دوفصل‌نامه علمی منادی امنیت فضای تولید و تبادل اطلاعات (افتا)
سال ۱۴۰۳، جلد ۲۲، شماره ۲ (۱-۱۱)
https://monadi.isc.org.ir

ارائه‌شده در بیست و یکمین کنفرانس بین‌المللی انجمن رمز ایران در دانشگاه تربیت مدرس و پژوهشگاه ارتباطات و فناوری اطلاعات، ۲۵ و ۲۶ مهرماه ۱۴۰۳

تحلیل امنیتی یک پروتکل سبک وزن انتقال مالکیت اشیا در سامانه‌های RFID★

قدسیه کریمی*، مرتضی عادلی و محمدعلی هادوی

مجتمع دانشگاهی برق و کامپیوتر، دانشگاه صنعتی مالک اشتر، ایران

اطلاعات مقاله

کلمات کلیدی:
RFID
پروتکل انتقال مالکیت اشیا
تحلیل امنیتی پروتکل
حمله افشای راز
پروتکل سبک‌وزن

نوع مقاله: پژوهشی

چکیده

با افزایش روز افزون استفاده از برچسب‌های RFID، نیاز به پروتکل‌های خاص برای مدیریت و ارتباط با این برچسب‌ها افزایش می‌یابد. در این بین، پروتکل انتقال مالکیت اشیاء که امنیت و حریم خصوصی اشیا را برای مالک جدید پس از تغییر مالکیت تأمین می‌کند، از اهمیت بسیاری برخوردار است. اخیراً یک پروتکل سبک‌وزن انتقال مالکیت اشیا در شبکه‌های RFID ارائه شده است. این پروتکل از یک تابع سبک‌وزن خطی برای تأمین امنیت خود استفاده می‌کند و طراحان این پروتکل مدعی هستند که در عین سبک‌وزن بودن، در مقابل حملات شناخته شده امن است. در این مقاله، ضعف‌های تابع استفاده شده در این پروتکل را شناسایی کرده و نشان می‌دهیم که این پروتکل در برابر حمله افشای راز آسیب‌پذیر است. همچنین نشان می‌دهیم که حداکثر با $L \times ۴$ مرتبه اجرای پروتکل (L طول کلید است)، می‌توان با شنود داده‌های دریافتی، اطلاعات لازم را برای اجرای این حمله بدست آورد و سپس کلیدهای مشترک استفاده شده در پروتکل را بازیابی نمود.

© ۱۴۰۳ انجمن رمز ایران

۱ مقدمه

با راه‌اندازی سیستم استاندارد جهانی ردیابی و استفاده از برچسب‌های کم هزینه RFID، اصطلاح «اینترنت اشیا» مطرح شد. طرح‌های اولیه صرفاً بر عملکرد متمرکز بودند و کمتر به انعطاف‌پذیری و امنیت توجه می‌کردند. پروتکل‌های امنیتی مجموعه‌ای از قوانین در رابطه با چگونگی تبادل داده بین دستگاه‌های الکترونیکی هستند که امنیت ارتباطات را تأمین می‌کنند. ویژگی‌های خاص برچسب RFID مانند آسیب‌پذیری کانال رادیویی، محدودیت‌های توان پردازشی، عملکرد محدود توان ذخیره‌سازی داده جزو مواردی است که باید در طراحی پروتکل‌های امنیتی برای این گونه سامانه‌ها مد نظر قرار گیرد.

مفهوم مالکیت در سامانه‌های RFID به این معناست که افراد حق دسترسی و بکارگیری دستگاه تحت تملک خود را داشته باشند. یک مالک و یک برچسب برای ایجاد ارتباط امن، برخی از رازها را به‌اشتراک می‌گذارند، هنگامی که مالکیت یک برچسب به یک مالک جدید منتقل می‌شود، این رازها باید به طور امن به مقادیر جدیدی که فقط در اختیار مالک جدید و برچسب RFID قرار دارد به‌روز شوند. پروتکل‌های سبک‌وزن انتقال مالکیت [1]، بدون نیاز به زیرساخت‌های پیچیده یا منابع محاسباتی سنگین، با استفاده از تکنیک‌های رمزنگاری سبک و توابع ساده و ساختار داده کارآمد، تلاش می‌کنند که فرایند انتقال مالکیت، سریع و مقرون به صرفه انجام شود. اما در این میان، امنیت پروتکل نیز باید حفظ شود. یک پروتکل امن انتقال مالکیت در اینترنت اشیا باید ویژگی‌هایی مانند عدم افشای کلید، غیرقابل تشخیص بودن [2] برچسب‌های مختلف، امنیت پیشرو [3]، امنیت پسرو [4]، عدم ردیابی [5]، مقاومت در برابر جعل

★از کمیته علمی بیست و یکمین کنفرانس بین‌المللی انجمن رمز ایران برای داوری این مقاله تشکر می‌شود.
*نویسنده مسئول.
آدرس‌های رایانامه: gh_karimi@mut.ac.ir (قدسیه کریمی)، adeli@mut.ac.ir (مرتضی عادلی)، hadavir@mut.ac.ir (محمدعلی هادوی)
© ۱۴۰۳ تمامی حقوق متعلق به انجمن رمز ایران است.

[1] Lightweight RFID Ownership Transfer Protocol
[2] Indistinguishability
[3] Forward security
[4] Bachward security
[5] Non-Traceability

برچسب، مقاومت در برابر حمله انکار سرویس و حفظ حریم خصوصی مالک را در نظر داشته باشد.

پروتکل‌های سبک‌وزن به دلیل استفاده از توابع رمزنگاری ساده همواره مورد تهدید و آسیب قرار دارند. اخیراً یک پروتکل سبک‌وزن انتقال مالکیت توسط بی و همکاران [۱] ارائه شده است. این پروتکل از یک تابع خطی ساده به عنوان الگوریتم رمز و تأمین‌کننده امنیت ارسال اطلاعات بین کاربران پروتکل استفاده می‌کند. طراحان این پروتکل کارایی و امنیت این پروتکل را بررسی کرده و ادعا کرده‌اند که این پروتکل در برابر حملات شناخته شده مقاوم است. در این مقاله نشان می‌دهیم که این پروتکل در برابر حمله افشای راز آسیب‌پذیر است و می‌توان کلیدهای ذخیره شده در برچسب و برچسب‌خوان را با استفاده از تعداد محدودی پیام جابجا شده بین دو طرف ارتباط، بازیابی کرد.

ادامه این مقاله به این ترتیب سازماندهی شده است: در بخش ۲ کارهای مشابه که در این زمینه انجام شده است را بررسی می‌کنیم. در بخش ۳ پروتکل بی و همکاران را به صورت مختصر معرفی می‌کنیم. بخش ۴ به شیوه اجرای حمله افشای راز که روی این پروتکل انجام شده است اختصاص دارد. در بخش ۵، نتایج بدست آمده از این کار تحقیقاتی را ارائه می‌دهیم.

۲ کارهای انجام شده

برچسب‌های RFID (بازشناسی با امواج رادیویی) ریزتراشه‌های کوچک و ارزان قیمتی هستند که قادرند شناسه‌های منحصربه‌فرد را به صورت بی‌سیم و در فاصله کوتاهی ارسال کنند. ایجاد پروتکل امن برای برچسب‌های ارزان قیمت RFID با توانایی و منابع محدود همواره چالشی بزرگ بوده است. پروتکل‌های مدیریت ارتباط اشیا در شبکه IoT عموماً شامل مکانیزم‌هایی برای احراز هویت، کنترل دسترسی، انتقال مالکیت و قابلیت حسابرسی برای اطمینان از یکپارچگی و امنیت فرآیند انتقال می‌باشند.

