توسعه یک پروتکل توزیع کلید دو سویه مبتنی بر تابع لگاریتم گسسته

مهدی برنجکوب
دانشجوی دوره دکتری
دانشکده برق و کامپیوتر، دانشگاه صنعتی اصفهان

محمدرضا عارف
استاد
دانشکده برق، دانشگاه صنعتی شریف

حسن سعیدی
استادیار
دانشکده برق و کامپیوتر، دانشگاه صنعتی اصفهان

چکیده

پروتکل های توزیع کلید دو سویه مبتنی بر ایده رمزنگاری نامتقارن زیرمجموعه مهمی از پروتکل های توزیع کلید را تشکیل می دهند. در این مقاله سه رویکرد متفاوت برای طراحی این خانواده از پروتکل ها مورد بحث و بررسی قرار می گیرند. دو رویکرد اول به طور متداول متکی به استفاده از یك سیستم رمز کلید عمومی (تابع یك طرفه دریچه دار) هستند، حال آنکه رویکرد سوم تنها به یك تابع یك طرفه بدون دریچه ولی جابجاپذیر متکی است. در ادامه یك گونه تازه و عملی از پروتکل مبتنی بر رویکرد سوم که از ایده صدور گواهی با زمان انقضای معین بهره می برد توسعه داده می شود. بدین ترتیب نیاز به دسترسی به فایل معتبر کلید عمومی کاربران شبکه مرتفع می گردد. در همین راستا پروتکل کارآمدی برای دریافت گواهی کلید عمومی از مرکز صدور گواهی ارائه می گردد. بدین ترتیب کاربران شبکه نیازی به دانش فنی قابل توجه و امکانات محاسباتی ویژه ندارند و ابزارهای تولید عدد تصادفی و محاسبه تابع نمای گسسته آنها را کفایت خواهد کرد.

A Two-Party Key Distribution Protocol Based on Discrete Logarithm

M. Berenjkoub
Ph.D. Student
Department of Electrical and Computer Engineering, Isfahan University of Technology

M.R. Aref
Professor
Department of Electrical Engineering, Sharif University of Technology

H. Saidi
Assistant Professor
Department of Electrical and Computer Engineering, Isfahan University of Technology

Abstract

An important subset of the key distribution protocols in cryptography is two-party key distribution protocols based on the idea of asymmetric cryptography. In this paper, we examine three different approaches to the design of these protocols. The first two approaches are classic ones based on a public key cryptosystem (trapdoor one-way function). However, the third approach is new and based only on using a commutative one-way function without any trapdoor. To achieve a practical system, we modify the protocol in the third approach based on discrete logarithm function that uses certificates with a definite expiration time. As a result, it is not necessary to access to an authenticated directory of users' public keys anymore. In completion of this system, we offer a low-cost protocol to get certificates from certification authority (by principals). Consequently, the principals do not need lots of expertise or knowledge and high processing power, just random generators and dicrete exponentors are enough.

۱ ـ مقدمه

پروتکل های توزیع کلید مبتنی بر ایده رمزنگاری نامتقارن[1] زیرمجموعه مهمی از پروتکل های توزیع کلید را تشکیل می دهند. استفاده از ایده تابع یک طرفه[2] در سیستم های رمز نامتقارن موجب می شود که نیاز به قرارداد و تبادل کلیدهای اصلی مشترک بین طرف های ارتباط مرتفع گردیده، پروتکل های غیرمتمرکز توزیع کلید مجال طرح و بکارگیری بیابند. بدین ترتیب بدون ایجاد محدودیت های جدی در مورد توزیع گسترده کلیدهای اصلی مشترکین، می توان از قید اجتناب ناپذیر راه اندازی مرکز توزیع کلید[3] رها شد. البته باید اعتراف کرد که حتی با بکارگیری ایده رمزنگاری نامتقارن نیز امکان طراحی پروتکل معتبر توزیع کلید بین دو سوی ارتباط، بدون وجود هیچ اقدام مقدماتی یا ارتباط اضافی، وجود ندارد. هر چند دیگر به مرکزیت امن یا کانال محرمانه ای نیاز نیست، لیکن لازم است هر مشترک کلید عمومی واقعی طرف ارتباط خود را بتواند در اختیار بگیرد. در واقع همچنان یک مرکز ذخیره سازی معتبر برای کلیدهای عمومی مشترکین شبکه ارتباطی مورد نیاز می باشد.

در این مقاله یک پروتکل کارآمد توزیع کلید مبتنی بر ایده رمزنگاری نامتقارن معرفی شده و با پروتکل های متناظر موجود، مورد مقایسه قرار می گیرد. پروتکل مورد نظر در واقع یک پروتکل توزیع کلید مبتنی بر تابع یکطرفه جابجاپذیر است که از تابع لگاریتم گسسته به عنوان تابع یک طرفه جابجاپذیر بهره می گیرد. این پروتکل برای اولین بار به طور اجمال و به عنوان یک مثال از پروتکل های مبتنی بر توابع یک طرفه جابجاپذیر در [۱] ارائه گردید ودر این مقاله فرصتی است که از زوایای مختلف و با تأکید بر جنبه های کاربردی مورد بررسی و مطالعه قرار گرفته با پروتکل های متناظر مقایسه گردد.

در ادامه این مقاله، پس از مقدمه، ابتدا با رویکردهای مختلف طراحی پروتکل با استفاده از ایده رمزنگاری نامتقارن آشنا می شویم و خواهیم دید که پروتکل مورد نظر در این مقاله متکی بر رویکرد تازه ای در طراحی این قبیل از پروتکل ها است. سپس رویکردهای طراحی بیان شده مورد ارزیابی قرار می گیرند. در این ارزیابی ارجحیت رویکرد پیشنهادی مقاله بر سایر رویکردها مشخص می شود. تا اینجای مقاله، فرض بر این است که دو سوی ارتباط از کلید عمومی یکدیگر آگاه می باشند. از آنجا که این فرض ممکن است در بسیاری از موارد عملی برقرار نباشد ،در بخش بعدی مقاله با استفاده از ایده صدور گواهی[4] نسبت به تکمیل فرآیند طراحی پروتکل پیشنهادی اقدام می گردد و نشان داده می شود که از این جنبه پروتکل توزیع کلید مبتنی بر تابع لگاریتم گسسته نشانی از ضعف ندارد. در پایان نتایج حاصله از تحقیق حاضر جمع بندی می گردد.

