International conference on Recent Research in Science and Technology
Kerman
In cooperation with
Payame Noor university and Farhangian university

Sponsored and Indexed by
CIVILICA
We Respect the Science
The conference is sponsored by CIVILICA, and articles will be indexed in CIVILICA and the National Consortium.

بررسی شرایط لازم برای موفقیت الگوریتم حمله ی وینر به کلید محرمانه در سیستم رمزنگاری آر.اس.ای

مهسا صادقی[۱]*، شاهد مشهودی[۲]

۱- دانشکده علوم ریاضی و کامپیوتر، دانشگاه خوارزمی، تهران، ایران. Mahsa.Sadeghi89@Gmail.com
۲- گروه ریاضی، دانشکده علوم پایه، دانشگاه آزاد اسلامی واحد رشت، ایران. Shahed.Mashhoodi@Gmail.com

چکیده

هدف مقاله ی حاضر، بررسی شرایطی است که در صورت برقراری آنها، الگوریتم حمله ی وینر می تواند با توجه به کلید عمومی در رمزنگاری آر.اس.ای، موفق به محاسبه ی دقیق کلید محرمانه شود، البته طبیعتا در هر مرحله باید درستی جواب به دست آمده، کنترل شود. واضح است که دلیل طراحی حملات مختلف به سیستمهای رمزنگاری و بررسی شرایط موفقیت آنها، اطمینان از نحوه ی انتخاب کلیدهای محرمانه و عمومی هنگام رمزنگاری می باشد تا بدینوسیله از شنود اطلاعات مخابره شده یا دسترسی افراد غیرمجاز به بانک های اطلاعاتی آفلاین و آنلاین، جلوگیری گردد. برای این منظور، ابتدا سختی تجزیه ی یک عدد طبیعی بسیار بزرگ، به صورت حاصلضرب دو عدد اول بزرگ، را به عنوان شاکله ی اصلی رمزنگاری آر.اس.ای معرفی می کنیم، چون سختی حدس عامل های اول بزرگ، باعث زمان بر شدن محاسبات مهاجمان هنگام تلاش برای شکستن رمز می شود. سپس به بررسی ساختار الگوریتم حمله ی وینر و ارتباط آن با مولفه های رمزنگاری آر.اس.ای و شرایط موفقیت این حمله، می پردازیم.

واژه‌های کلیدی: رمزنگاری آر.اس.ای، حمله ی وینر، شرایط موفقیت حمله، کلید محرمانه، کلید عمومی

۱- مقدمه

گسترش روش های ریاضی برای رمزنگاری در قرن اخیر سرعت زیادی گرفته است [۱]، اما یکی از مهمترین این پیشرفتها زمانی رخ داد که وایتفیلد دیفی و مارتین هلمن، از دانشکده ی مهندسی برق الکترونیک و آزمایشگاه هوش مصنوعی دانشگاه استنفورد، به واسطه ی معرفی توابع دریچه‌ای توانستند ایده ی رمزنگاری با کلید عمومی را برای تامین امنیت سیستمهای پرمخاطب ابداع کنند [۲]. مدتی بعد با الهام از این ایده، رونالد رایوست، آدی شمیر و لئونارد آدلمن، از آزمایشگاه تحقیقاتی علوم کامپیوتر در موسسه فناوری ماساچوست، موفق به دریافت جایزه ی تورینگ (معادل نوبل در رشته ی کامپیوتر) از انجمن ماشینهای محاسباتی (ACM) به خاطر معرفی یک روش جدید برای رمزنگاری، با بهره‌گیری از فرآیند به توان رساندن به پیمانه ی n در حساب همنهشتی روی میدان های متناهی شدند که بعدها به نام خودشان به صورت مخفف به آر.اس.ای معروف شد و امروزه یکی از پرکاربردترین الگوریتم‌های رمز به شمار می‌رود [۳]. این سیستم برای تأمین امنیت در مواردی مانند کارت اعتباری خرید، ایمیل و ورود مجازی به شبکه‌ها طراحی شده و اینترنت بدون آن نمی‌تواند کارایی فعلی را داشته باشد. اما امنیت آر.اس.ای در حقیقت بر پایه‌ی دشواری حل مسئله‌ی تجزیه به عامل‌های اول در اعداد بزرگ است، بنابراین طبیعی است که با هدف آزمودن امنیت آر.اس.ای، حملات زیادی روی آن صورت گرفته باشد. اگرچه هیچ حمله‌ای نتوانسته کاملاً رمز RSA را بشکند [۴]، اما محدودیت‌های زیادی در انتخاب مولفه ها و کلیدهای آر.اس.ای وضع شده است.