اوزاکا و همکاران [۲] براساس تابع چکیده‌ساز و سیستم رمزنگاری کلید متقارن، روش انتقال مالکیت یک به یک (یک برچسب‌خوان و یک برچسب) را پیشنهاد کردند. جاپینن و هامالاینن [۳] ثابت کردند که در طرح اوزاکا احتمال دستکاری پیام به‌روزرسانی کلید توسط مهاجمان و در نتیجه حمله ناهمگام‌سازی[1] وجود دارد. همچنین عدم تأمین امنیت پیشرو از دیگر ضعف‌های طرح اوزاکا بود.

در سال ۲۰۰۴ مفهومی به نام اثبات گروهی[2] مطرح شد که در زمان اسکن همزمان چند برچسب از وقوع مشکلاتی مانند اشتباه در تطابق داده‌های RFID و یا از دست دادن داده‌ها در هنگام انتقال و پردازش آن‌ها جلوگیری شود [۴]. زو [۵] برای اولین بار دو الزام انتقال مالکیت و اثبات گروهی را در هم ادغام کرده و پروتکلی برای انتقال مالکیت گروهی[3] (GOT) معرفی کرد که برای انتقال همزمان مالکیت همه برچسب‌ها در یک گروه استفاده می‌شد. این پروتکل مشکل مالکیت دوگانه برچسب را حل کرده و حریم خصوصی را حفظ می‌کرد. ولی جنتی و فلاحتی [۶] ثابت کردند این پروتکل در برابر حمله ناهمگام‌سازی و جعل برچسب آسیب‌پذیر است. آنها نشان دادند مالک فعلی با داشتن تعدادی از مقادیر می‌تواند در ارتباط بین برچسب و مالک جدید اختلال ایجاد کند و باعث شود که یک برچسب معتبر به عنوان یک برچسب غیرقانونی تلقی شود. همچنین به دلیل اینکه در انتقال مالکیت گروهی، گروه‌های برچسب‌ها دارای کلید گروهی مشترک هستند و کلید برچسب‌ها به هم مرتبط است، مالک فعلی با تبانی با یکی از برچسب‌ها می‌تواند به کلید مخفی سایر برچسب‌ها دسترسی پیدا کرده و در نتیجه امکان جعل برچسب فراهم می‌شود.

کولسنگ و همکاران [۷] ابتدا یک پروتکل احراز هویت طراحی کرده و سپس آن را گسترش دادند تا شامل انتقال مالکیت نیز باشد. از آنجا که توابع چکیده‌ساز رمزنگاری به تعداد گیت‌های زیادی نیاز دارند آنها در هر دو پروتکل خود از PUF و LFSR استفاده کردند تا تعداد گیت‌های استفاد شده به میزان قابل توجهی کاسته شده و پروتکل کارآمدتر باشد. ولی صفخانی و همکاران [۸] در پروتکل انتقال مالکیت، با بهره‌گیری از ضعف تابعی که بر پایه LFSR ساخته شده بود، تابع معکوس آن را طراحی و مقدار محرمانه پین را که حتی برای مالکان برچسب مخفی بود بازیابی کردند. کارداس و همکاران [۹] نیز با طراحی دو حمله مسدودسازی پیام[4] و تزریق پیام باعث ناهمگام‌سازی در پروتکل انتقال مالکیت کولسنگ شدند.

طرح‌های جدیدتر با استفاده از فناوری‌های جدیدتر نظیر فضای ابری و زنجیره بلوکی و همچنین توابع امن‌تر سعی در رفع مشکلات قبلی دارند. به عنوان مثال لی و همکاران [۱۰] یک پروتکل انتقال مالکیت گروهی را پیشنهاد کردند که به دلیل استفاده از محاسبات ابری، رمزگذاری همریختی[5] و باقیمانده‌های درجه دوم[6]، پردازش داده‌ها به طور مؤثر انجام می‌شود و هزینه ارتباط بسیار پایین است. آن‌ها مدعی هستند تمام الزامات امنیتی مهم را برآورده می‌کنند ولی معظمی و صفخانی [۱۱] حملات افشای راز[7]، جعل برچسب‌خوان، حمله ردیابی، نقض امنیت پیشرو و حمله ناهمگام‌سازی را روی پروتکل لی و همکاران پیاده کردند. چینگ‌کاون و همکاران [۱۲] یک پروتکل سبک‌وزن دوبخشی طراحی کردند که داده‌های حاوی اطلاعات برچسب را در زنجیره بلوکی ذخیره می‌کند. انتقال کامل مالکیت نیازمند واکشی، رمزگشایی و رمزگذاری، و بارگذاری اطلاعات محتوای برچسب از سرور زنجیره بلوکی است. انجام چندین نقل و انتقال متوالی مالکیت، نیازمند عملیات تکراری و سنگین پردازش داده‌هاست که به‌طور جدی کارایی کل فرآیند را تحت تاثیر قرار می‌دهد و مشکلات امنیتی را به‌همراه خواهد داشت. چینگ‌کاون برای بهبود عملکرد پروتکل از یک پروکسی رمزگذاری مجدد[8] برای پردازش اطلاعات محتوای برچسب استفاده می‌کند و برای اطمینان از امنیت انتقال مالکیت از یک محاسبه تابع چکیده‌ساز سبک‌وزن بهره می‌گیرد.

[1] Desynchronization attack [2] Yoking-Proofs [3] Group Ownership Transfer

[4] Blocking Attack [5] Homomorphic encryption [6] Quadratic residue [7] Secret disclosure attack [8] Proxy re-encryption

لوو و یانگ [۱۳] یک پروتکل انتقال مالکیت چند مالکی پیشنهاد کرده‌اند که برای انتقال یک یا چند برچسب، از گروهی از مالکان به سایرین، به مجوزهای تعدادی از مالکان نیاز است. در این پروتکل افزایش برچسب‌ها یا مالکان به طور قابل ملاحظه‌ای تعداد پیام‌ها و محاسبات را افزایش نمی‌دهد. آن‌ها با مقایسه‌ای بر روی انواع پروتکل‌های انتقال مالکیت و یافتن نقاط ضعف مشترک بین اکثر پروتکل‌ها مدعی شدند با رمزنگاری داده‌ها، استفاده از اعداد تصادفی و وابسته نبودن کلیدها به هم، مانع از حملاتی چون حمله تکرار[1]، ناهمگام‌سازی، جعل هویت برچسب یا مالک و سایر حملات شده‌اند.