۲ ـ رویکردهای مختلف طراحی پروتکل با بکارگیری ایده رمزنگاری نامتقارن

راه شناخته شده برای استفاده از ایده رمزنگاری نامتقارن، استفاده مستقیم از یک سیستم رمز کلید عمومی[5] در پروتکل توزیع کلید است. سیستم رمز کلید عمومی در حقیقت یک تابع یک طرفه دریچه دار[6] است که در آن محاسبه متن رمز شده از روی متن اصلی به سهولت و با بکارگیری تابع رمزگذاری E قابل انجام است. در حالیکه دست یابی به متن اصلی از روی متن رمز شده به دلیل یک طرفه بودن تابع E بسیار مشکل و در عمل غیرممکن می باشد، مگر برای کسی که کلید خصوصی[7] (یا دریچه تابع یک طرفه) را در اختیار دارد که وی قادر به محاسبه تابع رمزگشایی D خواهد بود ([۲]). هر سیستم رمز کلید عمومی به دو گونه قابل بکارگیری است:

- ابتدا تابع رمز گذاری Eبه متن اصلی اعمال می شود و متن رمز شده حاصل می گردد. سپس برای دست یابی مجدد به متن اصلی از تابع رمزگشایی D استفاده می شود.
- ابتدا تابع D (متناظر با کلید خصوصی) به متن اصلی اعمال می شود و بدین ترتیب متن اصلی امضاء[8] (S) می گردد. سپس برای وارسی امضاء از تابع E استفاده می شود (البته در برخی از توابع یک طرفه دریچه دار ممکن است نتوان به شکل مذکور عمل امضاء را انجام داد و روش های غیر مستقیمی برای این منظور پیشنهاد گردد).

بر حسب اینکه کدامیک از دو گونه فوق در طراحی پروتکل به کار گرفته شود، دو رویکرد حاصل می شود که اولی را رویکرد استفاده از تابع رمز E و دومی را رویکرد استفاده از تابع امضای S می نامیم.

گذشته از بکارگیری مستقیم یک تابع یک طرفه دریچه دار، رویکرد سومی را نویسندگان در [۱] مطرح ساخته اند که در آن از یک تابع یک طرفه بدون دریچه

جابجاپذیر در طراحی پروتکل توزیع کلید استفاده می‌شود. همان طور که خواهیم دید، رویکرد اخیر علاوه بر آنکه از جنبه نظری دارای تکیه گاه امنیتی اطمینان بخش تری است، امکان صرفه جویی بیشتری را نیز در لوازم طراحی و پیاده سازی پروتکل فراهم می‌آورد.

در ادامه به معرفی رویکردهای سه گانه فوق پرداخته می‌شود.

۲-۱- رویکرد استفاده از تابع رمز E

رویکرد استفاده از تابع رمز E، در [۳] به عنوان یک پروتکل قابل قبول مطابق نمودار شکل ۱ گزارش گردیده است. در این شکل، نماد $E_X[.]$ بیانگر رمزگذاری با کلید عمومی متعلق به کاربر X و نماد $K_S\{...\}$ بیانگر رمزگذاری با سیستم رمز متقارن مورد توافق با کلید جلسه[۹] است. همچنین N_X یک عدد تصادفی یک بار مصرف است که از سوی کاربر X تولید گردیده و اصطلاحاً نانس[۱۰] نامیده می‌شود.

قبل از توضیح پروتکل شکل ۱، مفروضاتی که پروتکل بر مبنای آنها کار می‌کند، معرفی می‌شوند. کاربران شبکه بر روی یک سیستم رمز کلید عمومی (برای بکارگیری در مراحل ۱ و ۲ پروتکل) ویک سیستم رمز کلید پنهان (برای بکارگیری در مرحله ۳ پروتکل و رمزنگاری داده‌ها با کلید جلسه) توافق کرده‌اند و هر کاربر علاوه بر کلید خصوصی خود قابل معتبری از کلید عمومی سایر کاربران را در دسترس دارد. فرض بر این است که کاربر A (به عنوان آغازگر) می‌خواهد ارتباطی امن با کاربر B (به عنوان مخاطب) برقرار نماید. برای این منظور آغازگر نانس N_A را تولید کرده همراه با شناسه خود، پس از رمزگذاری تحت کلید عمومی مخاطب، برای وی ارسال می‌کند. کاربر B پس از دریافت پیام ۱ ابتدا آن را با کلید خصوصی خود رمزگشایی می‌کند و از درخواست ارتباط کاربری که خود را A معرفی کرده مطلع می‌شود. مخاطب، در صورت تمایل به برقراری ارتباط، کلید جلسه K_S و نانس N_B را تولید نموده همراه با N_A دریافتی، آنها را پس از رمزگذاری تحت کلید عمومی آغازگر، برای وی باز می‌فرستد. کاربر A پس از دریافت و رمزگشایی پیام ۲ با کلید خصوصی خود، ابتدا از طریق بررسی وجود N_A در پیام دریافتی، هویت کاربر B را احراز می‌کند (چون پیام ۱ تحت کلید عمومی مخاطب رمز شده وی تنها فردی است که قادر به دست یابی به N_A و باز فرستادن آن برای آغازگر است). سپس آغازگر کلید جلسه K_S را قبول نموده و به منظور احراز اصالت[۱۱] خود برای کاربر B، نانس N_B را تحت کلید جلسه K_S و با استفاده از سیستم رمز کلید امن رمزگذاری کرده برای وی ارسال می‌نماید. مخاطب با رمزگشایی پیام ۳ و دست یابی به N_B از اصالت کاربر A اطمینان حاصل می‌نماید و از این پس شرایط برای تبادل داده‌های رمز شده تحت کلید جلسه آماده می‌باشد. دقت شود که در پیام ۳، آغازگر می‌توانست نانس N_B را تحت کلید عمومی مخاطب رمزگذاری نماید، لکن به دلیل آن که بار محاسباتی استفاده از سیستم‌های رمز کلید عمومی به مراتب بیشتر از سیستم‌های رمز کلید پنهان است، از این امکان اجتناب گردیده است. همچنین توجه شود که استفاده از نانس‌ها در پروتکل، اجازه حمله تکرار[۱۲] را از دشمن[۱۳] می‌گیرد و طرفین ارتباط را از تازه بودن پیام‌ها مطمئن می‌سازد.