یکی از بهترین این حملات منسوب به مایکل وینر، از شرکت شبکه های کامپیوتری نورتل کانادا، می باشد که نشان می‌دهد انتخاب یک مقدار بسیار کوچک برای کلید محرمانه در دستگاه آر.اس.ای، چگونه می تواند در اثر این حمله، به راحتی منجر به شکسته شدن رمز و ناامنی در سیستم شود، به طوری که همه ی اطلاعات سرّی رمز شده، در زمان کوتاهی قابل دسترسی برای افراد غیرمجاز شوند [۵]. البته حملات دیگری هم به آر.اس.ای انجام شده، اما با اعمال محدودیت‌هایی در انتخاب پارامترهای رمزنگاری و کلیدهای آر.اس.ای، هنوز هم می‌توان از امنیت آن مطمئن بود [۶]. در واقع امروزه تحقیقات تحلیل رمز بیشتر روی تقریب و نوع محدودیت‌های لازم برای پارامترهای رمزنگاری، با هدف کاهش نفوذپذیری سیستم‌های رمزنگاری، متمرکز شده است [۷]. در این مقاله پس از مرور کوتاهی بر رمزنگاری آر.اس.ای، الگوریتم حمله ی وینر و شرایط موفقیت این حمله را برای پیش بینی امکان آسیب پذیری آر.اس.ای بررسی می کنیم.

۲- پیش نیازها

در این بخش به معرفی پیش نیازها و برخی اصطلاحات و نمادگذاری ها در زمینه ی رمزنگاری می پردازیم.

۲-۱- نحوه ی انتخاب کلیدها در رمزنگاری آر.اس.ای

برای رمزنگاری آر.اس.ای، باید ابتدا دو عدد اول نسبتا بزرگ مانند p و q را انتخاب کنیم و حاصلضربشان را با n نمایش دهیم، به طوری که $n=pq$. واضح است که تابع فی اویلر $\phi(n)=(p-1)(q-1)$ مشخص کننده ی مرتبه ی گروه ضربی متناهی $\mathbb{Z}_n^*$ خواهد بود و اعداد متناظر با متن پیام ساده (مثلا m) و متن پیام رمز شده (مثلا r)، باید همگی از میان اعضای $\mathbb{Z}_n^*$ در نظر گرفته شوند. همچنین $\equiv_n$ به عنوان نماد حساب همنهشتی به پیمانه ی n در نظر گرفته می شود. اکنون «کلید عمومی» e باید طوری انتخاب شود که $(\phi(n),e)=1$، زیرا در این صورت طبق قضیه ای در نظریه ی اعداد، به ازای e یک عضو وارون به پیمانه ی $\phi(n)$ وجود خواهد داشت که معمولا با d نمایش داده می شود و «کلید محرمانه» نامیده می شود، پس داریم $ed\equiv_{\phi(n)}1$، یعنی $\exists k, ed=1+k\phi(n)$، که از اینجا d توسط الگوریتم تقسیم اقلیدس قابل محاسبه است. معمولا برای مقدار e، یک عدد اول کوچکتر از $\phi(n)$ در نظر گرفته می شود. لازم به تاکید است که چهار پارامتر $\{d,\phi(n),p,q\}$ را به عنوان مجموعه کلیدهای محرمانه، باید مخفی نگه داریم، ولی دو پارامتر $\{e,n\}$ را به عنوان مجموعه کلیدهای عمومی، می‌توانیم به همه مخاطبین سیستم اعلام کنیم.

۲-۲- رمزنگاری با کلید عمومی و رمزخوانی با کلید محرمانه

اکنون برای دو عدد $r,m\in\mathbb{Z}_n^*$ به عنوان پیام ساده و پیام رمز شده، تابع رمزنگار $E(m)=r$ را با ضابطه ی $m^e\equiv_n r$ تعریف می کنیم، به این مفهوم که پیام ساده ی m، پس از رسیدن به توان کلیدعمومی e، در پیمانه ی n، تبدیل به پیام رمزشده ی r می‌شود. همچنین برای رمزخوانی مجاز باید گیرنده ی پیام در مقصد، کلید محرمانه d را در اختیار داشته باشد و همانطور که گفتیم $ed=1+k\phi(n)$، از طرفی طبق قضیه‌ای از اویلر داریم $m^{p-1}\equiv_p 1$ و $m^{q-1}\equiv_q 1$، پس $m^{\phi(n)}\equiv_n 1$، در نتیجه $m^{k\phi(n)+1}\equiv_n m$، بنابراین $(m^e)^d\equiv_n m$، اکنون با توجه به تابع رمزنگار $m^e\equiv_n r$، داریم: $(r)^d\equiv_n m$ که رابطه ی اخیر در واقع همان ضابطه ی مورد نیاز ما برای تعریف تابع رمزخوانی $D(r)=m$ می باشد، به این مفهوم که باید پیام رمزشده ی r را به پیمانه ی n، به توان d رساند، تا پیام ساده محرمانه m به دست آید [۳].

