ششمین کنفرانس بین المللی
ترکیبیات، رمزنگاری، علوم کامپیوتر و محاسبات
۲۶ و ۲۷ آبان ۱۴۰۰

احراز هویت در رمزنگاری RSA

مریم رضایی کاشی[۱]
دانشگاه کاشان، کاشان، ایران

مجتبی بهرامیان
دانشگاه کاشان، کاشان، ایران

چکیده

در سامانه رمزنگاری RSA گیرنده، کلید عمومی را تولید کرده و منتشر می‌کند. حال فرض کنید شخصی پیام خود را با استفاده از کلید گیرنده رمز کرده و قصد ارسال آن را برای گیرنده داشته باشد. از آنجایی که ممکن است در کانال ناامن شخص دیگری در جایگاه گیرنده قرار بگیرد برای حفظ امنیت در تبادل پیام باید فرستنده از اصالت گیرنده اطمینان حاصل کند. برای این منظور در این مقاله یک پروتکل اثبات دانش صفر برای احراز هویت در سامانه RSA پیشنهاد می‌کنیم که امنیت آن بر سختی مسائل لگاریتم گسسته و تجزیه اعداد بنا شده است.

واژه‌های کلیدی: سامانه رمزنگاری RSA ، احراز هویت، پروتکل اثبات دانش صفر، تجزیه‌ی اعداد، لگاریتم گسسته.
[2010]: 11T71, 11A51

۱ مقدمه

در یک سامانه رمزنگاری ممکن است یک مهاجم در جایگاه یکی از طرفین قرار بگیرد و در فرایند تبادل کلید و یا تبادل پیام اختلال ایجاد کند. برای پیشگیری از چنین مشکلی سامانه‌های امضای دیجیتال و احراز هویت به وجود آمدند. پیامی که حاوی امضای فرستنده باشد می‌تواند این اطمینان رابه گیرنده بدهد که این پیام از سمت شخص فرستنده ارسال شده است، همچنین اگر با یک روش احراز هویت یکی از طرفین اصالت خود را برای طرف مقابل اثبات کند، طرف مقابل مطمئن می‌شود که شخص دیگری در سامانه مورد نظر شرکت نکرده است. یکی از روش‌های احراز هویت استفاده از یک پروتکل اثبات دانش-صفر[۲] است[۵، ۸]. در یک پروتکل اثبات دانش-صفر اثبات کننده مطلبی را برای تأیید کننده به اثبات می‌رساند بدون اینکه اطلاعات بیشتری را منتشر کند و این ویژگی بسیار در حفظ حریم خصوصی اهمیت دارد.

گلدواسر[۳]، میکالی[۴] و راکف[۵] در سال ۱۹۸۲ پروتکل اثبات دانش-صفر را معرفی کردند که خیلی زود به عنوان ابزاری برای حفظ حریم خصوصی و امنیت در رمزنگاری شناخته شد[۶]. در یک پروتکل دانش-صفر، اثبات کننده، تأیید کننده را در مورد اطلاع از مطلبی متقاعد می‌کند. پس از اجرای پروتکل، تأیید کننده متقاعد می‌شود که اثبات کننده آن مطلب را می‌داند، بدون اینکه اطلاعات بیشتری درباره‌ی مطلب به دست آورد و تأیید کننده نمی‌تواند در نقش اثبات کننده قرار گیرد. اثبات کننده در واقع دانشِ صفر را انتقال می‌دهد.

یک مثال برای کاربرد اثبات دانش-صفر استفاده از آن برای ورود به یک وب سایت است. فرض کنید کاربری برای ورود به یک وب سایت نیاز به یک گذرواژه داشته باشد. به صورت معمول سرور یک نسخه‌ی کد شده از رمز عبور را در اختیار دارد و چنانچه رمز عبور با آن مطابقت داشته باشد، کاربر اجازه ورود دارد. اگر مهاجمی به این سامانه حمله کند، گذرواژه‌ی آن کاربر که خصوصی است به خطر می‌افتد و ممکن است در دسترس بیگانه قرار گیرد. با استفاده از اثبات دانش-صفر بدون در معرض خطر قرار گرفتن اطلاعات خصوصی، هویت کاربر برای سرور به اثبات می‌رسد.