در بخش بعدی به ارزیابی این پرونکل پرداخته می‌شود و خواهیم دید که علیرغم استحکام ظاهری آن، رخنه مهمی در آن وجود دارد.

۲-۲- رویکرد استفاده از تابع امضای S

رویکرد دیگر در استفاده از ایده رمزنگاری نامتقارن در طراحی پروتکل‌های توزیع کلید در [۴] عرضه گردیده است. در اینجا نیز از یک سیستم رمز کلید عمومی بهره برداری می‌شود، لیکن به جای استفاده از تابع رمز E، تابع امضای S سیستم رمز به کار گرفته می‌شود. نمودار شکل ۲ این پروتکل را نمایش می‌دهد. در این شکل، نماد e (.) بیانگر تابع نمای گسسته[۱۴] است که تعریف دقیق آن عبارت است از:

$$e(x) := m^x \bmod p \qquad (۱)$$

که در آن p یک عدد اول قوی[۱۵] و m یک ریشه بنیادین[۱۶] در پیمانه p است. همچنین در شکل ۲ از نماد $S_X[.]$ برای بیان امضاء توسط کاربر X استفاده شده و نماد R_X بیانگر عددی تصادفی در بازه (p، ۱] است. این پروتکل را می‌توان گونه معتبری از سیستم توزیع کلید دیفی - هلمن[۱۷] ([۲]) قلمداد کرد. از این رو برای فهم بهتر پروتکل شکل ۲، ابتدا مروری کوتاه بر این سیستم توزیع کلید می‌شود. در حالی که محاسبه تابع نمای گسسته e (x) با بکارگیری الگوریتم مجذور و ضرب به سهولت انجام پذیر است، حصول x از روی e (x)، که اصطلاحاً لگاریتم گسسته نامیده می‌شود، بسیار مشکل

و در صورتی که پیمانه p به اندازه کافی بزرگ انتخاب شود عملاً ناممکن است. در سیستم توزیع کلید دیفی - هلمن از همین حقیقت برای قرارداد کلید جلسه بین طرفین ارتباط استفاده می‌شود. کاربر A و کاربر B، به عنوان دوسوی ارتباط، به ترتیب اعداد تصادفی R_A و R_B را تولید و ذخیره‌سازی نموده، تابع نمای گسسته آن را (یعنی $e(R_A)$ و $e(R_B)$) محاسبه و حاصل آن را بین یکدیگر از کانال ناامن رد وبدل می‌کنند. آنگاه کلید جلسه K_S به ترتیب زیر توسط هر یک از دو سوی ارتباط قابل محاسبه است:

$$K_S = e(R_A R_B) = m^{R_A R_B} \bmod p$$

$$= [e(R_A)]^{R_B} \bmod p$$

$$= [e(R_B)]^{R_A} \bmod p \qquad (۲)$$

دقت شود که شخص ثالث (دشمن) برای استخراج کلید جلسه از روی مقادیر $e(R_A)$ و $e(R_B)$ نیازمند محاسبه تابع لگاریتم گسسته است. همان طور که می‌دانیم سیستم توزیع کلید دیفی - هلمن غیر معتبر است و به سهولت با حمله «فردی در میان»[۱۸] شکسته می‌شود. پروتکل شکل ۲، راه چاره‌ای برای معتبرسازی این سیستم توزیع کلید است که از سوی دیفی و همکارانش پیشنهاد گردیده است.

همان طور که در شکل ۲ مشاهده می‌شود، در پیام‌های ۱ و ۲ مقادیر $e(R_A)$ و $e(R_B)$ بین دو سوی ارتباط مبادله می‌شوند و کلید جلسه K_S براساس رابطه (۲) از روی آنها ساخته می‌شود. همزمان مخاطب اقدام به امضای توابع مذکور پس از عبور آنها از یک تابع درهم یک طرفه[۱۹] کرده حاصل را به صورت رمزگذاری شده تحت کلید جلسه K_S به پیام ۲ منضم می‌نماید. آغازگر با دریافت، رمزگشایی و وارسی امضای مخاطب، اصالت مخاطب را احراز می‌نماید. آنگاه متقابلاً با تدارک پیام ۳ شرایط احراز اصالت خود را برای مخاطب فراهم می‌کند.

در خاتمه توجه خواننده به دو نکته در مورد این پروتکل جلب می‌شود. اول اینکه توابع نمای $e(R_A)$ و $e(R_B)$ از زاویه‌ای دیگر ایفاگر نقش نانس هستند و در نتیجه از امکان وقوع حمله تکرار جلوگیری می‌کنند. دوم آنکه رمزگذاری امضاءها در پیام‌های ۲ و ۳، امکان این را که اغیار بتوانند علیرغم عدم اطلاع از کلید جلسه خود را به عنوان یکی از دو سوی ارتباط جا بزنند، منتفی می‌سازد.

۲-۳- رویکرد پیشنهادی

رویکرد سومی نیز برای طراحی پروتکل توزیع کلید مبتنی بر ایده رمزنگاری نامتقارن قابل تصور است که بر اساس آن از تابع یک طرفه بدون دریچه جابجاپذیر استفاده می‌شود [۱]. تابع f با آرگومان‌های ورودی m و x، قابل نمایش مطابق شکل ۳، یک طرفه نامیده می‌شود اگر و تنها اگر:

- دامنه ورودی‌ها و حوزه مقادیر f یکسان بوده و در فضای متناهی پیام {M} واقع باشد،
- تابع f یک به یک یا حداقل شبه یک به یک باشد،
- محاسبه f با ورودی‌های معین m و x ساده و سر راست باشد،
- محاسبه x به ازای ورودی m و خروجی f معین عملاً امکان‌پذیر نباشد.

