دانشکده مهندسی کامپیوتر

استاد درس: دکتر ابوالفضل دیانت

بهار ۱۴۰۳

تمرین ایتریم

امنیت سیستم های کامپیوتری

گزارش تمرین

ستاره باباجانی - ملیکا محمدی فخار

۹۹۵۲۲۰۸۶-۹۹۵۲۱۱۰۹

۱ سوال اول

در الگوریتم RSA ، کلید عمومی به صورت (e،n) تعریف می‌شود و پیام m با استفاده از این کلید عمومی به وسیلهٔ این فرمول رمزگذاری می‌شود:

$$C \equiv m^e \pmod{n} \quad 0 < m < n$$

در اینجا، اگر (m،n) برابر با ۱ باشد یعنی m و n نسبت به هم اول هستند. این ویژگی اهمیت زیادی دارد و به آن نیاز داریم. اگر m و n نسبت به هم اول نباشند، به این معناست که m و n دارای عوامل مشترکی هستند و ممکن است با تحلیل مسائل مرتبه‌ی کوچکتر مانند الگوریتم اقلیدس برای محاسبه gcd ، حمله کننده بتواند به سادگی پیام رمزگذاری‌شده را باز کند. هرچند که در ادامه و طبق اسلاید ۴۰ و ۴۱ فصل سوم، طی مقاله shamir و Rivest و Adleman نشان داده شده که حتی اگر m و n نسبت به هم اول نباشند نیز رابطه فوق برقرار است.

۲ سوال دوم

میدانیم در RSA پارامتر رمزگذاری e باید شرط زیر را ارضا کند و در آن برقرار باشد.

$$1 = \gcd(e, \phi(n)) = \gcd(e, (p-1)(q-1))$$

در RSA اعداد اول p،q متمایز هستند، بنابراین حداقل یکی از آنها فرد است. و در موارد بهتر هردو فرد هستند. بنابراین توان رمزگذاری e هرگز نمی تواند یک عدد صحیح زوج باشد. زیرا دیگر شرط فوق برقرار نمی ماند و کوچکترین مضرب مشترک دو عدد زوج میدانیم یک نیست و حداقل ۲ میباشد.

معمولاً برای انتخاب e، عددی اول بزرگ و بدون مشترک اول با تعداد اعداد اولیه کوچکتر استفاده می‌شود. عدد ۶۵۵۳۷ به عنوان یک انتخاب معمولی برای e در الگوریتم RSA به کار می‌رود، چرا که این عدد بزرگ و معتدل است و با اکثر اعداد اول کوچک مشترک اول ندارد.

۳ سوال سوم

الگوریتم RSA یکی از الگوریتم‌های رمزنگاری و امضای دیجیتال محبوب است که بر اساس مسائل محاسباتی مرتبه‌ی بزرگ مانند مسئله‌ی عددی اول برپایه‌ی عملیات ریاضیاتی استوار است. در RSA، دو عدد اول به نام اعداد فرما (Prime Numbers) به عنوان کلیدهای اصلی برای ایجاد کلیدهای عمومی و خصوصی استفاده می‌شوند.

اعداد فرما به دو عدد اول گفته می‌شوند که در الگوریتم RSA برای تولید کلیدها به کار می‌روند. این دو عدد عبارتند از:

پارامتر اول (p): یک عدد اول که به عنوان یکی از اعداد اصلی در تشکیل کلیدهای RSA استفاده می‌شود.

پارامتر دوم (q): یک عدد اول دیگر که نیز به عنوان یکی از اعداد اصلی در تشکیل کلیدها در الگوریتم RSA به کار می‌رود. سپس از این دو عدد، مقداری به نام ماژول (modulus) تولید می‌شود که با ضرب دو عدد اول به دست می‌آید:

$$n = p \times q$$

این مقدار n به عنوان پارامتر ماژول در کلیدهای RSA استفاده می‌شود. سپس از مقدار n، اعداد دیگری

برای تولید کلیدهای عمومی و خصوصی محاسبه می‌شوند.
تولید کلیدهای عمومی و خصوصی در RSA از مبانی نظری تئوری اعداد اول، مختصات و معادلات دیوفانتی استفاده می‌کند. اعداد فرما در اینجا نقش اساسی دارند و امنیت الگوریتم به زیاد بودن اندازه این اعداد فرما مرتبط است، زیرا با افزایش اندازه اعداد فرما، پیچیدگی فرآیند شکستن کلیدها افزایش می‌یابد.

۴ سوال چهارم

الگوریتم RSA از عملیات به توان رسانی برای تولید کلیدها و رمزگذاری/رمزگشایی پیام‌ها استفاده می‌کند. در اینجا چند نمونه الگوریتم بهینه برای این عملیات در محیط پیمانه ای ذکر می‌شود:

۱.۴ Square-and-Multiply

این الگوریتم برای به توان رساندن اعداد صحیح به توانهای دیگر استفاده می‌شود. با این الگوریتم می‌توان به سرعت توانهای بزرگتر را محاسبه کرد.
- این الگوریتم از تقسیم و حاصلضرب برای سریعتر به توان رساندن اعداد استفاده می‌کند.
- با تجزیه توان به صورت دودویی، هر بیت را از چپ به راست می‌خواند و با توجه به بیت، مراحلی از محاسبه را انجام می‌دهد.
- این الگوریتم به توانهای بزرگ به صورت کارآمد می‌پردازد.

کاربرد در RSA: در محاسبات RSA، عددی را به توان دلخواه (معمولاً تابع اقتدار عدد پایه) می‌رساند که در کلیدهای عمومی و خصوصی به کار می‌رود.