واضح است که تابع رمزنگاری E تعریف شده در آر.اس.ای، دارای ویژگی های توابع دریچه ای دیفی-هلمن می باشد، به این مفهوم که رمزنگاری با آن ساده است، ولی رمزگشایی غیرمجاز بدون داشتن کلید محرمانه، عملی سخت و نیازمند صرف وقت و هزینه ی زیاد است، که این تلاش برای دسترسی غیرمجاز را اصطلاحا حمله برای شکستن رمز نیز می نامند و سختی کشف کلید محرمانه d، به دلیل سختی تجزیه ی n بزرگ به عامل های اول بزرگ p و q می باشد، زیرا با یافتن p و q، به سادگی $\phi(n)$ محاسبه می شود و سپس با توجه به داشتن کلید عمومی e، به راحتی کلید محرمانه d نیز محاسبه خواهدشد.

۳-۲- یک مثال ساده از رمزنگاری آر.اس.ای

فرض کنید یکی از حرف‌های کلمات پیامی که باید مخابره شود «ط» باشد، از آنجا که «ط» نوزدهمین حرف الفبای فارسی است پس پیام محرمانه‌ی اولیه m را عدد ۱۹ در نظر می‌گیریم. اکنون کافی است دو عدد اول دلخواه p و q را انتخاب کنیم مثلاً ۵ و ۱۱ که حاصل‌ضرب این دو عدد ۵۵ می‌شود که همان n است. حال از هر دو عدد یک واحد کم می‌کنیم که ۴ و ۱۰ بدست می‌آید که در این حالت حاصل‌ضرب آنها $\phi(n) =$ ۴۰ می‌شود. اکنون باید عدد دیگری را که نسبت به عدد ۴۰ اول باشد به عنوان کلید عمومی e در نظر بگیریم، مثلاً عدد ۳. حال سیستم ما برای به رمز درآوردن یک پیام آماده است، کافی است که عدد آن پیام را به توان ۳ برسانیم و سپس حاصل را بر ۵۵ تقسیم کنیم و باقیمانده را به عنوان پیام رمز شده در نظر بگیریم. اکنون برای به رمز در آوردن پیام ۱۹ که معادل حرف فارسی «ط» است، همین مراحل را اجرا می‌کنیم: $۱۹^{۳} \equiv_{۵۵}$ ۳۹. واضح است که ۳۹ پیام رمزشده‌ای است که می‌تواند با امنیت بیشتری به جای پیام اصلی یعنی ۱۹ مخابره شود چون در صورت شنود یا لو رفتن پیام برای شخص ثالث نامفهوم خواهد بود. اما پس از اینکه پیام رمزشده یعنی ۳۹ به مقصد رسید، چگونه گیرنده‌ی اصلی می‌تواند آن را به پیام اولیه یعنی ۱۹ تبدیل کند؟ برای این منظور، گیرنده علاوه بر در اختیار داشتن کلید عمومی یعنی اعداد ۵۵ و ۳، باید عدد محرمانه ی $\phi(n) =$ ۴۰ را نیز در اختیار داشته باشد تا به کمک آنها ابتدا معادله‌ی هم‌نهشتی $۳d \equiv_{۴۰} ۱$ را بنویسد. چون از حل این معادله کلید محرمانه ی $d =$ ۲۷ به دست می‌آید که فقط گیرنده‌ی اصلی در مقصد پیام باید به آن دسترسی داشته باشد. در این صورت گیرنده‌ی پیام باید از پیام رمزشده و نامفهوم یعنی ۳۹ ، به کمک $d =$ ۲۷، پیام محرمانه‌ی اولیه یعنی ۱۹ را طبق رابطه ی $۳۹^{۲۷} \equiv_{۵۵} ۱۹$ دوباره به دست آورد.

۳- حمله ی وینر

در این بخش ابتدا الگوریتم حمله ی وینر را می آوریم، سپس به بررسی درستی محاسبات ریاضی آن می پردازیم.