بنا به تعریف تابع یک طرفه f جابجاپذیر گفته می‌شود اگر و تنها اگر رابطه زیر برای عملگر ترکیب توابع (o) برقرار باشد.

$$f_{x_1} o f_{x_2}(m) = f_{x_2} o f_{x_1}(m) \qquad (۳)$$

به عنوان مثال، تابع نمای گسسته تعریف شده در رابطه (۱) یک تابع یک طرفه جابجاپذیر است. طراحی یک پروتکل معتبر توزیع کلید با استفاده از تابع یک طرفه جابجاپذیرf فوق برای اولین بار در [۱] ارائه گردیده است. در اینجا برای امکان مقایسه و تبیین بهتر ترجیح داده می‌شود که پروتکل مذکور در قالب مثال تابع یک طرفه جابجاپذیر نمای گسسته، تعریف شده در رابطه (۱)، معرفی شود (نمودار شکل ۴).

در این شکل از نماد D_X برای بیان کلید خصوصی متعلق به کاربر X استفاده شده که عبارت از یک عدد دلخواه در بازه (۱ , p) است. کلیدعمومی متناظر با کلید خصوصی D_X، تابع نمای گسسته آن، یعنی $e(D_X)$، است که توسط کاربر X در فایل عمومی قرار داده می‌شود.

پروتکل شکل ۴ را نیز می‌توان گونه معتبری ازسیستم توزیع کلید دیفی - هلمن قلمداد نمود. از این گذشته، در این پروتکل نیز مشابه پروتکل شکل ۲، برای احراز اصالت دو سوی ارتباط در پیام‌های ۲ و ۳ از کلید خصوصی کاربران بهره‌برداری شده است. با این حال، علیرغم تشابهات مذکور، وجود تفاوت ماهوی بین دو پروتکل به خوبی گویای اتخاذ رویکردهای متفاوت در طراحی آنها است. قبل از بیان این تفاوت مروری بر نحوه

عملکرد پروتکل پیشنهادی ضرورت دارد.

آغازگر برای درخواست مذاکره با مخاطب، عدد تصادفی $R_A \in F_P$ را تولید نموده اقدام به محاسبه $e(R_A)$ کرده آن را همراه با شناسه خود برای کاربر B ارسال می‌نماید. کاربر B, با دریافت پیام ۱ و آگاهی از قصد برقراری ارتباط توسط کسی که خود را A معرفی می‌کند در صورت تمایل به برقراری ارتباط، برای احراز هویت خود با بکارگیری کلید خصوصی D_B و نانس دریافتی $e(R_A)$، تابع یک طرفه $e(D_B R_A)$ را محاسبه می‌نماید (با توجه به محرمانه بودن D_B، کسی جز B قادر به تولید این تابع نیست). همچنین کاربر B، همچون آغازگر، اقدام به تولید نانس $e(R_B)$ نموده آن را به عنوان بخشی از پیام ۲ برای آغازگر باز می‌فرستد. در همین مرحله کلید جلسه مورد نیاز با ایفای نقش مساوی دو سوی ارتباط و با ترکیب نانس‌ها به شکل $e(R_A R_B)$ حاصل گردیده است (دقت شود که هر یک از دو سوی ارتباط با روشی مختص به خود قادر به دست یابی به کلید جلسه هستند ودر حقیقت جابجاپذیری تابع نمای گسسته عامل به نتیجه واحد رسیدن دو سوی ارتباط است. این موضوع همچنین در مورد نحوه تولید $e(D_B R_A)$ توسط مخاطب و نحوه وارسی آن از سوی آغازگر نیز مصداق دارد). برای آن که در طی پیام ۲ دشمن قادر به اجرای یک حمله فعال از طریق جایگزینی نانس $e(R_B)$ با تابع دلخواه دیگری نشده خود را به جای مخاطب واقعی جا نزند لازم است عبارت $e(D_B R_A)$ پس از رمزگذاری تحت الگوریتم رمز کلید پنهان مورد توافق و با استفاده از کلید جلسه $K_S = e(R_A R_B)$ همراه با نانس $e(R_B)$ و شناسه مخاطب برای کاربر A ارسال گردد. بالاخره، در پیام ۳، آغازگر با بکارگیری کلید خصوصی خود (یعنی D_A) و نانس دریافتی، تابع یک طرفه $e(D_A R_B)$ را محاسبه نموده همراه با شناسه‌اش برای کاربر B باز پس می‌فرستد و بدین ترتیب امکان احراز اصالت خود را برای مخاطب فراهم می‌آورد. دشمن برای شکستن پروتکل در مراحل مختلف چاره‌ای جز محاسبه تابع لگاریتم گسسته ندارد که عملاً توان انجام آن را نخواهد داشت.

در پروتکل شکل۲، با توجه به وجود تابع یک طرفه دریچه دار، هنگام انجام امضاء، تابع یک طرفه مذکور از طرف مشکل آن به کمک کلید خصوصی محاسبه گردید. حال آن که در پروتکل شکل۴ ، هنگام بکارگیری هر کدام از کلیدهای عمومی و خصوصی، همواره تابع یک طرفه تنها از طرف ساده‌اش محاسبه می‌گردد. به عبارت دیگر به دلیل عدم نیاز به محاسبه تابع یک طرفه از طرف مشکل آن، در پروتکل پیشنهادی از هیچ تابع یک طرفه دریچه داری استفاده نشده است. در حقیقت ایده اساسی رویکرد پیشنهادی در درک عدم نیاز پروتکل توزیع کلید به محاسبه تابع یک طرفه از طرف مشکل آن نهفته است و در نتیجه تابع یک طرفه بدون دریچه جابجاپذیر جایگزین تابع یک طرفه دریچه دار گردیده است. چون تابع یک طرفه جابجاپذیر مورد نیاز در شکل ۴ تابع لگاریتم گسسته انتخاب شده این پروتکل عملاً به صورت گونه ای معتبر از سیستم توزیع کلید دیفی ـ هلمن در آمده است.

۳ ـ ارزیابی رویکردهای طراحی پروتکل

پس از ارائه سه رویکرد طراحی پروتکل براساس استفاده از ایده رمزنگاری نامتقارن، در این بخش به تحلیل، ارزیابی و مقایسه پروتکل های معرفی شده پرداخته می‌شود. در این راستا، ابتدا ضعف امنیتی رویکرد استفاده از تابع رمز E با ارائه یک سناریوی مخاطره آفرین نشان داده می‌شود و سپس تلاش می‌گردد، مقایسه جامعی بین دو رویکرد باقیمانده انجام شود.

