درس رمز ارز و بلاک چین - تمرین اول

محمود چوپانی - ۹۹۲۴۳۰۳۱

عباس یزدان مهر - ۹۹۲۴۳۰۷۷

.1

- مثلث و پنج ضلعی نیازمندی های امنیتی:
  - محرمانگی (confidentiality)
    - اطمینان از اینکه فقط افراد مجاز می توانند اطلاعات حساس را ببینند.
  - صحت (integrity)
    - اطمینان از اینکه داده ها هنگام ارسال، پردازش و ذخیره سازی ثابت و سالم می مانند.
  - دسترس پذیری (availability)
    - اطمینان از اینکه افراد مجاز می توانند اطلاعاتی که مربوط به آنهاست را ببینند.
  - احراز هویت (authentication)
    - اطمینان از هویت شخصی که دسترسی میگیرد و جلوگیری از دسترسی غیرمجاز.
  - مجوز داشتن (authorization) یا عدم انکار (non-repudiation)
    - اطمینان از منبع و جلوگیری از انکار بعد از دسترسی به اطلاعات
- همچنین از مورد زیر هم ممکن است نام برده شود:
  - حساب پذیری (accountability)
    - داشتن تاریخچه دسترسی ها به اطلاعات

.2

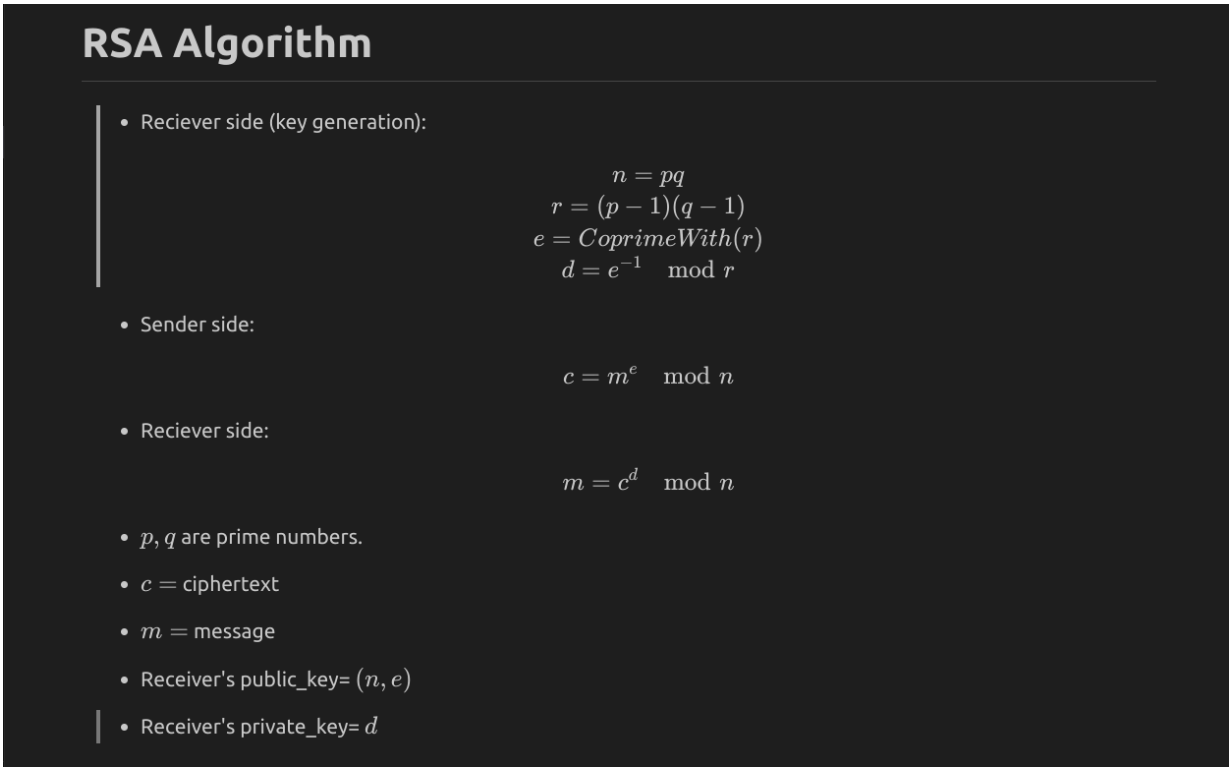
- در رمزنگاری برای تبدیل متن آشکار/رمزی به متن رمزی/آشکار باید از کلید استفاده کرد.
- اگر این کلید برای تبدیل متن آشکار به رمزی و برعکس یک کلید باشد به آن رمزنگاری متقارن می گویند.
- مشکلی که در رمزنگاری متقارن وجود دارد اشتراک گذاری کلید مشترک است.
- اگر کلیدی که برای رمز کردن و باز کردن رمز استفاده می شود متفاوت باشد به آن رمزنگاری نامتقارن می گویند.