گونارسون و گرمان [۱۴] مشکل انتقال مالکیت را تحلیل می‌کنند و یک پروتکل جدید انتقال مالکیت در زیرساخت اینترنت اشیا را برای کلیدهای متقارن پیشنهاد می‌کنند که بر اساس استفاده از یک شخص ثالث مورد اعتماد کار می‌کند. پروتکل آنها نیز سبک‌وزن است زیرا فقط از عملیات کلید متقارن استفاده می‌کند. یانگ و همکاران [۱۵] روشی مناسب برای انتقال همزمان مالکیت تعداد زیادی از کالاها در سیستم‌های چند مالکی را پیشنهاد کرده‌اند. روش آن‌ها به شخص ثالث قابل اعتماد نیاز ندارد و در برچسب‌خوان‌های سیار می‌تواند مالکیت گروهی چندین مالک با چندین برچسب را به طور امن و کارآمد منتقل کند. این روش شامل احراز هویت متقابل بین برچسب‌ها، برچسب‌خوان‌ها و سرورهای مرکزی است و تضمین می‌کند که فقط مالکان اختصاص داده شده می‌توانند مالکیت برچسب را به دست آورند. یانگ و همکاران مدعی هستند روش آن‌ها می‌تواند از حملاتی از سوی صاحبان اصلی متقلب جلوگیری کند و با تعداد پیام‌های ارسالی کمتر، در برابر اکثر حملات شناخته شده مانند حملات افشای مخفی و حملات تکراری مقاومت کند.

۳ بررسی پروتکل بی و همکاران

پروتکل‌های بسیاری در زمینه انتقال مالکیت طراحی شده‌اند. با این‌حال، حملات مختلفی که روی این پروتکل‌ها اعمال شده نشان می‌دهد به دلیل استفاده از توابع خطی در پروتکل‌های سبک وزن و داشتن محدودیت در توان پردازشی و محدودیت حافظه در دستگاه‌های اینترنت اشیا، این پروتکل‌ها نمی‌توانند تمام نیازهای امنیتی را فراهم کنند.

در این مقاله، پروتکل سبک‌وزن جدیدی که بی و همکارانش [۱] برای انتقال مالکیت اشیا RFID طراحی کرده‌اند تحلیل و بررسی شده و حملات شناخته شده‌ای روی آن پیاده‌سازی می‌شود تا ادعای آنها در امن بودن کامل پروتکل به چالش کشیده شود.

۱.۳ مدل سیستم

در پروتکل بی و همکاران سه موجودیت وجود دارد: مالک قدیمی، مالک جدید و برچسب. فرض شده که کانال بین سرور پشتیبان و برچسب‌خوان‌ها به دلیل استفاده از الگوریتم‌های رمزگذاری سنتی در ارتباط، امن و قابل اعتماد است. در عین حال، برچسب‌خوان‌ها و برچسب‌ها از ارتباطات بی‌سیم برای تعامل داده‌ها استفاده می‌کنند که به شدت در برابر حملات آسیب‌پذیرند. مالک جدید به کلید برچسب دسترسی ندارد. این کلید که در برچسب ذخیره شده از مالکیت مالک قدیمی محافظت می‌کند. مجاز بودن مالکیت مالک جدید توسط مالک قدیمی بررسی می‌شود.

در این پروتکل از سه عملیات بیتی XOR، تابع چرخش $\mathrm{Rot}(\cdot,\cdot)$، و تابع جایگشت f_x استفاده شده است که سربار محاسباتی را تا حد زیادی کاهش می‌دهد. فرض کنید دنباله‌های X و Y به صورت زیر باشند:

$$X = x_1 x_2 \cdots x_n, \quad x_i \in \{0,1\},\ i = 1,2,\ldots,n$$
$$Y = y_1 y_2 \cdots y_n, \quad y_i \in \{0,1\},\ i = 1,2,\ldots,n$$

تعریف ۱ (تابع $\mathrm{Rot}(\cdot,\cdot)$). تابع چرخش $\mathrm{Rot}(X,Y)$ ورودی X را به اندازه $wt(Y) \pmod L$ به سمت چپ چرخش می‌دهد که در آن $wt(Y)$ وزن همینگ Y و L طول رشته Y می‌باشد.

تعریف ۲ (تابع جایگشت $f_x(X,Y)$). تابع جایگشت $f_x(X,Y)$ یک جایگشت از ورودی X را بر حسب کلید Y محاسبه می‌کند. این تابع این گونه عمل می‌کند که ابتدا بیت‌های ۱ از رشته Y به ترتیب شماره‌گذاری می‌شوند. سپس در ادامه بیت‌های ۰ آن شماره‌گذاری می‌شوند به طوری که:

$$y_{k_1} = y_{k_2} = y_{k_3} = \cdots = y_{k_m} = 1$$
$$y_{k_{m+1}} = y_{k_{m+2}} = y_{k_{m+3}} = \cdots = y_{k_n} = 0$$

واضح است که

$$1 \le k_1 < k_2 < \cdots < k_m \le n$$
$$1 \le k_{m+1} < k_{m+2} < \cdots < k_n \le n$$

تابع f_x جایگشت ورودی X برحسب Y را به صورت زیر اعمال می‌کند:

$$f_x(X,Y) = Z$$
$$Z = x_{k_m} x_{k_{m-1}} \cdots x_{k_1} x_{k_n} x_{k_{n-1}} \cdots x_{k_{m+1}} \quad (۱)$$

عملکرد این تابع در شکل ۱ نشان داده شده است.

تابع f_x دارای ویژگی خطی است و رابطه (۲) برای آن صدق می‌کند.

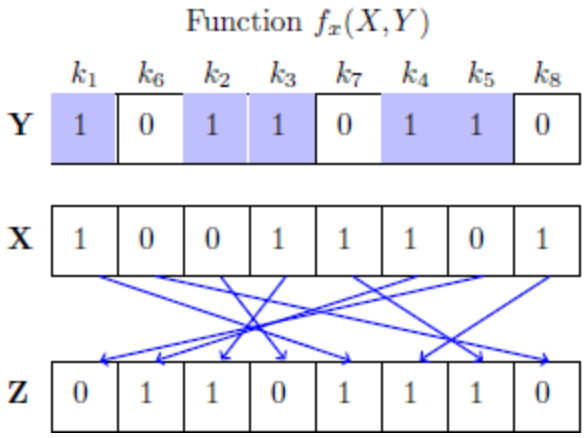
$$f_x(X,Y) \oplus f_x(Z,Y) = f_x(X \oplus Z, Y) \quad (۲)$$

برای اثبات رابطه (۲) عملگر $\oplus$ روی متن‌های رمز شده اعمال می‌شود:

$$f_x(X,Y) = x_{k_m} x_{k_{m-1}} \cdots x_{k_1} x_{k_n} x_{k_{n-1}} \cdots x_{k_{m+1}} \quad (۳)$$
$$f_x(Z,Y) = z_{k_m} z_{k_{m-1}} \cdots z_{k_1} z_{k_n} z_{k_{n-1}} \cdots z_{k_{m+1}} \quad (۴)$$

$$f_x(X,Y) \oplus f_x(Z,Y) = (x_{k_m} \oplus z_{k_m}) \| (x_{k_{m-1}} \oplus z_{k_{m-1}}) \| \cdots \| (x_{k_1} \oplus z_{k_1}) \| (x_{k_n} \oplus z_{k_n}) \| \cdots$$

[1] Replay attack



شکل ۱. عملکرد تابع $f_x(X,Y)$

$$\|(x_{k_{m+1}} \oplus z_{k_{m+1}}) \quad (۵)$$

در ضمن رابطه زیر برقرار است:

$$f_x(X \oplus Z, Y) = (x \oplus z)_{k_m} \|(x \oplus z)_{k_{m-1}} \| \cdots \|(x \oplus z)_{k_1} \|(x \oplus z)_{k_n} \| \cdots \|(x \oplus z)_{k_{m+1}} \quad (۶)$$

بنابراین عملگر $\oplus$ روی متن رمز شده عیناً روی متن آشکار اعمال می‌شود. ویژگی خطی در پروتکل بی و همکاران [۱] برای محافظت از مالکیت مالک قدیمی و به‌دست آوردن امضاهای تجمیعی[1] مالکان قدیمی و جدید استفاده شده است.