[۱] سخنران

[2] Zero-Knowledge
[3] Goldwasser
[4] Micali
[5] Rackoff

حفظ امنیت تبادل کلید در سامانه‌های رمزنگاری از کاربردهای دیگر اثبات دانش-صفر است [۱۰]. برای مثال فرض کنید در یک سامانه رمز RSA گیرنده $N = pq$ را که در آن p و q دو عدد اول متمایز هستند به عنوان بخشی از کلید عمومی منتشر کرده باشد. فرض کنید شخصی به عنوان فرستنده‌ی پیام در یک کانال ناامن بخواهد بداند که این کلید متعلق به طرف مقابل است. اگر طرف مقابل بتواند با استفاده از اثبات دانش-صفر ثابت کند که p و q را می‌داند می‌تواند اصالت خود را برای فرستنده ثابت کند.

جان کمنیش[6] و مارکوس میکلس[7] در سال ۱۹۹۸، یک پروتکل دانش-صفر ارائه کردند که ثابت می‌کند یک عدد پیمانه RSA حاصل‌ضرب دو عدد اول امن است [۲]. رزاریو جنرو [8] و همکارانش در همان سال، شبیه چنین طرحی را برای حالتی که عدد پیمانه RSA حاصل‌ضرب دو عدد اول شبه امن است، معرفی کردند [۴]. عدد اول p عدد اول امن نامیده می‌شود اگر $(p-۱)/۲$ نیز یک عدد اول باشد، و شبه امن نامیده می‌شود اگر $(p-۱)/۲$ توانی از یک عدد اول باشد. تا کنون، طبق بررسی‌های انجام شده، پروتکل دانش-صفر با هدف دانستن عوامل اول یک پیمانه‌ی RSA ارائه نشده است. در این مقاله یک پروتکل دانش-صفر برای دانستن عوامل اول یک پیمانه‌ی RSA معرفی می‌کنیم که امنیت آن بر مبنای سختی مسائل تجزیه اعداد و لگاریتم گسسته است.

در ادامه‌ی این مقاله ابتدا به بیان مفاهیم و قضایای مورد نیاز از نظریه اعداد پرداخته، سپس تعریف اثبات دانش-صفر و یک مثال در این زمینه را بیان می‌کنیم. در نهایت پس از مروری بر سامانه رمزنگاری RSA به ارائه یک پروتکل دانش-صفر برای اطلاع از عوامل اول یک پیمانه RSA می‌پردازیم.

۲ احکام اصلی

۱.۲ مفاهیمی از نظریه اعداد

فرض کنید G یک گروه، $g \in G$ و $h \in \langle g \rangle$. لگاریتم گسسته‌ی h در مبنای g، عدد صحیح k است که به ازای آن $g^k = h$. مسأله‌ی یافتن k با استفاده از g و h به طوری که $h = g^k$ را مسأله‌ی لگاریتم گسسته در G می‌گوییم. برای عدد صحیح مرکب n، مسأله‌ی یافتن عوامل اول n را مسأله‌ی تجزیه‌ی عدد n می‌نامیم.

در رمزنگاری، الگوریتمی که در آن، تعداد عملیات لازم تابعی چندجمله‌ای بر حسب اندازه‌ی ورودی باشد، یک الگوریتم کارا نامیده می‌شود. عمل توان رسانی در گروه‌ها و ضرب اعداد صحیح دارای الگوریتم‌های کارا است، اما برای حل عکس این دو مسأله یعنی حل مسأله‌ی لگاریتم گسسته و مسأله‌ی تجزیه‌ی اعداد تا کنون الگوریتمی کارا پیدا نشده است و از مسایل دشوار ریاضی است. بنابراین اساس طراحی برخی سامانه‌های رمزنگاری از جمله سامانه‌ی رمز RSA [۹] و پروتکل تبادل کلید دیفای-هلمن [۳] این دو مسأله است.