۲.۴ Montgomery Exponentiation

این الگوریتم نیز برای به توان رساندن سریع اعداد در محیط پیمانه ای استفاده می‌شود و به خصوص برای اعداد بزرگ مؤلفه فرد.
این الگوریتم از یک عمل تبدیل خاص برای اجتناب از تقسیم و استفاده از ضرب متوالی به منظور افزایش سرعت استفاده می‌کند.
کاربرد در RSA: در محاسبات RSA، معمولاً از اعداد بزرگ و مؤلفه فرد استفاده می‌شود، بنابراین الگوریتم Exponentiation Montgomery بهینه است.

۳.۴ Sliding Window Exponentiation

این الگوریتم نیز یک ترکیب از Square-and-Multiply با بهینه‌سازی‌های اضافی است که برای سرعت بخشیدن به محاسبات به توان رساندن اعداد مورد استفاده قرار می‌گیرد.
- توان دلخواه را به صورت باینری جدا می‌کند و بر اساس بیت‌های مجموعه شده، محاسبات را انجام می‌دهد.
- با استفاده از پنجره‌های متغیر، این الگوریتم می‌تواند به صورت موثرتری با توان‌های بزرگ کار کند.

کاربرد در RSA: در محاسبات RSA، این الگوریتم می‌تواند به سرعت توانهای بزرگتر را محاسبه کرده و در عملیات کلیدی مورد استفاده قرار گیرد.

۵ سوال پنجم

اثربخشی سیستم های رمزنگاری کلید عمومی به غیرقابل حل بودن (محاسباتی و نظری) برخی مسائل ریاضی مانند فاکتورسازی اعداد صحیح بستگی دارد. حل این مشکلات زمان بر است، اما معمولا سریعتر از امتحان کردن همه کلیدهای ممکن با brute force است. بنابراین، کلیدهای نامتقارن برای مقاومت برابر در برابر حمله باید طولانی تر از کلیدهای الگوریتم متقارن باشند. متداول ترین روش ها در برابر کامپیوترهای کوانتومی به اندازه کافی قدرتمند در آینده ضعیف فرض می شوند.
کلیدهای RSA ۱۰۲۴ بیتی از نظر قدرت معادل کلیدهای متقارن ۸۰ بیتی، و کلیدهای RSA ۲۰۴۸ بیتی با کلیدهای بلوکی ۱۱۲ بیتی معادل می باشند.

۶ سوال ششم

RSA یک الگوریتم رمزنگاری اسقاطی است که بر اساس مسائل اعداد اول بزرگ استوار است. فرآیند تولید اعداد اول در RSA به شکل زیر است:
۱) انتخاب دو عدد اول بزرگ (p ، q): و ابتدا دو عدد اول بزرگ و مختلف به صورت تصادفی انتخاب می‌شوند. این دو عدد باید بسیار بزرگ باشند تا فرآیند فاکتورگیری (تجزیه به عوامل اول) برای یک شخص ثالث زمان‌بر شود.
۲) محاسبه مقدار N: مقدار N برابر با حاصلضرب دو عدد اول p و q می‌شود: $N = p \times q$ این مقدار N برای ایجاد کلیدهای رمزنگاری و رمزگشایی در الگوریتم RSA استفاده می‌شود.
۳) محاسبه تابع فای آیلر (Euler's Totient Function):
تابع فای آیلر (φ) از رابطه زیر محاسبه می‌شود: φ(N)=(p-۱)(q-۱)
این تابع مهم است زیرا تاثیر مستقیم در انتخاب کلیدهای رمزنگاری دارد.
۴) انتخاب عددی برای کلید عمومی (e):
عددی که با تابع فای آیلر نسبتی اول باشد، به عنوان کلید عمومی انتخاب می‌شود. معمولاً اعدادی از خانواده اعداد اول مانند ۳۲۸۱، ۲۵۶۳ و ... به عنوان e استفاده می‌شود.
۵) محاسبه کلید خصوصی (d):
کلید خصوصی d به گونه‌ای انتخاب می‌شود که ضرب مودولو تابع فای آیلر به این شرط برسد:
(e*d) mod φ(N) = ۱ با انجام این عملیات، کلید خصوصی d به دست می‌آید.
در نهایت، کلید عمومی (N،e) و کلید خصوصی (N،d) به عنوان کلیدهای رمزنگاری و رمزگشایی در RSA استفاده می‌شوند. این کلیدها توسط افراد مختلف برای ارتباط امن از طریق این الگوریتم استفاده می‌شوند.
یکی از روش‌های معمول برای تولید اعداد اول بزرگ در RSA به کمک الگوریتم‌های تولید اعداد اول تصادفی است. یک الگوریتم معروف برای این کار الگوریتم اراتوستن (Eratosthenes) است. این

الگوریتم به صورت خلاصه به شرح زیر است:
۱) انتخاب یک محدوده:
ابتدا یک بازه از اعداد صحیح بزرگ انتخاب می‌شود. این بازه ممکن است بسیار وسیع باشد.
۲) استفاده از الگوریتم اراتوستن:
از الگوریتم اراتوستن برای تولید اعداد اول در این بازه استفاده می‌شود. این الگوریتم به این صورت عمل می‌کند که ابتدا یک لیست از اعداد از ۲ تا حداکثر عدد در بازه ایجاد می‌شود. سپس اعداد غیراول از لیست حذف می‌شوند.
۳) انتخاب تصادفی:
از میان اعداد اول بازه، دو عدد اول تصادفی p و q انتخاب می‌شوند. این دو عدد به عنوان اعداد اول p و q برای الگوریتم RSA استفاده می‌شوند.
۴) بررسی اندازه اعداد انتخابی:
اعداد انتخابی باید بسیار بزرگ باشند تا فرآیند فاکتورگیری توسط حمله‌های کلید عمومی مانند حمله فاکتورگیری Fermat-Kraitchik کارآیی نداشته باشد.