۲.۳ فاز مقداردهی اولیه

در مرحله مقداردهی اولیه مالک قدیمی اطلاعات مربوط به انتقال مالکیت را در جدول داده‌های شاخص سیستم RFID ثبت می‌کند، این جدول مالکان مجاز برای مالکیت برچسب‌ها را مشخص می‌کند. جدول ۱ جزئیات جدول داده‌های شاخص و جدول ۲ نمادهایی را که برای نمایش پروتکل بی و همکاران استفاده شده نشان می‌دهند.

۳.۳ احراز هویت و به‌روزرسانی

بی و همکاران پروتکل انتقال مالکیت فوق سبک خود را بر اساس ویژگی خطی موجود در تابع جایگشت f_x پیشنهاد کردند. بر اساس این ویژگی خطی، محاسبات روی داده‌های رمزگذاری شده انجام می‌شود و مالک جدید از کلیدهای خود برای محاسبه نتیجه رمزگذاری بدون اطلاع از کلیدهای مالک قدیمی استفاده می‌کند. مثلاً مالک قدیمی رابطه (۷) و مالک جدید رابطه (۸) را حساب می‌کند.

$$f_x(g(k_1), g(k_2)) \quad (۷)$$

$$f_x(K_(R_n), g(k_2)) \quad (۸)$$

[1] Aggregated signatures

جدول ۱. جدول داده‌های شاخص [۱]

محتوای شاخص	مقدار شاخص
IDS_1	$f_x(IDS_1, ID_1)$
IDS_2	$f_x(IDS_2, ID_1)$
...	...
IDS_i	$f_x(IDS_i, ID_1)$
IDS_{i+1}	$f_x(IDS_{i+1}, ID_2)$
...	...
IDS_j	$E(IDS_j, ID_3)$
...	...

جدول ۲. نمادهای استفاده شده در پروتکل بی و همکاران [۱]

نماد	شرح
R_c	مالک قدیمی
R_N	مالک جدید
K_1, K_2, K_3	کلیدهای مشترک بین برچسب و مالک قدیمی
K_i^{new}	کلیدهای مشترک به روز شده
K_T	راز مشترک موقت بین تگ و مالک جدید
K_R	راز مشترک موقت بین مالک قدیمی و جدید
PRNG()	مولد اعداد شبه تصادفی
Rot	تابع چرخش چپ
$f_x(x,y)$	تابع جایگشت
ID_o	شناسه مالک قدیمی
ID_n	شناسه مالک جدید
IDS_x	نام مستعار برچسب
$IDS_x^n ew$	نام مستعار به روز شده برای برچسب
r_i	اعداد تصادفی تولید شده توسط برچسب‌خوان‌ها
$hw(x)$	وزن همینگ

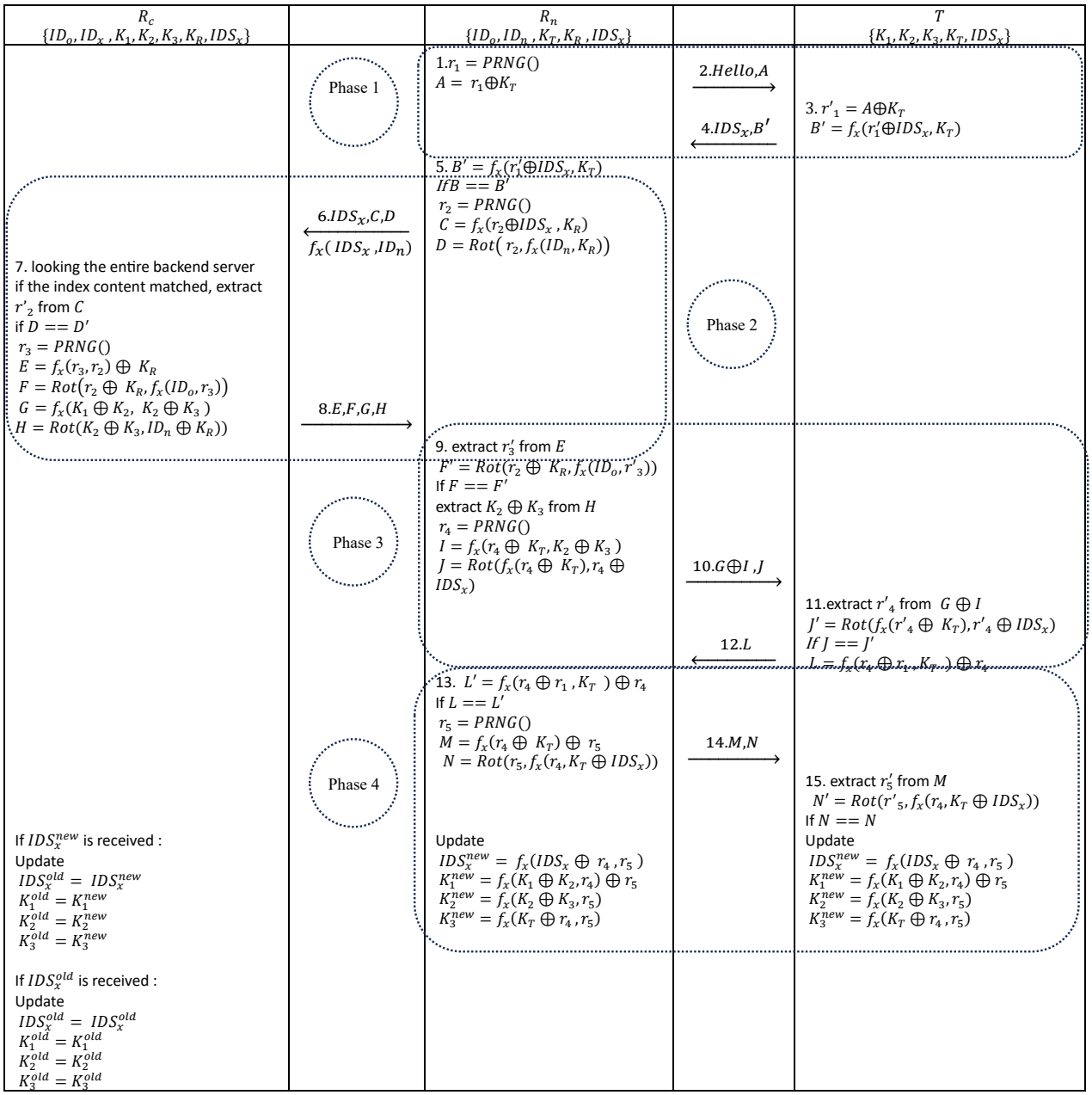
مالک جدید می‌تواند مقدار $f_x(g(k_1), g(k_2))$ را با رابطه (۹) به دست آورد بدون اینکه از مقدار k_1 و k_2 مطلع باشد.