فرض کنید n عددی طبیعی باشد و $\mathbb{Z}_n = \{۰, ۱, ۲, \ldots, n-۱\}$. تعریف می‌کنیم $\mathbb{Z}_n^* = \{a \in \mathbb{Z}_n \mid \gcd(a,n) = ۱\}$ و مرتبه‌ی $\mathbb{Z}_n^*$ را با نماد $\varphi(n)$ نشان می‌دهیم. تابع φ را نیز تابع فی اویلر می‌نامیم. اگر p و q اعداد اول متمایز باشند، آنگاه $\varphi(pq) = (p-۱)(q-۱)$. قضیه اویلر بیان می‌کند که اگر n یک عدد طبیعی باشد و عدد صحیح a چنان باشد که $\gcd(a,n) = ۱$، آنگاه

$$a^{\varphi(n)} \equiv ۱ \pmod{n}.$$

فرض کنید n یک عدد طبیعی باشد. گوییم عدد صحیح a در پیمانه‌ی n وارون حسابی دارد، هرگاه عدد صحیح b چنان موجود باشد که $ab \equiv ۱ \pmod{n}$. در این صورت b وارون حسابی a در پیمانه‌ی n نامیده می‌شود. به راحتی می‌توان دید، a در پیمانه‌ی n دارای عکس حسابی است، اگر و تنها اگر $\gcd(a,n) = ۱$. در چنین حالتی بنا به قضیه‌ی اویلر، در پیمانه‌ی n، $a^{\varphi(n)-۱}$ یک عکس حسابی a است. اگر a یک عدد صحیح متباین با n باشد مرتبه‌ی a در پیمانه‌ی n، کوچکترین عدد طبیعی k است که $a^k \equiv ۱ \pmod{n}$. در این صورت می‌نویسیم $O_n(a) = k$. همچنین بنا به قضیه‌ی اویلر، $O_n(a)$ موجود است و بعلاوه $O_n(a) \leq \varphi(n)$. حال فرض کنید $O_n(a) = k$. در این صورت $a^h \equiv ۱ \pmod{n} \leftrightarrow k|h$. اگر برای عدد صحیح g، داشته باشیم $O_n(g) = \varphi(n)$، عدد صحیح g یک ریشه‌ی اولیه در پیمانه‌ی n نامیده می‌شود. اگر p یک عدد اول باشد، آنگاه یک ریشه‌ی اولیه در پیمانه‌ی p موجود است. برای مطالعه‌ی بیشتر در زمینه نظریه اعداد می‌توان به مراجع [۱، ۷] مراجعه کرد.

[6] Jan Camenisch
[7] Markus Michels
[8] Rosario Gennaro

۲.۲ اثبات دانش-صفر

در یک پروتکل اثبات دانش-صفر اثبات کننده گزاره‌ای را برای تأیید کننده ثابت می‌کند به طوری که تأیید کننده در مورد آن گزاره قانع می‌شود اما خود نمی‌تواند آن را ثابت کند. به طور معمول یک پروتکل اثبات دانش صفر از یک سری مرحله تشکیل شده است که چندین بار توسط طرفین تکرار می‌شود و همواره اثبات کننده باید به چالش تأیید کننده پاسخ صحیح بدهد. در واقع بعد از توافق روی یک سری داده‌ها و تبادل اطلاعات لازم، تأیید کننده اثبات کننده را به چالشی دعوت می‌کند که اثبات کننده با واکنش درست در مقابل این چالش ثابت می‌کند که با یک احتمال ثابتی گزاره مورد نظر درست است با تکرار این فرایند چالش-واکنش احتمال اشتباه بودن گزاره و پاسخ صحیح دادن به چالش در تمام فرایندها، به نزدیک صفر می‌رسد و در نتیجه تأیید کننده قانع می‌شود که گزاره درست است.

۳.۲ سامانه رمزنگاری RSA

ریوست[۹]، شامیر[۱۰] و آدلمن[۱۱] در سال ۱۹۷۸ سامانه‌ی رمزRSA را معرفی کردند [۹]. این سامانه در بسیاری از تبادلات الکترونیکی به کار می‌رود و در صورت استفاده درست با کلیدهای به اندازه کافی بزرگ هم‌چنان امن به نظر می‌رسد. امنیت سامانه‌ی رمز RSA بر سختی تجزیه‌ی اعداد مرکب بزرگ به شکل $N = pq$ بنا شده است. بنابراین در رمزنگاری، اعداد مرکب به شکل $N = pq$ که در آن p و q اعداد اول متمایزاند، از اهمیت بالایی برخوردارند.