۳-۱- الگوریتم حمله ی وینر

در واقع *الگوریتم حمله وینر* که ورودی های آن e و n و خروجی های آن d و سپس p و q هستند، به این صورت است:

۱) مقادیر e و n را دریافت می‌کنیم،

۲) قرار می‌دهیم $a_{۰} = \left[\frac{e}{n}\right]$، سپس قرار می‌دهیم $\frac{k_{۰}}{u_{۰}} = a_{۰}$ که $(k_{۰}, u_{۰}) =$ ۱، اکنون قرار می‌دهیم $\phi(n) = \left[e\frac{u_{۰}}{k_{۰}}\right]$،

۳) اگر $\phi(n) \neq$ ۰ برو به مرحله‌ی (۸)، ولی اگر $\phi(n) =$ ۰ پس حدس ما در مورد $a_{۰}$ مناسب نبوده (و باید $a_{۱}$ را بیازماییم)،

۴) قرار می دهیم $v_{۰} = \frac{e}{n} - a_{۰}$ و $a_{۱} = \left[\frac{۱}{v_{۰}}\right]$، و به طور کلی از روی این جملات اولیه، رابطه ی بازگشتی

$$a_i = \left[\frac{۱}{v_{i-۱}}\right], \quad (۱)$$

را برای $i \in \mathbb{N}$ تعریف می نماییم که در آن $v_{i-۱} = \frac{۱}{v_{i-۲}} - a_{i-۱}$.

۵) قرار می دهیم

$$\begin{aligned} k_{۰} &= a_{۰} && , \quad u_{۰} = ۱ \\ k_{۱} &= a_{۱}k_{۰} + ۱ && , \quad u_{۱} = a_{۱} \\ &\vdots \\ k_i &= a_i k_{i-۱} + k_{i-۲} && , \quad u_i = a_i u_{i-۱} + u_{i-۲} \end{aligned} \quad (۲)$$

۶) اگر i فرد باشد، کسر متعارفی $\frac{k_i}{u_i}$ را تشکیل می دهیم (که معادل با کسر مسلسل $[a_0;a_1,...,a_i]$ خواهد بود)، و اگر i زوج باشد، کسر متعارفی معادل با کسر مسلسل $[a_0;a_1,...,a_i+1]$ را محاسبه و جایگزین $\frac{k_i}{u_i}$ می کنیم.

۷) اکنون دوباره $\phi(n)=\left[e\frac{u_i}{k_i}\right]$ را محاسبه می کنیم.

۸) اگر $\phi(n)\neq 0$، قرار می دهیم $g\equiv_{k_i} eu_i$ و $w=\frac{n-\phi(n)+1}{2}$ ، ولی اگر $\phi(n)=0$ پس حدس ما برای a_i مناسب نبوده و باید یک واحد به i افزوده و به رابطه (۱) بازگردیم و دوباره همه مراحل را تکرار نماییم.

۹) اگر $w\neq 0$ ، آنگاه قرار می دهیم $z=w^2-n$ ، ولی اگر $w=0$ پس حدس ما برای a_i مناسب نبوده و باید یک واحد به i افزوده و به رابطه (۱) بازگردیم و دوباره همه مراحل را تکرار نماییم.

۱۰) اگر z یک مربع کامل باشد، آنگاه قرار می دهیم $d=\frac{u_i}{g}$ ، به علاوه $p=w+\sqrt{z}$ و $q=w-\sqrt{z}$ و الگوریتم پایان می یابد، ولی اگر z یک مربع کامل نیست پس حدس ما برای a_i مناسب نبوده و باید یک واحد به i افزوده و به رابطه (۱) بازگردیم و دوباره همه مراحل را تکرار نماییم [۵].

۲-۳- بررسی درستی محاسبات ریاضی الگوریتم حمله ی وینر

فرض کنیم $n=pq$ پیمانهٔ دستگاه آر.اس.ای باشد که در آن p و q اعدادی اول هستند. این الگوریتم ساده را زمانی می توان به کار برد که $e<n$ ، $g.c.m$ یا بزرگترین مقسوم علیه مشترک $p-1$ و $q-1$ (که آن را با G نمایش دهیم) عدد کوچکی باشد، و p و q برای ذخیره در رایانه تقریبا دارای تعداد بیت های یکسانی باشند. اگرکوچکترین مضرب مشترک $p-1$ و $q-1$ را با H (یا $l.c.m$) نشان دهیم، رابطهٔ همنهشتی $ed\equiv_H 1$ برای توان رمزگذار (یا کلید عمومی) e و توان رمزخوان (یا کلید محرمانه) d ، همواره برقرار است. در نتیجه عدد صحیح K وجود دارد به طوری که

$$ed=KH+1 \qquad (۳)$$

در این صورت خواهیم داشت $ed=\frac{K}{G}(p-1)(q-1)+1$، و هرگاه شکل ساده شده کسر $\frac{K}{G}$ را به صورت $\frac{k}{g}$ نمایش دهیم که در آن $(k,g)=1$ ، آنگاه رابطه

$$ed=\frac{k}{g}(p-1)(q-1)+1 \quad ، \qquad (۴)$$

برای بعضی از اعداد صحیح k برقرار است. از تقسیم طرفین بر dpq خواهیم داشت $\frac{e}{pq}=\frac{k}{dg}\frac{(p-1)(q-1)}{pq}+\frac{1}{dpq}$ ،