$$f_x(K_(R_n), g(k_2)) \oplus f_x(g(k_1), g(k_2)) = f_x(K_{R_n} \oplus g(k_1), g(k_2)) \quad (۹)$$

مراحل اجرای پروتکل به صورتی که در شکل ۲ دیده می‌شود به این شرح است:

مرحله ۱و۲: مالک جدید یک عدد تصادفی تولید کرده و با کلید مشترک بین خود و برچسب، A را با رابطه (۱۰) محاسبه کرده و به همراه پیام Hello به نشانه شروع پروتکل برای برچسب ارسال می‌کند.

$$A = r_1 \oplus K_T \quad (۱۰)$$



شکل ۲. شرح پروتکل بی و همکاران [۱]

مرحله ۳ و ۴: برچسب، عدد تصادفی r'_1 را از پیام دریافتی استخراج کرده و B' را با رابطه (۱۱) محاسبه کرده و به همراه نام مستعار خود به عنوان پاسخ برای مالک جدید ارسال می‌کند.

$$B' = f_x(r'_1 \oplus IDS_x, K_T) \qquad (۱۱)$$

مرحله ۵ و ۶: مالک جدید، B را با مقادیری که از قبل دارد محاسبه کرده باشد، C و D را محاسبه کرده و همراه با نام مستعار خود و $f_x(IDS_x, ID_n)$ برای مالک قبلی ارسال می‌کند.

$$C = f_x(r_2 \oplus IDS_x, K_R) \qquad (۱۲)$$

$$D = \mathrm{Rot}(r_2, f_x(ID_n, K_R)) \qquad (۱۳)$$

مرحله ۷ و ۸: مالک قدیمی جدول شاخص را جستجو کرده و چنانچه مقادیر دریافتی و IDS_x با جدول مطابقت داشت، مالک جدید را مجاز دانسته و r'_2 را از C استخراج می‌کند و با آن D' را با رابطه (۱۳) محاسبه می‌کند. چنانچه $D = D'$ باشد پروتکل را ادامه می‌دهد. در غیر این صورت پروتکل را متوقف می‌کند. در ادامه عدد تصادفی r_3 را تولید کرده و مقادیر E, F, G, H را با رابطه‌های (۱۴)، (۱۵)، (۱۶) و

(۱۷) محاسبه کرده و برای مالک جدید ارسال می‌کند.

$$E = f_x(r_3, r_2) \oplus K_R \quad (۱۴)$$

$$F = \mathrm{Rot}(r_2 \oplus K_R, f_x(ID_o, r_3)) \quad (۱۵)$$

$$G = f_x(K_1 \oplus K_2, K_2 \oplus K_3) \quad (۱۶)$$

$$H = \mathrm{Rot}(K_2 \oplus K_3, ID_n \oplus K_R)) \quad (۱۷)$$

مرحله ۹ و۱۰: مالک جدید r'_3 را از E استخراج کرده و F' را مشابه F محاسبه می‌کند. اگر $F' = F$ بود پروتکل را ادامه داده و در غیر این صورت پروتکل را متوقف می‌کند. سپس عدد تصادفی r_4 را تولید کرده و I و J را با رابطه (۱۸) و (۱۹) محاسبه کرده و J و $G \oplus I$ را برای برچسب ارسال می‌کند.

$$I = f_x(r_4 \oplus K_T, K_2 \oplus K_3) \quad (۱۸)$$

$$J = Rot(f_x(r_4 \oplus K_T), r_4 \oplus IDS_x) \quad (۱۹)$$

مرحله ۱۱ و ۱۲: برچسب، r'_4 را از روی I استخراج کرده و با آن J' را با رابطه (۱۹) محاسبه می‌کند. اگر $J' = J$ بود پروتکل را ادامه داده و در غیر این صورت آن‌را متوقف می‌کند. سپس L را با رابطه (۲۰) محاسبه کرده و برای مالک جدید می‌فرستد.

$$L = f_x(r_4 \oplus r_1, K_T) \oplus r_4 \quad (۲۰)$$

مرحله ۱۳ و ۱۴: مالک جدید با $r_4$۴ و r_1 خودش L' را می‌سازد. اگر $L' = L$ بود ادامه می‌دهد. در غیر این صورت پروتکل را متوقف می‌کند. سپس با تولید عدد تصادفی r_5 مقادیر M و N را محاسبه کرده و برای برچسب ارسال می‌کند.

$$M = f_x(r_4 \oplus K_T) \oplus r_5 \quad (۲۱)$$

$$N = Rot(r_5, f_x(r_4, K_T \oplus IDS_x)) \quad (۲۲)$$

مرحله ۱۵: برچسب، مقدار r_5 را از M استخراج کرده و N' را با رابطه (۲۲) محاسبه می‌کند. اگر $N = N'$، مقادیر کلیدهای جدید بین مالک جدید و برچسب و نام مستعار برچسب را به‌روزرسانی می‌کند. در غیر این‌صورت، پروتکل را متوقف می‌کند و انتقال مالکیت ناتمام می‌ماند.

مالک جدید نیز مقادیر کلیدهای جدید بین خود و برچسب و نیز نام مستعار برچسب را به‌روزرسانی می‌کند.

$$IDS_x^{new} = f_x(IDS_x \oplus r_4, r_5) \quad (۲۳)$$

$$K_1^{new} = f_x(K_1 \oplus K_2, r_4) \oplus r_5 \quad (۲۴)$$

$$K_2^{new} = f_x(K_2 \oplus K_3, r_5) \quad (۲۵)$$

$$K_3^{new} = f_x(K_T \oplus r_(4), r_5) \quad (۲۶)$$

۴ تحلیل امنیت پروتکل بی و همکاران

بی و همکاران [۱] مدعی هستند در پروتکل پیشنهادی آنها محرمانگی و احراز هویت به درستی انجام شده و پروتکل در برابر انواع حملات تکرار، غیرهمزمانی و قابلیت ردیابی مقاوم است و می‌تواند از حریم خصوصی مالک جدید و قدیم محافظت کرده و امنیت پیشرو و امنیت پسرو را تأمین کند.

در این بخش، ابتدا تابع جایگشتی f_x را مورد تحلیل قرار داده و آسیب‌پذیری آن را مشخص می‌کنیم. در ادامه با استفاده از این آسیب‌پذیری، نشان می‌دهیم که این پروتکل در برابر حمله افشای راز آسیب پذیر است.

۱.۴ بررسی ضعف‌های تابع f_x

ابتدا ضعف‌های تابع $f_x(X, Y)$ که به عنوان یک الگوریتم رمز سبک‌وزن در پروتکل بی و همکاران استفاده شده است را بررسی می‌کنیم. فرض کنید دنباله‌های n بیتی X و Y به صورت زیر باشد:

$$X = x_1 x_2 \ldots x_n$$

$$Y = y_1 y_2 \ldots y_n$$

مشکلات تابع $f_x(X, Y)$ به قرار زیر است:

(۱) تابع f_x وزن همینگ رشته ورودی را تغییر نمی‌دهد. می‌توان از قابلیت برابری وزن همینگ کلید و رشته ورودی برای حدس زدن رشته ورودی استفاده کرد.