۳ ـ ۱ ـ معرفی رخنه در پروتکل مبتنی بر رویکرد استفاده از تابع رمز E

فرض کنید آغازگر یک مؤسسه تجاری است که قصد دارد طی ایجاد یک ارتباط امن با مخاطب خود که بانکی است که در آن حساب دارد دستور پرداختی از حساب خود صادر کند. همچنین فرض کنید که مؤسسه A در بانک دیگری (به نام C) نیز حساب دارد و بانک مخاطب (B) از این موضوع آگاه است. چنانچه به دنبال آغاز یک ارتباط از سوی مؤسسه A با بانک B، بانک B بتواند از روی قرائن حدس بزند که هدف A از این ارتباط صدور یک دستور پرداخت از حساب خودش است، براساس سناریویی که در شکل ۵ ارائه شده بانک B قادر خواهد بود یک ارتباط همزمان با بانک C برقرار نموده خود را مؤسسه A معرفی کند و به جای پرداخت مورد نظر A، بانک C را وادارد که اقدام به پرداخت مذکور نماید، بدون آنکه مؤسسه A و بانک C از واقعیت امر آگاه شوند.

مطابق شکل ۵، بانک B پس از تعویض پیام ۱ با پیام $1'$، در ادامه سناریو نقش یک واسطه غیر فعال را بازی می‌کند. بدیهی است که B قادر به دست یابی به K_S نبوده و در نتیجه از محتوای پیامهایی که واسطه انتقال

آنها بین A و C است، نمی‌تواند آگاه شود. با این وجود با گمراه کردن A و C هویت واقعی دو سوی ارتباط را مخدوش کرده است.

بررسی دقیق سناریوی فوق گویای این واقعیت است که پیام ۲ در پروتکل مورد بحث، فاقد ویژگی خاصی از صادرکننده آن (یعنی مخاطب) است. به عبارت دیگر پیام ۲ پروتکل شکل ۱ بایستی به نوعی از هویت مخاطب بهره‌مند گردد.

۳ـ۲ـ بررسی مقایسه‌ای پروتکل‌های برآمده از دو رویکرد دیگر

پروتکل‌های معرفی شده در شکل‌های ۲ و ۴ هر دو گونه‌ای معتبر از سیستم توزیع کلید دیفی ـ هلمن محسوب می‌شوند. بنابراین هر دو پروتکل به طور بنیادی به تابع یک طرفه لگاریتم گسسته اتکاء دارند. در حالی که پروتکل پیشنهادی در شکل ۴ بی نیاز از تکیه به تابع دیگری است، پروتکل مبتنی بر رویکرد استفاده از تابع امضای S، همچنین محتاج اتکاء به یک تابع یک طرفه دریچه‌دار است تا براساس آن محاسبه تابع امضاء میسر شود. این تفاوت اساسی از دو جنبه امنیتی و پیچیدگی حائز اهمیت است. از دیدگاه امنیتی هر چه تعدد مسائلی که پروتکل به آنها متکی است، بیشتر شود بر رخنه‌های بالقوه قابل دست‌یابی در پروتکل افزوده می‌شود. به لحاظ نظری، تکیه یک پروتکل به تابع یک طرفه بدون دریچه به مراتب اطمینان بخش‌تر از تکیه آن به تابع یک طرفه دریچه‌دار است (در عمل نیز در دو دهه گذشته شاهد شکستن دریچه برخی از توابع یک طرفه دریچه‌دار بوده‌ایم). از دیدگاه پیچیدگی، به طور طبیعی پروتکل شکل ۲ به دلیل نیاز به طراحی و پیاده‌سازی یک تابع یک طرفه دریچه‌دار اضافی مستلزم پیچیدگی و صرف هزینه بیشتری است. واقعیت این است که در پروتکل شکل ۲، برای آنکه عملاً از امنیت تابع امضاء اطمینان نسبی حاصل شود و نقطه بحرانی پروتکل از لحاظ امنیتی، حل مسأله لگاریتم گسسته باشد، پیچیدگی علیحده‌ای را باید پذیرفت.

برای آنکه بتوان به طور دقیق‌تری پیچیدگی دو پروتکل را مقایسه نمود لازم است سیستم امضای نمونه‌ای برای پروتکل شکل ۲ در نظر گرفته شود. برای این منظور فرض می‌کنیم که این پروتکل از سیستم مشهور RSA بهره‌برداری کند. برای تحقق الگوریتم RSA بایستی اقدامات اضافی زیر به طور جداگانه توسط هر یک از کاربران شبکه انجام شود:

ـ تهیه دو عدد اول قوی پنهان و به اندازه کافی بزرگ: انجام این کار باتوجه به اهمیت محرمانه ماندن این اعداد، برای کاربران دارای دانش فنی کم و توان محاسباتی محدود می‌تواند مسأله آفرین باشد.

ـ محاسبه کلیدهای عمومی و خصوصی از روی اعداد اول فوق.

در پروتکل پیشنهادی شکل ۴، کلید خصوصی هر کاربر با انتخاب تصادفی عددی دربازهٔ (p , ۱) به دست می‌آید و کلید عمومی متناظر با آن از محاسبه یک نمای گسسته حاصل می‌شود. اهمیت روند پیچیده‌تر دست‌یابی به کلیدها برای کاربران وقتی بارزتر می‌شود که بدانیم حصول اطمینان از امنیت سیستم، ایجاب می‌کند که کاربران کلیدهای خود را هر از چند گاه تعویض نمایند.

در ازای تمامی مزایای بیان شده در فوق، باید اعتراف کرد که پروتکل پیشنهادی شکل ۴ نیز از یک نقطه ضعف نسبت به پروتکل مبتنی بر رویکرد استفاده از تابع امضای S رنج می‌برد. پروتکل پیشنهادی ما به طور ذاتی در مظان حمله متن اصلی معلوم[۲۰] و حتی حمله متن اصلی برگزیده[۲۱] قرار دارد. عبارات $e(R_B)$ (در پیام ۲) و $e(D_A R_B)$ (در پیام ۳) در دسترس دشمن قرار دارند و بنابر این با هر بار اجرای پروتکل یک جفت متن اصلی ـ متن رمز شده در اختیار وی قرار می‌گیرد تا سعی در یافتن D_A از روی آنها بنماید. دشمن با جا زدن خود به عنوان آغازگر و ارسال نانس دلخواه $e(R_A)$ قادر به دست یابی به عبارت $e(D_B R_A)$ از روی پیام ۲ می‌باشد و این شرایط حمله متن اصلی برگزیده را برای دست یابی به کلید خصوصی D_B فراهم می‌کند. پروتکل شکل ۲، به دلیل استفاده از تابع در هم و خلاصه کردن نانس‌ها قبل از اعمال تابع امضاء در مظان حملات فوق نیست. به هر حال، این ضعف در قریب به اتفاق پروتکل‌های توزیع کلید وجود دارد و به نظر می‌رسد که از اهمیت به سزایی برخوردار نیست.