سامانه RSA با هدف ارسال پیام m از سوی فرستنده برای گیرنده به این صورت است که ابتدا شخص گیرنده‌ی پیام دو عدد اول بزرگ p و q را انتخاب کرده و عدد $N = pq$ را محاسبه می‌کند. سپس $\varphi(N) = (p-1)(q-1)$ را در نظر گرفته و عدد e را طوری انتخاب می‌کند که $\gcd(e, \varphi(N)) = 1$. فرض کنید d عکس حسابی e به پیمانه $\varphi(N)$ باشد. گیرنده، d را به عنوان کلید خصوصی نزد خود حفظ کرده و (N, e) را به عنوان کلید عمومی منتشر می‌کند(تولید کلید). حال فرستنده m را به وسیله‌ی کلید عمومی و به صورت $c \equiv m^e \pmod N$ رمز کرده و آن را برای گیرنده‌ی پیام ارسال می‌کند. در نهایت گیرنده‌ی پیام پس از دریافت c، با استفاده از کلید خصوصی خود یعنی d، پیام را به صورت $m \equiv c^d \pmod N$ رمزگشایی می‌کند. توجه کنید که

$$c^d \equiv (m^e)^d \equiv m^{ed} \equiv m \pmod N.$$

در این سامانه اگر مهاجم بتواند N را تجزیه کند، آنگاه با داشتن p و q و محاسبه‌ی $\varphi(N)$ به کلید خصوصی d دسترسی پیدا کرده و می‌تواند پیام را رمزگشایی کند. بنابراین امنیت این سامانه به سختی تجزیه‌ی اعداد بنا شده است.

در این سامانه، برای این که فرستنده از اصالت گیرنده مطمئن شود، می‌تواند پس از دریافت کلید عمومی از گیرنده بخواهد تا با استفاده از پروتکل دانش-صفر، او را در مورد اطلاع از عوامل اول N متقاعد کند. در این صورت بدون این که امنیت کلید خصوصی به خطر بیفتد، فرستنده از اصالت گیرنده و کلید عمومی اطمینان حاصل می‌کند.

۴.۲ یک پروتکل اثبات دانش-صفر برای اطلاع از عوامل اول یک عدد

در این بخش، یک پروتکل دانش-صفر ارائه می‌کنیم که در آن، اثبات کننده، تأیید کننده را قانع می‌کند که عوامل اول عدد N را می‌داند. در این بخش فرض بر این است که N یک عدد پیمانه RSA یعنی حاصلضرب دو عدد اول بزرگ و متمایز است که آن‌ها را با p و q نشان می‌دهیم. امنیت این طرح بر اساس سختی تجزیه‌ی اعداد بزرگ و سختی مسأله‌ی لگاریتم گسسته است. طرح پیشنهادی به این صورت است که ابتدا اثبات کننده عدد اول L را که $L > N$ و $\gcd(L-1, N) = 1$ انتخاب می‌کند. بعلاوه ریشه اولیه‌ی g از L را انتخاب کرده و (L, g) را منتشر می‌کند. سپس طرفین مراحل زیر را چندین بار تکرار می‌کنند:

۱. اثبات کننده $E_0 \equiv g^p \pmod L$ و $E_1 \equiv g^q \pmod L$ را محاسبه کرده و برای تأیید کننده ارسال می‌کند.

۲. تأیید کننده بررسی می‌کند که $E_0, E_1 \not\equiv g, g^N \pmod L$.

همه این روابط باید برقرار باشد، اگر یکی از این روابط برقرار نباشد، تأیید کننده به این نتیجه می‌رسد که اثبات کننده تجزیه‌ی N را نمی‌داند و عوامل مد نظر اثبات کننده ۱ و N هستند. در این صورت تأیید کننده متقاعد نمی‌شود و فرایند ادامه پیدا نخواهد کرد.

[9] Rivest
[10] Shamir
[11] Adleman

۳. تأیید کننده $t \in \{۰, ۱\}$ را برای اثبات کننده می‌فرستد. (چالش)

۴. اثبات کننده $c_۰, c_۱ \in \mathbb{Z}_{L-۱}^*$ را انتخاب کرده و اعداد

$$\begin{aligned} y_۰ &\equiv p^t c_۱ \pmod{L-۱} \\ y_۱ &\equiv q^{۱-t} c_۰ \pmod{L-۱} \\ K_۰ &\equiv g^{c_۰} \pmod{L} \\ K_۱ &\equiv g^{c_۱} \pmod{L} \end{aligned}$$

را برای تأیید کننده ارسال می‌کند (واکنش).

