucchio and M. Rebaudengo, "Tampering in RFID: A Survey on Risks and Defenses" Journal on Special Topics in Mobile Networks and Applications, Vol. 15, pp. 502-516, 2010.

[7] M. Feldhofer, S. Dominikus and J. Wolkerstorfer, "Strong Authentication for RFID Systems using the AES Algorithm", Workshop on Cryptographic Hardware and Embedded Systems, LNCS, Vol. 3156, pp. 357-370, Aug. 2004.

[8] A. Juels, R. L. Rivest and M. Szydlo, "The Blocker Tag: Selective Blocking of RFID Tags for Consumer Privacy", Proceedings of the 10th ACM Conference on Computer and Communications Security, ACM Press, pp. 1-9, 2003.

[9] J. Grobschadle and S. Tillich, "Design of instruction Set Extensions and Functional Units for Energy Efficient Public Key Cryptography", Workshop on RFID and Lightweight Crypto, July 2005.

[10] R. Lingle, "MIT's Economical RFID Field Probe", Packaging World, 2005.

[11] P. Bernardi, F. Gandino, F. Lamberti, B. Montrucchio, M. Rebaudengo and E. R. Sanchez, "An Anti Counterfeit

[۳۷] Cryptoanalysis

[۳۸] Differential Cryptanalysis

[۳۹] Hamming distance

Mechanism for the Application Layer in Low Cost RFID Devices", 4th European Conference on Circuits and Systems for Communications, pp. 227-231, 2008.
[12] T. Cheng and L. Jin, "Analysis and Simulation of RFID Anti Collision Algorithms", Proceedings of 9th International Conference on Advanced Communication Technology, pp. 697-701, 2007.
[13] V. Podar and E. Cheng, "Tamper Detection in RFID Tags using Fragile Watermarking" International Conference Industrial Technology, Mumbai, pp. 2846-2852, 2006.
[14] M. Q. Fan and H. X. Wang, "Tamper Discrimation in RFID Tags using Chaotic Fragile Watermark", IEEE Conference on Network Security, Wireless Communiction and Trusted Computing, Wuhan, Hubei, pp. 147-150, 2009.
[15] F. Hartung and M. Kutter, "Multimedia Watermarking Techniques", Proceedings of the IEEE, Vol. 87, No. 7, pp. 1079-1107, 1999.
[16] A. A.-A. Gutub, F. Al-Haidari, K. M. Al-Kahsah and J. Hamodi, "e-Text Watermarking Utilizing 'Kashida' Extensions in Arabic Language Electronic Writing", Journal of Emerging Technologies in Web Intellegence, Vol. 2, No. 1, pp. 48-55, 2010.
[17] H. Si and C. T. Li, "Maintaining Information Security in E-Government through Steganology", Department of Computer Science, University of Warwick, Citesser, 2008.
[18] C. E. Shannon, "Communication Theory of Secrecy Systems", Bell System Technical Journal, No. 28, pp. 656-715, 1949.
[19] N. Nagy and S. G. Akl, "One-Time Pads without Prior Encounter", Parallel Processing Letters, Vol. 20, No. 3, pp. 263-273, 2010.
[20] V. Soni, M. S. Tanwar and K. V. Prema, "Implementation of Hash Function based on Neural Cryptography", International Journal of Computer Science and Mobile Computing, Vol. 3 No. 4, pp. 1380-1386, April 2014.
[21] E. Barker, W. Barker, W. Burr, W. Polk, and M. Smid, "Recommendation for Key Management", NIST Special Publication, pp. 800-857, August 2007.
[22] M. Stevens, "Single Block Collision Attack on MD5", Cryptology ePrint Archive, Report, 2006.
[23] J. H. Hopman and B. Jacobs, "Increased Security through Open Source", Communction of the ACM, Vol. 50, No 1, pp. 79-93, 2007.
[24] R. Thompson, "RFID Technical Tutorial", Journal of Computing Sciences in Colleges, Vol 21, No. 5, pp 8-9, 2006.

جدول ۳. جدول علائم و اختصارات.

شماره سریال	SN	نشان دهنده جمع نظیر به نظیر	+
تابع پله	step	عملگر الحاق رشته بیتی	||
عملگر فصل (OR)	υ	شیفت چرخشی	>>
وزن نرون‌ها	W	بزرگتر یا کوچکتر	<>
XOR	⊕	مدیریت EPC	EM
مکان تصادفی	X	کلید	K(x)
بیت ورودی شبکه عصبی	x	رشته بیتی	M
ورودی تابع فعال‌ساز	σ	باقی مانده تقسیم	mod
خروجی هر نرون	τ	دسته برچسب	OC
تابع ویژه آموزش	θ	خروجی شبکه عصبی اول	C(x)

زیر نویس حروف		بالا نویس حروف	
بیانگر شماره نرون	i	وزن بروز رسانی شده نرون	+
بیانگر شماره لایه نرون	j	نماد دو نرون متصل به یکدیگر	a,b