در پایان این بخش مناسب است به نقاط قوت مشترک دو پروتکل نسبت به پروتکل‌های مبتنی بر رویکرد استفاده از تابع رمز E اشاره شود:

ـ در این پروتکل کلید جلسه با مسئولیت مشترک دو سوی ارتباط تولید می‌شود. در پروتکل‌هایی که غیرمتمرکز می‌باشند و طرف سوم مورد اعتمادی وجود ندارد، بهترین راه حل برای تولید کلید جلسه

همین است.

- ویژگی استثنایی این دو پروتکل آن است که چنانچه دشمن به کلیدهای خصوصی دو سوی ارتباط دست یافت، تنها قادر است با اجرای حمله فردی در میان از محتوای مذاکرات از آن پس آنها مطلع گردد و به هر حال راهی برای دست یابی به کلید جلسات قبلی آنها نداشته لذا محتوای مذاکرات قبلی همواره از چشم وی پنهان باقی می‌ماند (در حقیقت در این دو پروتکل به دلیل عدم مبادله مستقیم K_S، دست یابی به محتوای مذاکرات مستلزم اجرای همزمان حمله غیرفعال - یعنی شنود - و حمله فعال - یعنی حمله فردی در میان - است).

٤ - تکمیل فرآیند طراحی پروتکل پیشنهادی با استفاده از صدور گواهی

پروتکل هایی که تاکنون در این مقاله مورد بحث قرار گرفتند همگی بر این فرض استوارند که کاربران شبکه قابل معتبر کلیدهای عمومی را درا ختیار دارند. اما این فرض در بسیاری از کاربردهای عملی ممکن است برقرار نباشد و تحقق آن نیازمند تدارک شیوه های کارآمد است. اصلی ترین مزیت بکارگیری ایده رمزنگاری نامتقارن در عدم نیاز دو سوی ارتباط به تبادل اطلاعات قبلی نهفته است. بنابراین دست یابی به کلید عمومی معتبر کاربران حتی الامکان نباید این مزیت اساسی را کاهش دهد. در این راستا ابتدا مفهوم گواهی و ساختار آن شرح داده می شود، سپس با استفاده از ایده گواهی ، پروتکل مبتنی بر رویکرد استفاده از تابع یک طرفه جابجاپذیر اصلاح می گردد وبالاخره نحوه دست یابی هر کاربر به گواهی کلید عمومی خویش بیان می گردد.

٤ - ١ - مفهوم گواهی و ساختار آن

گواهی یک کلید عمومی عبارت از امضای آن کلید از سوی یک مرجع شناخته شده است. در واقع ، نیاز دو سوی ارتباط به اطلاع از کلید عمومی واقعی یکدیگر مستلزم وجود یک کانال معتبر - هر چند غیر پنهان - بین آن دو است و این نیاز به کمک یک مرجع شناخته شده مرتفع می گردد. ایده اساسی این است که هر کاربر پس از تولید کلید عمومی خود، یک ارتباط امن با مرکز صدور گواهی برقرار می کند و از طریق آن گواهی کلید خود را اخذ می نماید. پس از این کافی است که کاربر مذکور در حین هر ارتباط، گواهی کلید عمومی خود را در اختیار طرف دیگر آن ارتباط قرار دهد تا او با وارسی امضاء گواهی به اعتبار کلید عمومی مربوطه پی ببرد.

برای آن که ساختار گواهی را مشخص سازیم از جمله لازم است که محدود بودن عمر مفید کلیدها را از نظر دور نداریم. از آنجا که فرآیند لغو گواهی کلید به مراتب پیچیده تر از فرآیند صدور آن است به نظر می رسد که یک روش مناسب آن است که در هنگام صدور گواهی، مدت اعتبار محدودی نیز برای کلید پیش بینی شود. اگربرای نمایش گواهی کلید عمومی کاربر A از نماد $Cert_A$ استفاده شود، ساختار آن تا به اینجای بحث عبارت است از:

$$Cert_A = S_C [A, e (D_A) , T_{exp}] \qquad (۴)$$

که در آن S_C بیانگر امضای مرکز صدور گواهی است و سه متغیر مورد استفاده در آن به ترتیب شناسه کاربر، کلید عمومی وی و زمان انقضای اعتبار کلید است.

در رمزنگاری به دلایل امنیتی توصیه اکید می شود که از امضای متنی که توسط دیگران تنظیم شده خودداری گردد. رعایت این نکته برای در امان باقی ماندن سیستم امضاء از تهدیدهای متنوع ناشی از حملات متن معلوم و متن برگزیده اهمیت به سزایی دارد. برای تحقق این هدف دو ایده مختلف مطرح شده است. یکی آنکه ابتدا متن مورد نظر از یک تابع درهم یک طرفه عبور داده شود وسپس امضاء بر روی خروجی تصادفی تابع مذکور اعمال گردد. ایده دیگر این است که یک بخش تصادفی به متن مورد نظر منضم شده سپس امضاء روی آن انجام شود. اگر از ایده اخیر برای جلوگیری از حمله متن برگزیده استفاده کنیم، ساختار نهایی گواهی کلید عمومی A عبارت می شو د از:

$$Cert_A = S_C [A, e (D_A) , T_{exp} , R_C] \qquad (۵)$$

٤ - ٢ - اصلاح پروتکل پیشنهادی

فرض می شود که هر کاربر شبکه علاوه بر داشتن حداقل یک گواهی منقضی نشده از کلیدعمومی خود، فقط کلید عمومی معتبر مرکز صدور گواهی (C) ر ادر اختیار دارد. بدین ترتیب، بدون آنکه کاربران نیازی به فایل معتبر کلیدهای عمومی داشته باشند، می توان از پروتکل اصلاح شده شکل ۶ بهره گرفت.