و به عبارت دیگر

$$\frac{e}{pq}=\frac{k}{dg}(1-\delta) \quad ; \quad \delta=\frac{p+q-1-\frac{g}{k}}{pq}=\frac{1}{q}+\frac{1}{p}-\frac{1}{pq}-\frac{g}{kpq} \quad ، \qquad (۵)$$

و از آنجا که $\left(1+\frac{k}{g}\right)$ بسیار کوچکتر از $p+q$ است، می توان از آن صرفنظر کرد و حذفش به اعتبار تساوی اخیر لطمه‌ای نمی‌زند [۸]. بنابراین:

$$\delta\approx\frac{p+q}{pq}=\frac{1}{q}+\frac{1}{p}. \qquad (۶)$$

توجه شود که dg معادل با همان متغیر u در الگوریتم می باشد. ضمنا $\frac{e}{pq}$ شامل اطلاعات عمومی است و نیز تقریب نقصانی نزدیکی از $\frac{k}{dg}$ به شمار می رود، به عبارت دیگر $\frac{k}{dg}$ یک همگرای کسر مسلسل $\frac{e}{pq}$ است. از (۳) و (۴) نتیجه می شود که $(k,dg)=1$ ، بنابراین تا زمانی که δ به قدر کافی کوچک باشد، می توان از روش کسرهای مسلسل برای یافتن k و dg استفاده کرد. حال فرض کنیم $\frac{e}{pq}$ دارای بسط کسر مسلسل به فرم $[a_0;a_1,...,a_m]$ باشد. از طرف دیگر از کسر مسلسل مفروض $[a_0;a_1,...,a_m]$ ، می‌توان $\frac{e}{pq}$ را با در نظر گرفتن به صورت $\frac{k_m}{u_m}$ بازسازی نمود (به کمک روابط بازگشتی (۲) برای محاسبه k_i و u_i، که $0 \le i \le m$) [۹].

اکنون قرار می‌دهیم $[a_0;a_1,...,a_i]$ i-امین همگرای کسر مسلسل $[a_0;a_1,...,a_m]$ باشد. به سادگی ملاحظه می‌شود که اگر i فرد باشد، آنگاه $\frac{e}{pq}<[a_0;a_1,...,a_i]$، و اگر i زوج باشد نیز خواهیم داشت $[a_0;a_1,...,a_i]<\frac{e}{pq}<[a_0;a_1,...,a_i+1]$. زیرا از رابطه (۵) نتیجه می‌شود $\frac{k}{dg}>\frac{e}{pq}$، پس باید بتوان $\frac{k}{dg}$ را برابر با عدد گویای $\frac{k_i}{u_i}$ به دست آمده از روابط (۲) در نظر گرفت، که آن هم اگر i زوج باشد برابر است با: $[a_0;a_1,...,a_i+1]$ و اگر i فرد باشد برابر است با $[a_0;a_1,...,a_i]$. به بیان دقیق تر، عدد گویای $\frac{k_i}{u_i}$ می‌تواند برابر با کسر $\frac{k}{dg}$ شود، اگر شرط

$$kdg \le \frac{1}{\frac{3}{2}\delta}\ , \qquad (۷)$$

برقرار باشد که شرط اخیر از تلفیق دو رابطهٔ (۵) و (۴) بدست آمده است [۶]، به علاوه با توجه به رابطه (۶) خواهیم داشت:

$$kdg \le \frac{pq}{\frac{3}{2}(p+q)}\ . \qquad (۸)$$

اکنون نشان می‌دهیم که چگونه می‌توان درستی حدس‌های موجود برای k و dg را آزمود. به محض اینکه یک عدد گویای معین $\frac{r}{s}$ را حدس بزنیم، باید بررسی کنیم که آیا $\frac{r}{s}$ با $\frac{k}{dg}$ برابر است یا نه؟ برای سادگی این آزمایش G را آنقدر بزرگ فرض می‌کنیم که بتوان نوشت $ed > pq$. در نتیجه از رابطه (۴) خواهیم داشت $k > g$. سپس با ضرب طرفین رابطه (۴) در g، خواهیم داشت: $edg = k(p-1)(q-1)+g$.

بنابراین، با محاسبه تقسیم edg بر k ، خارج قسمت $\left[\frac{edg}{k}\right]=(p-1)(q-1)$ و باقیمانده g را بدست $edg \equiv_k g$ می‌آوریم که در آن نماد $[.]$ همان عملگر جزء صحیح می‌باشد. اگر $\left[\frac{edg}{k}\right]$ برابر صفر باشد، آنگاه حدس‌های اولیه ما برای k و dg نادرست بوده‌اند. در غیر این صورت مقدار بدست آمده برای $(p-1)(q-1)$ می تواند درست باشد، پس می‌توان از روی آن مقدار $\frac{p+q}{2}=\frac{pq-(p-1)(q-1)+1}{2}$ را محاسبه کرد. اگر این مقدار یک عدد صحیح نشد که باز هم حدس‌هایمان نادرست بوده‌اند و در غیر این صورت مقدار $\left(\frac{p-q}{2}\right)^2=\left(\frac{p+q}{2}\right)^2-pq$ را می‌توان محاسبه کرد.