(۲) با ایجاد تفاضل در یک بیت از رشته ورودی، می‌توان تغییر را در رشته خروجی ملاحظه کرد و بر اساس مکان این بیت، مقدار بیت کلید را تشخیص داد. بنابراین با مقایسه رشته‌های ورودی و خروجی تابع f_x نتایج زیر حاصل می‌شود:

(i) چنانچه بیت اول کلید (با ارزش‌ترین بیت) دارای مقدار ۰ باشد، اولین بیت از رشته ورودی حتماً در انتهای رشته خروجی قرار می‌گیرد.

$$y_1 = 0 \leftrightarrow z_n = x_1$$

$$Z = z_1 z_2 \cdots x_1$$

(ii) چنانچه بیت اول کلید (با ارزش‌ترین بیت) دارای مقدار ۱ باشد، اولین بیت از رشته ورودی در اندیس وزن همینگ کلید قرار می‌گیرد:

$$y_1 = 1 \leftrightarrow z_h w(Y) = x_1$$

$$Z = z_1 z_2 \cdots z_{hw(Y)} \cdots z_n = z_1 z_2 \cdots x_1 \cdots z_n$$

(iii) چنانچه بیت آخر کلید (کم ارزش‌ترین بیت) دارای مقدار ۰ باشد، آخرین بیت از رشته ورودی در اندیس وزن همینگ به علاوه یک کلید قرار می گیرد:

$$y_n = 0 \leftrightarrow z_{hw(Y)+1} = x_n$$

$$Z = z_1 z_2 \cdots z_{hw(Y)+1} \cdots z_n = z_1 z_2 \cdots x_n \cdots z_n$$

(iv) چنانچه بیت آخر کلید (کم ارزش‌ترین بیت) دارای مقدار ۱ باشد، آخرین بیت از رشته ورودی در ابتدای رشته خروجی قرار می‌گیرد:

$$y_n = ۰ \leftrightarrow z_۱ = x_n$$
$$Z = z_۱ z_۲ \cdots z_n = x_n \cdots$$

به همین ترتیب می‌توان مکان تک‌تک بیت‌های رشته ورودی را در رشته خروجی به‌دست آورد. وقتی iاُمین بیت در X تغییر کند، jاُمین بیت در Z تغییر می‌کند. بنابراین ترتیب k_iها مشخص می‌شود.

(۳) چنانچه رشته ورودی X فقط شامل بیت‌های ۱ باشد خروجی تابع f_x با ورودی برابر خواهد بود و با هر کلیدی این خروجی تولید خواهد شد (حتی اگر کلید اشتباه باشد).

(۴) چنانچه کلید ترکیبی از ۰ها و ۱ها به‌صورت پشت سر هم باشد به طوری که اندیس ۰ها کمتر از اندیس ۱ها باشد (مثلاً ۰۰۰۱۱۱۱۱) دو مشکل زیر به وجود می‌آید:

- فرقی نمی‌کند که تعداد ۰ها یا ۱ها در کلید چقدر باشد. یعنی کلیدهای ۰۰۰۰۰۱۱۱، ۰۰۰۰۰۰۰۰، ۱۱۱۱۱۱۱۱ و ... همه مشابه هم عمل می‌کنند و خروجی یکسانی دارند.
- این کلیدها فقط رشته ورودی را معکوس می‌کنند و رشته ورودی به راحتی قابل بازیابی است.

مثال: برای تشخیص کلید در عبارت زیر ابتدا در بیت iاُم ورودی تفاضل ایجاد کرده و تفاضل در خروجی بررسی می‌شود تا محل هر بیت در خروجی به دست آید.

به عنوان مثال در شکل ۳، بیت اول رشته ورودی در بیت چهارم خروجی ظاهر شده، بیت دوم ورودی در بیت هشتم خروجی و به همین ترتیب محل هر بیت در خروجی مشخص است. مقادیر jها به صورت نزولی در جدول دیده می‌شود. در سمت چپ تا جایی که ترتیب نزولی jها در جدول ادامه دارد، اندیس بیت‌های ۱ در کلید محسوب می‌شود و جایی‌که مقدار j افزایش یافته ($hw(Y) = ۴$) و مجدداً نزولی می‌شود (بیت‌های باقی‌مانده کلید) بیت‌های ۰ کلید را تعیین می‌کند.

$$A[j] = ۸, ۵, ۴, ۱ \rightarrow Y_(A[j]) = ۱$$
$$A[j] = ۷, ۶, ۳, ۲ \rightarrow Y_(A[j]) = ۰$$
$$Y = ۱۰۰۱۱۰۰۱$$

۲.۴ حمله افشای راز روی پروتکل بی و همکاران

در این بخش با استفاده از نقاط ضعف تابع جایگشتی که در پروتکل پیشنهادی بی و همکاران استفاده شده حمله‌ای طراحی شده است که در آن یک مهاجم در نقش مالک جدید قصد کسب اطلاعات محرمانه برچسب را دارد. شماره مراحل با توجه به شکل ۲ عنوان شده است.

$f_x(11010111\ , Y) = 10111101$

i	1	2	3	4	5	6	7	8
X	1	1	0	1	0	1	1	1

Z	1	0	1	1	1	1	0	1
j	1	2	3	4	5	6	7	8
$A[j]$	8	5	4	1	7	6	3	2
$Y_{A[j]}$	1	1	1	1	0	0	0	0

شکل ۳. تشخیص بیت‌های کلید از روی محل بیت‌های ورودی در خروجی

۱.۲.۴ توانایی های مهاجم

ساده‌ترین راهبرد یک مهاجم، شنود پیام‌هایی است که بین برچسب و برچسب‌خوان ارسال می‌شود که ممکن است اطلاعاتی را استنباط کند و پیام‌هایی برای جعل هویت یا ردیابی یک برچسب به دست بیاورد. همچنین مهاجم ممکن است پیام‌های ارسالی را تغییر دهد و سپس پیشرفت نشست ارتباطی را زیر نظر بگیرد. در این حمله فرض می‌شود مهاجم زمان کافی برای تعامل با برچسب را دارد و چالش‌های به دقت طراحی شده‌ای را به منظور کسب اطلاعات برای برچسب ارسال کرده و نیز می‌تواند اجرای پروتکل بین برچسب‌ها و برچسب‌خوان‌ها را متوقف کند.

۳.۴ فاز ۱: تبادل اطلاعات بین مهاجم و برچسب

برای شروع پروتکل، مهاجم باید یک عدد تصادفی را با کلید مشترک بین خود و برچسب XOR کرده و پیام تولید شده را به همراه پیام Hello برای برچسب ارسال کند. ولی به دلیل نداشتن K_T، نمی‌تواند B را در مرحله ۵ محاسبه کند و پروتکل متوقف می‌شود.

با توجه به اینکه در مرحله ۴، IDS_x به صورت رمز نشده در کانال ارسال می‌شود، مهاجم با یک حمله تفاضلی و تغییر دلخواه A، K_T را بازیابی می‌کند.

برای این کار به جای تولید عدد تصادفی $r_۱$، طی چند مرحله، A را برای برچسب می‌فرستد و هر بار در یک بیت از A تفاضل ایجاد می‌کند.