ساختار گواهی های مورد استفاده در این پروتکل همان است که در رابطه (۵) معرفی شده است. دقت شود

که سبب پرهیز از رمز کردن گواهی ها ونیز عبارت $e (D_A R_B)$ در پیام ۳ پروتکل بی ثمر بودن این کار است (حتی اگر این عبارات رمز شوند کسی که در پی یافتن آنها است کافی است که به عنوان آغازگر پروتکل را راه‌اندازی نموده با دریافت پیام ۲ به هدف مورد نظر خود برسد).

برای آنکه بتوان قضاوت دقیقی از پیچیدگی پروتکل اصلاح شده داشت اولاً لازم است که نحوه دست یابی هر کاربر به گواهی کلید عمومی اش مشخص شود که در زیر بخش بعدی به آن پرداخته می شود و ثانیاً بایستی تابع یک طرفه دریچه دار مورد استفاده جهت تحقق سیستم امضاء معین گردد. به عنوان مثال، اگر از الگوریتم RSA برای تحقق سیستم امضاء بهره بگیریم کاربران شبکه نیاز به هیچ قابلیت اضافه ای نسبت به قبل ندارند. دقت شود که پیمانه تابع نمای گسسته مورد استفاده روی نانس ها (برای احراز یا وارسی اصالت) با پیمانه مورد استفاده برای وارسی گواهی ها متفاوت است. به خصوص با توجه به ساختار گواهی در رابطه (۵) حتی تعداد ارقام پیمانه های مذکور نیز به طور قابل ملاحظه ای با یکدیگر اختلاف دارند.

٤ـ۳ـ پروتکل اخذ گواهی از مرکز صدور گواهی

هدف این زیربخش، دست یابی به پروتکل معتبری است که طی آن کاربر A بتواند کلید عمومی مورد نظرش را به مرکز صدور گواهی C ارائه دهد و گواهی ان را اخذ نماید. مطلوب آن است که بتوان از خود پروتکل مبتنی بر تابع یک طرفه جابجاپذیر کمک گرفت. چون در این پروتکل کاربر متقاضی گواهی، نقش آغازگر را ایفا می کند، به طور طبیعی مرکز در طی پیام ۲ گواهی را برای وی خواهد فرستاد. اما این بدان معنی است که قبل از احراز اصالت آغازگر برای مرکز، گواهی کلید عمومی در اختیارش قرار گیرد. بنابراین پروتکل شکل ۴ برای انجام هدف ما مناسب نیست. در شکل ۷، گونه اصلاح شده ای از پروتکل شکل ۴، که وافی مقصود است ارائه گردیده است.

در این پروتکل D_A کلید خصوصی جاری کاربر A و D'_A کلید خصوصی جدید وی است که در طی پروتکل گواهی آن صادر می شود. مفروضات مورد نیاز پروتکل شکل ۷ عبارتند از:

- هر کاربر حداقل یک کلید عمومی جاری دارد که هنوز اعتبارش منقضی نشده است،
- کاربران کلید عمومی معتبر مرکز صدور گواهی را در اختیار دارند،
- مرکز کلید (های) عمومی منقضی نشده کاربران را در اختیار دارد.

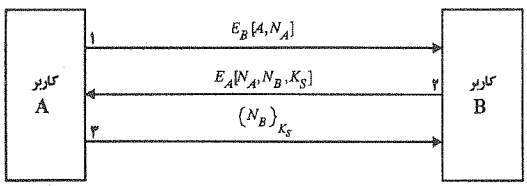
دقت شود که هدف از اجرای پروتکل شکل ۷ از سوی یک کاربر می تواند درخواست تمدید اعتبار کلید عمومی جاری یا اخذ گواهی برای یک کلید عمومی جدید باشد. در طراحی پروتکل مذکور ملاحظات زیر مدنظر بوده اند:

- برای انکه سرباری طراحی و پیاده سازی این پروتکل برای کاربران شبکه حداقل باشد حداکثر استفاده از ساختار و پارامترهای پروتکل شکل ۴ صورت گرفته است.
- همان طور که اجمالاً بیان شد امکان طراحی این پروتکل در قالب سه پیام وجود ندارد ولذا از گزینه مناسب تر تقدم احراز اصالت آغازگر نسبت به مخاطب استفاده شده است. به ویژه که در این حالت دیگر مرکز به کلید خصوصی متناظر با تابع نمای گسسته (یعنی D_C) نیازی ندارد و به جای آن از امضای خود برای احراز اصالت (در پیام ۴) بهره می گیرد.
- رمزگذاری امضاء در پیام ۴ ضرورت دارد. از آنجا که برای عدم افزایش بیشتر تعداد پیام های پروتکل از ارسال تأیید اخذ امضاء از سوی آغازگر برای مرکز صرفنظر شده است، این امکان فراهم گردیده که این امضاء هیچگاه به آغازگر نرسد. در این شرایط، اگر امضاء در اختیار دیگری قرار گیرد ممکن است منجر به سوء استفاده گردد بخصوص که خود آغازگر از وجود آن بی اطلاع است. رمز کردن امضاء امکان هر گونه بهره برداری مفید از آن را توسط سارق بالقوه سلب می کند.

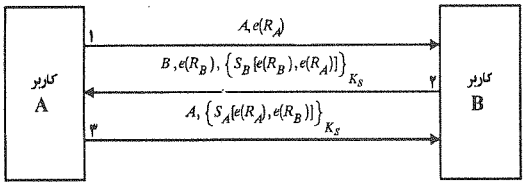
۵ـ نتیجه گیری

در این مقاله پروتکل های توزیع کلید دو سویه مبتنی بر ایده رمزنگاری نامتقارن معرفی شده مورد ارزیابی و مقایسه قرار گرفتند. این پروتکل ها براساس سه رویکرد مختلف قابل طراحی هستند: استفاده از تابع رمز E، استفاده از تابع امضای S و استفاده از تابع یک طرفه بدون دریچه جابجاپذیر. مشاهده کردیم که طراحی براساس رویکردا ول از لحاظ امنیتی قابل خدشه است. رویکرد دوم همزمان متکی به تابع لگاریتم گسسته و یک تابع یک طرفه دریچه دار دیگر است. این موضوع هم منافذ بالقوه امنیتی پروتکل را افزایش می دهد و هم بر