اگر مقدار اخیر نیز یک مربع کامل باشد، این بار دیگر می توان اطمینان داشت که حدس‌مان برای k و dg حتماً صحیح بوده است. یادآوری می‌کنیم که g را به عنوان باقیمانده تقسیم edg بر k بدست آورده بودیم. بنابراین اکنون کلید محرمانه d را می‌توان از تقسیم dg بر g یافت و عوامل اول p و q را نیز به وسیلهٔ دو مقدار $\left(\frac{p+q}{۲}\right)$ و $\left(\frac{p-q}{۲}\right)$ می‌توان محاسبه نمود، که این معادل با موفقیت حمله ی وینر در شکستن رمز سیستم آر.اس.ای خواهد بود.

۴- بررسی شرایط لازم برای موفقیت الگوریتم حمله ی وینر

در اجرای حمله وینر نکات فرعی و حالات خاصی هم وجود دارند که به افزایش دقت یا کوتاه شدن زمان کمک می کنند، که در ادامه به آنها خواهیم پرداخت:

۱) برای کاهش حداکثری اندازه کلید محرمانه‌ای که می‌توان با حمله از راه کسرهای مسلسل در آر.اس.ای بدست آورد، با توجه به رابطه (۸)، دو شیوه وجود دارد: یا باید k را بزرگتر کرد و یا g را بزرگتر نمود.

برای بزرگتر کردن k، می‌توان توان عمومی e را بزرگترکرد (طبق رابطه (۴)). این کار را می توان با جمع کردن مضربی از H با e انجام داد. فرض کنیم $e > (pq)^{\frac{۳}{۲}}$. دراین صورت $\frac{k}{dg} > (pq)^{\frac{۱}{۲}}$ (طبق رابطه (۵)). جایگذاری $k = dg\,(pq)^{\frac{۱}{۲}}$ در رابطه (۸) به $d < ۱$ منجر می‌شود. اما به جواب رسیدن در این روش برای هر اندازهٔ کلید محرمانه قطعی نیست و به علاوه افزایش اندازه e ممکن است به طولانی شدن زمان رمزگشایی بیانجامد که البته در بعضی دستگاه‌ها قابل قبول است[۴].

اما برای بزرگتر کردن g، باید p و q را طوری انتخاب نمود که G به قدر کافی بزرگ گردد. اگرچه برای یافتن g یا عوامل g تحت شرایط خاصی راه‌های دیگری هم وجود دارند [۴]. البته برای ایجاد بهبودهای احتمالی در حمله به کلیدهای محرمانه، شیوه‌های دیگری نیز وجود دارد که در موارد بعدی به آنها اشاره می‌نماییم.

۲) می توانیم بگذاریم الگوریتم کسرهای مسلسل به جستجوی d به کندی و با حفظ شرط (۸) ادامه بدهد. این الگوریتم تنها تحت این شرط رسیدن به جواب را تضمین می‌کند، اما ممکن است این شرط سرعت را کند نماید و این ممکن است تا حدی بر اندازهٔ کلید محرمانه‌ای که بدست می‌آید نیز بیفزاید [۱۰].

۳) این بهبود در صورتی امکان پذیر است که مشاهده نماییم مخرج $\frac{e}{pq}$ (که تقریب نقصانی آن $\frac{k}{dg}$ است)، به سهولت دارای تقریب اضافی $(p-۱)(q-۱)$ باشد. در واقع تقریب اضافی دقیق‌تری به صورت $\left[\left(\sqrt{pq}-۱\right)^{۲}\right]$ نیز برای آن وجود دارد که در آن $[.]$ نماد جزء صحیح می باشد. با استفاده از این تقریب، رابطه (۸) به شکل زیر در می‌آید:

$$kdg < \frac{۲}{۳}\left(\frac{\sqrt{pq}-۱}{\sqrt{p}-\sqrt{q}}\right)^{۲} \quad ، \qquad (۹)$$

البته این رابطه اندازه کلید محرمانه‌ای که ممکن است پیدا شود را افزایش خواهد داد و با کاهش $|p-q|$، جواب بهبود بیشتری خواهد یافت [۱۱].