۵. تأیید کننده برای $t \in \{۰, ۱\}$ درستی روابط $\gcd(y_t, L-۱) = ۱$ و $E_t^{y_۰ y_۱} \equiv K_t^{y_i N} \pmod{L}$ را مورد بررسی قرار می‌دهد. در صورتی که این روابط برقرار نباشد، تأیید کننده بیان می‌کند که قانع نشده است.

در هر تکرار این دوره، اثبات کننده باید به چالش تأیید کننده پاسخ درست دهد.

توجه کنید که اثبات کننده باید در هر دوره مقادیر $c_۰$ و $c_۱$ را تغییر دهد. زیرا در صورت استفاده از مقادیر ثابت $c_۰$ و $c_۱$ در دو دوره با چالش‌های متفاوت، عوامل اول N به دست می‌آید. به این صورت که تأیید کننده با ارسال چالش $t = ۰$، مقدار $c_۱ \pmod{L}$ و با ارسال چالش $t = ۱$، مقدار $pc_۱ \pmod{L}$ را به دست می‌آورد و p را در پیمانه‌ی L محاسبه می‌کند. از آنجایی که $p < N < L$، عامل p و در نهایت تجزیه‌ی N به دست می‌آید.

از آنجایی که یافتن یک عدد اول به اندازه‌ی کافی بزرگ سخت است، در چنین الگوریتم‌هایی به طور معمول از یک عدد اول L در تمام دوره‌ها استفاده می‌شود. از طرفی L باید طوری انتخاب شود که بتوان به تعداد کافی مقادیر $c_۰$ و $c_۱$ را از $\mathbb{Z}_{L-۱}^*$ انتخاب کرد. به عبارت دیگر $\#\mathbb{Z}_{L-۱}^* = \varphi(L-۱)$ باید به اندازه کافی بزرگ باشد.

امنیت پروتکل

امنیت اثبات کننده زمانی به خطر می‌افتد که تأیید کننده بتواند مقادیر p و q را به دست آورد. برای این منظور او باید عدد N را تجزیه کند و یا اینکه با حل یک مسأله لگاریتم گسسته با استفاده از $E_۰$ و $E_۱$، p و q را به دست آورد. بنابراین امنیت اثبات کننده بر سختی مسائل لگاریتم گسسته و تجزیه اعداد بنا شده است.

امنیت تأیید کننده زمانی به خطر می‌افتد که اثبات کننده بدون اینکه p و q را بداند بتواند به تمام چالش‌ها پاسخ مثبت بدهد اما تأیید کننده در گام ۲ راستگویی اثبات کننده را مورد بررسی قرار می‌دهد. در این گام برای تأیید کننده به اثبات می‌رسد که p و q مورد نظر اثبات کننده مخالف ۱ و N است. همچنین در گام نهایی داریم

$$\begin{aligned} & E_i^{c_۰ c_۱} \equiv K_t^{c_t N} \pmod{L} \\ \longrightarrow\ & g^{pqc_۰c_۱} \equiv g^{Nc_۰c_۱} \pmod{L} \\ \longrightarrow\ & g^{c_۰c_۱(pq-N)} \equiv ۱ \pmod{L} \\ \longrightarrow\ & c_۰c_۱(pq-N) \equiv ۰ \pmod{L-۱} \\ \longrightarrow\ & L-۱ | c_۰c_۱(pq-N) \\ \longrightarrow\ & L-۱ | pq-N \end{aligned}$$

حال با توجه به این که $L > N$، بنابراین $L-۱ > pq-N$ و در نتیجه $N = pq$. بنابراین با بررسی درستی روابط در گام پایانی تأیید کننده قانع می‌شود که اثبات کننده مقادیر p و q را می‌داند. در ادامه مثالی برای روشن شدن مطلب بیان می‌کنیم:

مثال ۱.۲. عدد $N = ۲۰۵۹ = ۷۱ \times ۲۹ = pq$ را در نظر بگیرید. فرض کنید اثبات کننده ادعا کند که p و q را می‌داند. او برای اثبات این موضوع برای تأیید کننده به روش اثبات دانش صفر ابتدا عدد اول $L = ۱۴۹$ و ریشه اولیه‌ی $g = ۳$ از آن را انتخاب کرده و منتشر می‌کند. سپس طرفین مراحل زیر را چندین بار تکرار می‌کنند.