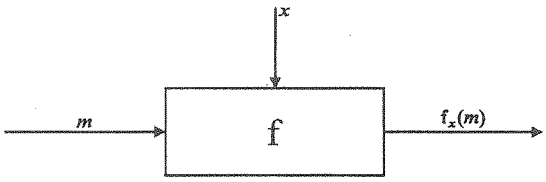
پیچیدگی پروتکل و هزینه های طراحی و پیاده سازی آن می افزاید. به هر حال پروتکلی مبتنی بر رویکرد استفاده از تابع امضای S، در صورت دقت در انتخاب تابع امضاء و مراقبت در طراحی می تواند از جنبه امنیتی بدون مشکل باشد. رویکرد سوم، که اخیراً از سوی نویسندگان این مقاله معرفی شده است تنها به تابع لگاریتم گسسته ـ به عنوان یک تابع یک طرفه بدون دریچه جابجاپذیر ـ تکیه دارد. این موضوع اولاً از دیدگاه نظری در خانواده پروتکل های مورد بحث یک گام به جلو در عرصه امنیت است و ثانیاً از دیدگاه کاربردی شرایط ایده آلی را از نظر هزینه های طراحی و پیاده سازی جلوه گر می سازد. به هر حال، نقطه ضعف پروتکل برآمده از این رویکرد آن است که در مظان حمله متن اصلی برگزیده قرار دارد. هر چند این ضعف چندان جدی به نظر نمی رسد، لیکن می توان برای کاستن هر چه بیشتر از حدّت آن به کاربران توصیه نمود که کلیدهای خصوصی و عمومی خود را در فواصل زمانی کمتری تعویض کنند. در ادامه مقاله، به منظور کاربردی تر کردن پروتکل مذکور گونه تازه ای از آن که از ایده صدور گواهی برای کلیدهای عمومی بهره می برد معرفی شد. در همین راستا پروتکل کارآمدی برای دریافت گواهی کلید عمومی از مرکز صدور گواهی ارائه گردید. بدین ترتیب نیاز به دسترسی به فایل معتبر کلیدهای عمومی کاربران شبکه مرتفع می گردد. علیرغم آنکه استفاده از ایده صدور گواهی مستلزم بکارگیری یک سیستم امضاء است اما کاربران تنها به وارسی امضاءها نیاز دارند که در صورت استفاده از الگوریتم RSA انجام وارسی امضاء به چیزی جز همان تابع نمای گسسته ـ که در اختیار کاربران بود ـ نیاز نخواهد داشت.



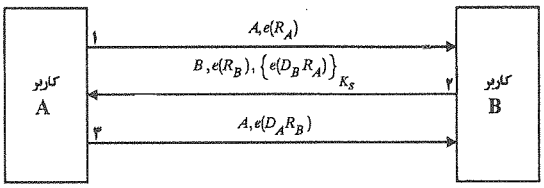
شکل (۱) پروتکل توزیع کلید مبتنی بر رویکرد استفاده از تابع رمز E.



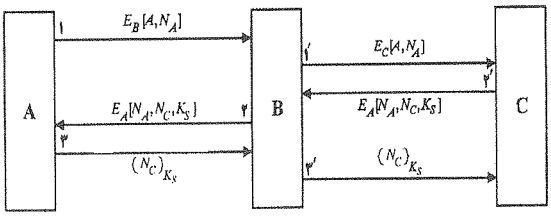
شکل (۲) پروتکل توزیع کلید مبتنی بر رویکرد استفاده از تابع امضای S.



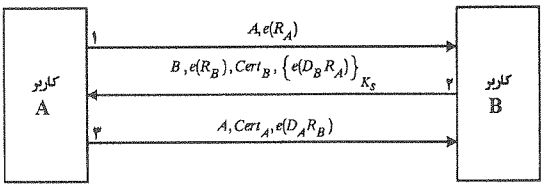
شکل (۳) معرفی تابع یک طرفه f .



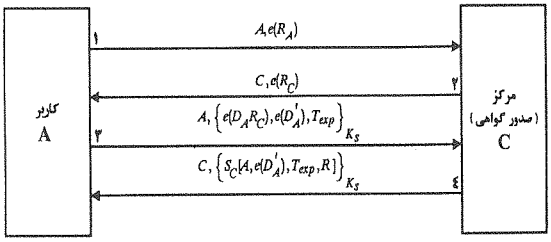
شکل (۴) پروتکل توزیع کلید مبتنی بر رویکرد استفاده از تابع یک طرفه جابجاپذیر.



شکل (۵) سناریوی رخنه در پروتکل مبتنی بر رویکرد استفاده از تابع رمز E.



شکل (۶) پروتکل اصلاح شده مبتنی بر رویکرد استفاده از تابع یک طرفه جابجاپذیر.



شکل (۷) پروتکل پیشنهادی جهت اخذ گواهی کاربران از مرکز صدور گواهی.

زیرنویس‌ها

1 - asymmetric
2 - one - way function
3 - key distribution center
4 - certificate
5 - public key cryptosystem
6 - trapdoor one - way function
7 - private key
8 - signature
9 - session key
10 - nonce
11 - authentication
12 - replay attack
13 - enemy (intruder)
14 - discrete exponentor
15 - strong prime
16 - primitive root
17 - Diffie - Hellman
18 - man - in - the - middle attack
19 - one - way hash function
20 - known plaintext attack
21 - chosen plaintext attack

مراجع

[۱] مهدی برنجکوب، محمدرضا عارف و حسین سعیدی، «یک پروتکل توزیع کلید مبتنی بر تابع یک طرفه»، ششمین کنفرانس مهندسی برق ایران، دانشگاه خواجه نصیر الدین طوسی، ۱۳۷۷.

[2] Diffie, W. and M. E. Hellman, "New Direction in Cryptography", IEEE Transaction on Information Theory, V. 22, n. 6, 1976.

[3] Tanenbaum, A. S., "Computer Networks", 3rd Edition, P. 612, Prentice Hall, 1996.

[4] Diffie, W., P. C. Van Oorschot and M. J. Wiener, "Authentication and Authenticated Key Exchanges", Designs, Codes and Cryptography, V. 2, P. 107 - 125, 1992.