مهاجم در مرحله ۲، A را به صورت رشته‌ی ۰ برای T (برچسب) ارسال می‌کند. وقتی برچسب مقدار $r'_۱$ را استخراج کند، $K_T = r'_۱$ خواهد بود و از آن برای ساخت B' استفاده می‌کند:

$$A = ۰۰۰۰۰۰۰۰$$
$$r'_۱ = A \oplus K_T = ۰۰۰۰۰۰۰۰ \oplus K_T = K_T$$
$$B' = f_x(K_T \oplus IDS_x, K_T)$$

با استفاده از مشکل شماره ۱ تابع f_x و با ایجاد تفاضل در تک تک بیت‌های رشته ورودی (A) و مقایسه آن با خروجی (IDS_x ارسال شده در کانال)، می‌تواند K_T را به دست آورد. به این صورت که در L

Algorithm 1: Disclosure attack algorithm to find secret key Y

Data: X and $f_x(X,Y)$
Result: Value of secret key Y

```
input X ;
for i=1 to 128 do
    Change ith bit of X and calculate Z = f_x(X,Y)
    j = location of changed bit in Z ;
    A[j] = i ;
j = 1 ;
while A[j] > A[j+1] And j < 128 do
    Y[A[j]] = 1;
    j = j + 1;
Fill the remaining bits of Y with 0s ;
```

Result: Y as key

شکل ۴. تشخیص بیت‌های کلید از روی محل بیت‌های ورودی در خروجی

مرحله (که L برابر است با تعداد بیت‌های رشته ورودی) هر بار فقط در یکی از بیت‌های A (و در نتیجه r'_1) تفاضل ایجاد کرده و برای برچسب ارسال می‌کند.

با ثابت بودن K_T و IDS_x وقتی iاُمین بیت در A تغییر می‌کند، jاُمین بیت در B نیز تغییر می‌کند. به این ترتیب نگاشت i به j (جایگشت) تشخیص داده می‌شود و K_T بازیابی می‌شود. در شکل ۴ نحوه بازیابی کلید تابع f_x شرح داده شده است.

برای تشخیص اینکه بیت‌های کلید ۱ یا ۰ هستند بدین طریق عمل می‌کنیم که تا جایی که ترتیب نزولی jها ادامه دارد، اندیس بیت‌های ۱ در کلید محسوب می‌شود و جایی که مقدار j افزایش یافته و مجددا نزولی می‌شود (بیت‌های باقیمانده کلید) بیت‌های ۰ کلید را تعیین می‌کند. چنانچه بیت اول کلید ۱ باشد این اندیس برابر با وزن همینگ کلید است.

برای تست اینکه K_T درست محاسبه شده یا نه، یک بار دیگر مرحله ۲ تکرار می‌شود و این بار $A = K_T$ برای برچسب ارسال می‌شود.

$$r'_1 = A \oplus K_T = K_T \oplus K_T = 00000000$$
$$B' = f_x(00000000 \oplus IDS_x, K_T) = f_x(IDS_x, K_T)$$

$f_x(IDS_x, K_T)$ را محاسبه کرده و با B مقایسه می‌کند. چنانچه با هم برابر باشند یعنی K_T درست بازیابی شده است.

بعد از مشخص شدن اندیس بیت‌ها، یک رشته تماماً ۱ ارسال می‌شود تا خروجی تابع جایگشت، IDS_x باشد. اندیس‌های j هم که مشخص شده است. پس می‌توان کلید K_T را نیز حدس زد (یا حداقل کلیدهای مشابهی که خروجی یکسان دارند را به دست آورد؛ طبق نقطه ضعف شماره ۴ تابع جایگشت).

۴.۴ فاز ۲: تبادل اطلاعات بین مهاجم و مالک قبلی

در مرحله ۵، نیاز به تولید B توسط مهاجم نیست و با فرض اینکه اطلاعات دریافت شده از برچسب درست است، عملیات با تولید C و D ادامه پیدا می‌کند.

در مرحله ۵، مهاجم باید $f_x(IDS_x, ID_n)$ را طوری ارسال کند که در جدول داده‌های شاخص موجود باشد تا مالک قبلی او را به‌عنوان خریدار مجاز شناسایی کرده و انتقال مالکیت را قانونی بداند. برای این کار می‌تواند از شنود اطلاعات منتقل شده بین برچسب‌خوان‌ها استفاده کند و این مقدار را از پیام‌های قبلی استخراج کند (حمله تکرار). همچنین مهاجم باید C و D را به گونه‌ای تولید کند که مالک قبلی تشخیص ندهد که او K_R را نداشته است. در این‌صورت مالک قبلی r'_2 را از C استخراج کرده و D را درست محاسبه کرده و پروتکل متوقف نمی‌شود. بنابراین به جای C، رشته‌ی ۰ را ارسال می‌کند که $f_x(00000000, K_x)$ با هر کلیدی همان ۰ خواهد بود. برای داشتن رشته‌ی ۰، r_2 باید برابر با IDS_x باشد به طوری که:

$$r_2 = IDS_x \rightarrow r_2 \oplus IDS_x = 0$$
$$C = f_x(r_2 \oplus IDS_x, K) = f_x(0, K) = 0$$

برای ارسال D به مالک قبلی، r_2 باید به اندازه وزن همینگ $f_x(ID_n, K_R)$ چرخش داده شود ولی تعداد چرخش مشخص نیست. بنابراین با توجه به ضعف تابع f_x که وزن همینگ را تغییر نمی‌دهد، کافی است r_2 (یا همان IDS_x) به تعداد L بار چرخش داده شده و هر بار برای مالک قبلی ارسال شود. به محض اینکه مالک قبلی پاسخ پیام‌ها را در کانال ارسال کرد مشخص می‌شود که $D = D$ شده و چرخش کافی بوده است. این تعداد چرخش به عنوان وزن همینگ ID_n در نظر گرفته می‌شود.

$$D = Rot(r_2, f_x(ID_n, K_R))$$
$$D = Rot(IDS_x, x),$$
$$x = 0, 1, 2, \ldots, wt(ID_n), \quad 0 < x < L(ID_n)$$

بنابراین مهاجم بدون داشتن K_R و ID_n پیام‌ها را از مالک قبلی دریافت می‌کند و در پایان این مرحله، مالک قبلی مقادیر E، F، G، H را در کانال ارسال می‌کند.

۵.۴ فاز ۳: تبادل مجدد اطلاعات بین مهاجم و برچسب

با فرض اینکه مالک قبلی به درستی احراز هویت شده، مهاجم نیازی به محاسبه r'_3 و F' ندارد. برای بازیابی $K_2 \oplus K_3$، چندین بار H را چرخش می‌دهد و هر بار با $(K_2 \oplus K_3)'$ به دست آمده I' را حساب کرده و $G \oplus I'$ را برای برچسب ارسال می‌کند. چنانچه مقادیر دریافتی با مقادیر محاسبه شده توسط برچسب مطابقت نداشته باشد، برچسب پروتکل را متوقف می‌کند و پاسخی نمی‌دهد. مهاجم آنقدر چرخش H را ادامه می‌دهد تا برچسب پاسخ دهد. پاسخ برچسب به این معنی است که

مهاجم موفق به یافتن $K_۲ \oplus K_۳$ شده است و با استفاده از آن در فرمول G، $K_۱ \oplus K_۲$ را بازیابی می‌کند.

در مرحله ۱۳ و ۱۴، مهاجم با K_T و IDS_x، M و N را تولید کرده و برای برچسب ارسال می‌کند.