۴) این بار برای بهبود حمله با کسرهای مسلسل به آر.اس.ای، الگوریتم را بر اساس بعضی از حدس‌های $\frac{k}{dg}$ اجرا می نماییم. می‌توان ابتدا از حدس‌های مقدماتی شروع کرد و سپس حدس‌های موفقیت آمیزتری را آزمود. برای این کار، می توان یک جستجوی خطی برای $\frac{k}{dg}$ انجام داد. برای کلیدهای محرمانه تحت شرط (۸)، زمان مورد نیاز برای الگوریتم از مرتبه چندجمله‌ای است. با افزایش اندازه کلید محرمانه تحت این شرط، تعداد دفعاتی که الگوریتم باید انجام شود به طور نمایی افزایش می‌یابد [۱۲].

۵) در این راهکار می‌خواهیم با تلاش برای یافتن g یا عوامل g ، الگوریتم را بهبود دهیم. فرض کنید t عاملی از g باشد. در این صورت می‌توان از $t(\frac{e}{pq})$ به عنوان یک تقریب نقصانی $\frac{k}{d(\frac{g}{t})}$ استفاده کرد. بنابراین رابطه (۸) به شکل زیر در می‌آید:

$$kd(\frac{g}{t}) < \frac{pq}{\frac{3}{2}(p+q)} \quad ، \qquad (۱۰)$$

این رابطه اندازه d را افزایش می‌دهد که ممکن است به وسیله عاملی از t بدست آید. اکنون به راهی نیازمندیم که عوامل g را بیابد. چون g، عدد G را عادی‌کند پس g هر دوی p-۱ و q-۱ را نیز می‌شمارد، یعنی g عبارت pq-۱ را نیز عاد می‌کند، زیرا:

$$pq-1=(p-1)(q-1)+(p-1)+(q-1) \quad . \qquad (۱۱)$$

بنابراین شاید از تجزیه pq-۱ به عامل‌هایش بتوان عوامل g را نیز یافت. اگر g به قدر کافی بزرگ اختیار شود، و همه عوامل اول g بزرگ باشند، آنگاه ممکن است یافتن عوامل g بوسیلهٔ تجزیهٔ pq-۱ مشکل باشد. در صورتی که g آنقدر بزرگ باشد تا $\frac{(p-1)}{g}$ و $\frac{(q-1)}{g}$ به قدر کافی کوچک باشند، آنگاه یافتن g با جستجوی مقادیر احتمالی $\frac{(p-1)}{g}$ و $\frac{(q-1)}{g}$ نیز امکانپذیر است. با توجه به قضیهٔ لژاندر برای تقریب های دیوفانتی و رابطه (۵) خواهیم داشت:

$$\frac{(k+g)}{n}+\frac{k}{gp}+\frac{k}{gq}<\frac{1}{2d} \quad ، \qquad (۱۲)$$

پس $\frac{k}{dg}$ باید یک برآورد کسر مسلسل برای $\frac{e}{n}$ باشد. از آنجا که $\frac{e}{n}$ جزء داده های عمومی است و تقریب های کسر مسلسل آن نیز به سادگی توسط الگوریتم اقلیدس قابل محاسبه است، بنابراین یافتن کلید محرمانه d امکانپذیر می گردد [۶].

۶) در حالت کلی اگر راه فوق هیچ دستاورد خاصی نداشته باشد، آنگاه باید به یافتن کرانی برای d روی آورد. در این صورت در عین کوچک بودن g می‌توان فرض کرد $k<dg$، که البته زمان کار به مراتب بیشتر است. ضمنا از رابطه (۸) و فرض اخیر نتیجه می‌شود $kdg<d^2<\frac{pq}{\frac{3}{2}(p+q)}\approx\sqrt[2]{n}$، و از آن خواهیم داشت: $d<\sqrt[4]{n}$. ضمنا در حالت خاصی که $p\sim q\sim\sqrt{n}$ ، $g<d$ و $k\sim d$ داریم $e\sim\frac{n}{g}$ ، که عبارت اخیر از مرتبه $\frac{1}{d}$ خواهد بود، در صورتی که d حداکثر از مرتبه $\sqrt[4]{n}$ باشد. پس نهایتا حمله موفق خواهدشد اگر d دارای طول بیشینه حدود $\lambda=\frac{1}{4}=0/25$ برابر طول n باشد، زیرا از $d<n^{\lambda}$ و $\lambda<\frac{1}{4}$ نتیجه می شود $d<\sqrt[4]{n}$. به عبارت دیگر انتخاب d کوچکتر از $\sqrt[4]{n}$ موجب سهولت شکسته شدن رمز در دستگاه آر.اس.ای می‌گردد [۱۳].