۱. اثبات کننده $E_۰ \equiv g^p \equiv ۱۱ \pmod L$ و $E_۱ \equiv g^q \equiv ۱۲۸ \pmod L$ را برای تأیید کننده ارسال می‌کند.

۲. تأیید کننده $g^N \equiv ۱۳ \pmod L$ را محاسبه کرده و سپس روابط $E_۰, E_۱ \not\equiv ۳, ۱۳ \pmod L$ را مورد بررسی قرار می‌دهد.

۳. تأیید کننده $t \in \{۰, ۱\}$ را برای اثبات کننده می‌فرستد. (فرض کنید $t = ۱$).

۴. اثبات کننده $c_۰ = ۷۲ \in \mathbb{Z}^*_{L-۱}$ و $c_۱ = ۵۳ \in \mathbb{Z}^*_{L-۱}$ را (به طور تصادفی) انتخاب می‌کند سپس مقادیر $y_۰ \equiv p^t c_۱ \equiv ۶۳ \pmod{L-۱}$، $y_۱ \equiv q^{۱-t} c_۰ \equiv ۷۲ \pmod{L-۱}$ و $K_۰ \equiv g^{c_۰} \equiv ۳۳ \pmod L$ و $K_۱ \equiv g^{c_۱} \equiv ۵۷ \pmod L$ را محاسبه کرده و آن‌ها را برای تأیید کننده ارسال می‌کند.

۵. تأیید کننده برای $t \in \{۰, ۱\}$ درستی روابط $\gcd(L-۱, y_t) = ۱$ و $E_t^{y_۰ z_۱} \equiv K_t^{y_i N} \pmod L$ را بررسی می‌کند. در این مثال به ازای $t = ۱$،

$$E_۱^{y_۰ y_۱} \equiv ۸۵ \pmod L,$$
$$K_۱^{y_۱ N} \equiv ۸۵ \pmod L.$$

مراجع

[1] D. M. Burton, *Elementary Number Theory*, 4th ed. New York:McGraw-Hill, 1997.

[2] J. Camenisch, M. Michels, Proving in Zero-Knowledge that a Number is the Product of Two Safe Primes, *International Conference on the Theory and Applications of Cryptographic Techniques*, EUROCRYPT, LNCS, **1592**, (1998), 107–122.

[3] W. Diffie, M. E. Hellman, New Directions in Cryptography, *IEEE Transactions on Information Theory*, **IT-22**(6), (1976), 644–654.

[4] R. Gennaro, D. Micciancio, T. Rabin, An Efficient Non-Intractive Statistical Zero-Knowledge Proof System for Quasi-Safe Prime Products, *Proceedings of the 5th ACM conference on Computer and communications security*, (1998), 67–72.

[5] A. Giani, Identification with Zero Knowledge Protocols, *Security Essentials - Version 1.2e*, (2001).

[6] S. Goldwasser, S. Micali, C. Rackoff, The knowledge complexity of interactive proof systems, *SIAM Journal on Computing, Philadelphia: Society for Industrial and Applied Mathematics*, **18**(1), (1989), 186–208.

[7] J. Hoffstein, J. Pipher, J. H. Silverman, *An Introduction to Mathematical Cryptography*, Springer Science+Business Media, LLC, 2008.

[8] A. M. Jaafar, A. Samsudin, Visual Zero-Knowledge Proof of Identity Scheme: A New Approach, *Second International Conference on Computer Research and Development*, (2010), 205–212.

[9] R. Rivest, A. Shamir, L. Adleman, A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. *Communications of the ACM.* **21**(2), (1978), 120–126.

[10] D. R. Stinson, J. Wut, A Zero-Knowledge Identification and Key Agreement Protocol, *David R. Cheriton School of Computer Science*, University of Waterloo 200 University Ave. W., Waterloo, Ontario, N2L 3G1, Canada dstinson, j32wu@uwaterloo.ca, (2007).

پست الکترونیکی: mrezaei.k@grad.kashanu.ac.ir
پست الکترونیکی: bahramianh@kashanu.ac.ir