۶.۴ به‌روزرسانی مقادیر محرمانه در مرحله آخر

در آخرین مرحله، برچسب پس از بررسی صحت M و N کلیدها را به‌روزرسانی می‌کند. مهاجم با داشتن همه موارد بازیابی شده، می‌تواند کلیدهای جدید $k_۱^{new}$ و $K_۲^{new}$ و $K_۳^{new}$ و شناسه جدید برچسب (IDS_x^{new}) را بازیابی کند. به این ترتیب از همه اطلاعات محرمانه برچسب اطلاع دارد.

نداشتن ID_n مشکلی برای مهاجم ایجاد نمی‌کند، زیرا در هیچ قسمتی از پروتکل از خود ID_n استفاده نمی‌شود بلکه از کد همینگ آن استفاده می‌شود که مهاجم در مرحله ۶ کد همینگ ID_n را به دست آورده است.

به این ترتیب مشخص شد که ادعای بی و همکاران در طراحی پروتکل امن سبک وزن انتقال مالکیت صحیح نیست و با $۴ \times L$ مرتبه اجرای پروتکل می‌توان به کلیدهای برچسب دسترسی پیدا کرد.

۵ نتیجه‌گیری

مزیت‌های سیستم‌های RFID سبب استفاده روزافزون آنها شده است. از طرفی محدودیت منابع محاسباتی و نگهداری اطلاعات در برچسب‌ها مانع بکارگیری پروتکل‌های رمزنگاری امن برای سامانه‌های RFID شده است. در این مقاله در ابتدا یک تحلیل کامل در مورد آسیب‌پذیری تابع جایگشتی f_x ارائه نمودیم. سپس نشان دادیم که استفاده از تابع جایگشتی سبک‌وزن در طراحی پروتکل انتقال مالکیت چگونه باعث آسیب‌پذیری این پروتکل در برابر حمله افشای راز می‌شود. همچنین با محاسبه تعداد مرتبه اجرای این حمله نشان دادیم که این حمله قابلیت پیاده‌سازی به صورت عملی را خواهد داشت.

مراجع

[1] Bi, Ye, Fan, Kai, Zhang, Kuan, Bai, Yuhan, Li, Hui, and Yang, Yintang. A secure and efficient two-party protocol enabling ownership transfer of rfid objects. *IEEE Internet of Things Journal*, 10(18):16225–16237, 2023.

[2] Osaka, Kyosuke, Takagi, Tsuyoshi, Yamazaki, Kenichi, and Takahashi, Osamu. An efficient and secure rfid security method with ownership transfer. in *2006 International Conference on Computational Intelligence and Security*, vol. 2, pp. 1090–1095. IEEE, 2006.

[3] Jäppinen, Pekka and Hämäläinen, Harri. Enhanced rfid security method with ownership transfer. in *2008 International Conference on Computational Intelligence and Security*, vol. 2, pp. 382–385. IEEE, 2008.

[4] Juels, Ari. " yoking-proofs" for rfid tags. in *IEEE Annual Conference on Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second*, pp. 138–143. IEEE, 2004.

[5] Zuo, Yanjun. Changing hands together: a secure group ownership transfer protocol for rfid tags. in *2010 43rd Hawaii International Conference on System Sciences*, pp. 1–10. IEEE, 2010.

[6] Jannati, Hoda and Falahati, Abolfazl. Cryptanalysis and enhancement of a secure group ownership transfer protocol for rfid tags. in *International Conference on e-Democracy*, pp. 186–193. Springer, 2011.

[7] Kulseng, Lars, Yu, Zhen, Wei, Yawen, and Guan, Yong. Lightweight mutual authentication and ownership transfer for rfid systems. in *2010 proceedings IEEE infocom*, pp. 1–5. IEEE, 2010.

[8] Safkhani, Masoumeh, Naderi, Majid, Bagheri, Nasour, and Sanadhya, Somitra Kumar. Cryptanalysis of some protocols for rfid systems. *Cryptology ePrint Archive*, 2011.

[9] Kardaş, Suleyman, Akgün, Mete, Kiraz, Mehmet Sabir, and Demirci, Huseyin. Cryptanalysis of lightweight mutual authentication and ownership transfer for rfid systems. in *2011 Workshop on Lightweight Security & Privacy: Devices, Protocols, and Applications*, pp. 20–25. IEEE, 2011.

[10] Lee, Cheng-Chi, Li, Chun-Ta, Cheng, Chung-Lun, and Lai, Yan-Ming. A novel group ownership transfer protocol for rfid systems. *Ad Hoc Networks*, 91:101873, 2019.

[11] Moazami, Farokhlagha and Safkhani, Masoumeh. Seotp: a new secure and efficient ownership transfer protocol based on quadric residue and homomorphic encryption. *Wireless Networks*, 26:5285–5306, 2020.

[12] Qingkaun, Dong, Wenxin, Gao, Li, Li, Xiaolong, Ren, and Xiaoqian, Zhao. Lightweight rfid ownership transfer protocol based on blockchain. in *2021 IEEE Globecom Workshops (GC Wkshps)*, pp. 1–7. IEEE, 2021.

[13] Luo, Jia-Ning and Yang, Ming-Hour. A secure partial rfid ownership transfer protocol with multi-owners. *Sensors*, 20(1):22, 2019.

[14] Gunnarsson, Martin and Gehrmann, Christian. Secure ownership transfer for the internet of things. in *ICISSP*, pp. 33–44, 2020.

[15] Yang, Ming-Hour, Chen, Chien-Hung, Luo, Jia-Ning, Vijayalakshmi, M, and Shalinie, S Mercy. An efficient and secure method for simultaneous ownership transfer of multiple mobile readers. *IEEE Systems Journal*, 17(3):3611–3622, 2023.

MonadiAfta-Bianual Scientific Journal for Cyberspace Security
2024, Volume 22, Number 2 (pp. 1–11)
https://monadi.isc.org.ir

Presented at the ISCISC 2024 in Tarbiat Modares University & Research Institute for Information and Communication Technology, Tehran, Iran

Security analysis of a lightweight ownership transfer protocol in RFID systems★

Ghodsieh Karimi *, Morteza Adeli and Mohammad Ali Hadavi

Electeronical and Computer Engineering Department, Malek Ashtar University of Technology, Tehran, Iran

ARTICLE INFO.

Keywords:
RFID
Ownership transfer protocol
Security analysis
Secret disclosure attack

Type: Research paper

ABSTRACT

With the increasing use of RFID tags, there is a need for specific protocols to communicate with these tags. Among these protocols, the ownership transfer stands out as it ensures the security and privacy of objects for the new owner after a change of ownership. Recently, a lightweight object ownership transfer protocol has been proposed for RFID networks. This protocol utilizes a lightweight linear function for security. The designers of the protocol claim that it is secure against known attacks while also being lightweight. In this paper, we identify vulnerabilities in the function used in this protocol and demonstrate that it is susceptible to the secret disclosure attack. We show that with at most $4 \times L$ executions of the protocol (where L is the key length), one can obtain the necessary information from intercepted data to execute the attack and subsequently recover the shared keys used in the protocol.

© 2024 ISC

★ The ISCISC 2024 Program Committee effort is highly acknowledged for reviewing this paper.
* Corresponding author
Email addresses: gh_karimi@mut.ac.ir (Ghodsieh Karimi), adeli@mut.ac.ir (Morteza Adeli), hadavi@mut.ac.ir (Mohammad Ali Hadavi)
© 2024 ISC. All rights reserved.