با وجود چنین نتیجه‌ای، باید در برابر این وسوسه مقاومت نمود که انتخاب d کوچک برای سرعت بخشیدن به فرآیند رمزخوانی مناسب‌تر است، (مثلا زمانی که فرآیند رمزخوانی باید به یک پردازنده محدود مانند کارت هوشمند منتقل شود)، بنابراین d باید بزرگتر از ربع طول n انتخاب گردد. در نتیجه با توجه به رابطه (۳) از آنجا که $ed>n$، پس e باید بزرگتر از $\sqrt[4]{n^3}$ باشد، بنابراین می‌توان نتیجه گرفت که لازمه یافتن کلیدمحرمانه d برای موفقیت در حمله، برقراری دو شرط $d<\sqrt[4]{n}$ و $\sqrt[4]{n^3}<e<n$ می‌باشد. با تعمیم این روال می‌توان نشان‌داد که $d<n^{\frac{1}{4}}$، $n^{\frac{7}{4}}<e<n^2$ و در حالت کلی $d<\frac{\sqrt[4]{n}}{\sqrt[2]{3t}}$، $n^{t-\frac{1}{4}}<e<n^t$ که درآن t یک عدد صحیح کوچک است، نیز حمله موفق می‌شود، که یک ضعف برای دستگاه رمز آر.اس.ای محسوب می‌گردد.

۵- نتیجه‌گیری

در این مقاله پس از مرور کوتاهی بر تاریخچه ی رمزنگاری با کلید عمومی آر.اس.ای و حمله ی وینر به کلید محرمانه ی آن، به تشریح مکانیسم ریاضی این دستگاه رمزنگاری پرداختیم و سپس الگوریتم حمله ی وینر را بررسی و درستی مراحل این الگوریتم را از لحاظ محاسبات ریاضی نشان دادیم. سپس ایده هایی برای تسریع الگوریتم را ضمن بررسی شرایط لازم برای موفقیت حمله ی وینر ارائه کردیم. بنابراین رمزنگاران باید هنگام استفاده از سیستم آر.اس.ای، نسبت به رعایت محدودیت های لازم در انتخاب کلیدهای عمومی و محرمانه، دقت کافی مبذول نمایند.

مراجع

[۱] هاشمی‌پرست، س.م، خمسه م. مشهودی ش.، تعمیمی بر رمزنگاری طلایی به کمک حاصلضرب هادامارد ماتریس‌های k-فیبوناتچی، هفتمین کنفرانس بین المللی انجمن رمز ایران، ۱۳۸۹.

[2] Diffie, W., Hellman, M.E., Multiuser cryptographic techniques, AFIPS international computer conference proceedings, Vol. 45, pp. 109-112, 1976.

[3] Rivest, R.L., Shamir, A., Adleman, L., "A method for obtaining digital signatures and public key cryptosystems", Communications of the ACM, Vol. 21, No. 2, pp. 158-164, 1978.

[4] Nitaj, A., Ariffin, M.R.K., Adenan, N.N.H., Abu, N.A., Classical Attacks on a Variant of the RSA Cryptosystem, Progress in Cryptology; LATIN-CRYPT, 2021.

[5] Wiener, M.J., "Cryptanalysis of short RSA secret exponents", IEEE Transactions on Information Theory, vol. 36, pp. 553-558, 1990.

[6] Ruzai, W.N.A., Ariffin, M.R.K, Asbullah, M.A., Mahad, Z., Nawawi, A., On the Improvement Attack Upon Some Variants of RSA Cryptosystem via the Continued Fractions Method, 2020.

[7] Hinek, M.J., Lam, C.C.Y., Common modulus attacks on small private exponent RSA and some fast variants (in practice), Journal of Mathematical Cryptology, 4, pp. 57–93, 2010.

[8] Maitra, S., Sarkar, S., Revisiting Wiener's Attack, New Weak Keys in RSA, Lecture Notes in Computer Science, Springer, vol. 52, 22, pp. 228-243, 2008.

[9] Nemaneypour A., Number theory and related alghoritms in cryptography, M.Sc. thesis, JAIST, 2002.

[10] Verheul, E.R., Van Tilborg, H.C.A., Cryptanalysis of `less short' RSA Secret Exponents, Applicable Algebra in Engineering, Communication and Computing, 8 (5), pp. 425 – 435, 1997 .

[11] Nitaj, A., Another Generalization of Wiener's Attack on RSA, International Conference on Cryptology in Africa: Progress in Cryptology; AFRICA-CRYPT, pp. 174–190, 2008.

[12] Wu, M.E., Chen, C.M., Lin, Y.H., Sun, H.M., On the Improvement of Wiener Attack on RSA with Small Private Exponent, The Scientific World Journal, 2014.

[13] Bunder, M., Nitaj, A., Susilo, W., Tonien, J., A generalized attack on RSA type cryptosystems, Theoretical Computer Science, 704, pp. 74–81, 2017.