- با استفاده از رمزنگاری نامتقارن یا کلید عمومی می توان کلید رمزنگاری متقارن را به اشتراک گذاشت و آن مشکل را حل کرد و پس از آن از رمزنگاری متقارن استفاده کرد.

.3

- در stream cipher در هر بار رمزنگاری یک بیت یا یک بایت رمز می شود ولی در block cipher در هر بار رمزنگاری یک بلوک (چند بایت همزمان) رمز می شوند.
- رمزنگاری stream فقط از confusion استفاده می کند. (کاهش ارتباط کلید با متن رمزی)
- رمزنگاری block هم از confusion و هم از diffusion استفاده می کند. (افزایش تغییرات روی متن رمزی بر اساس تغییرات کوچک روی متن آشکار)
- مدهای stream cipher: CFB, OFB
- مدهای block cipher: ECB, CBC

.4



- الگوریتم هایی مثل rsa بدلیل غیر ممکن بودن محاسبات زیاد در زمان کم کار می کنند و rsa بطور دقیق تر بر این اساس کار می کند که یک عدد بسیار بزرگ r داریم که از ضرب دو عدد بزرگ تشکیل شده است، پس اگر شخصی الگوریتمی پیدا کند که بتواند به راحتی اعداد بسیار بزرگ را

تجزیه کند در آن زمان او می تواند ابتدا مقدار r و سپس مقدار d یا کلید خصوصی را به راحتی بدست بیاورد و عملا این الگوریتم دیگر به هیچ عنوان امن نیست.

.5

بسیاری از الگوریتم های رمزنگاری و دیگر الگوریتم های استفاده شده در دنیای کنونی شامل دنیای رمز ارزها فقط با این فرض امن یا درست کار می کنند که بعضی محاسبات با هیچ کامپیوتری روی زمین قابل انجام نیستند و عملا امنیت محاسباتی (computational) دارند و نه امنیت کامل. حال اگر کامپیوترهای کوانتومی ای وجود داشته باشد دیگر فرض در نظر گرفته شده توسط این الگوریتم ها درست نیست و با این کامپیوترها می توان محاسبات را در زمان کم ممکن کرد و احتمالا امنیت این الگوریتم ها به خطر می افتد. برای همین امروزه بحث هایی مثل رمزنگاری های quantum-resistant مثل lattice-based cryptography مطرح است.

.6

- p - prime number and $2^{L-1} < p < 2^L$
- q - prime divisor of $p-1$ that is also prime.
- g(global component) - $g = h^{\frac{p-1}{q}} \mod p$ - $(1 < h < p-1)$
- $y = g^x \mod p$
- Sender's public_key: (p, q, g, y)
- Sender's private_key: x - random number - $(0 < x < q)$
- k - secret number - $(0 < k < q)$
- $r = (g^k \mod p) \mod q$
- $s = [k^{-1}(H(M) + x.r)] \mod q$
- Sender's Signature: (r, s)
- $v = [(g^{u_1} y^{u_2}) \mod p] \mod q$
- $w = (s')^{-1} \mod q$
- $u_1 = [H(M').w] \mod q$
- $u_2 = [r'.w] \mod q$
- verify: $v == r'$
- $'$ means reciever side.

- با کوچک بودن p , q طبق فرمول ها مقادیر g و y کوچکتر می شوند و به تبع آن حدس x از فرمول y = g^x mod p و با شرط x < q-1 خیلی راحت می شود. (مثلا با حمله brute force)
- از طرفی اندازه کلید ها کوچک تر می شود.
- به تبع آن تعداد کمتری کلید داریم و خطر collision بالا می رود.
- حملات با استفاده از فرمول های تجزیه ای ریاضی هم بسیار راحت تر می شود.

.7

- الگوریتم رمزنگاری خم های بیضوی یا elliptic-curve cryptography یا ecc ، جایگزینی برای الگوریتم های قدیمی تر کلید عمومی مثل rsa هستند.
- این الگوریتم ها از مسئله محاسباتی سخت ECDLP یا لگاریتم گسسته خم بیضوی استفاده می کنند.
- دلیل برتری آنها داشتن اندازه کلید کوچکتر و به تبع آن حافظه و ظرفیت انتقال کمتر همزمان با امنیت بیشتر می باشد. به عنوان مثال، یک کلید عمومی ecc - 256 bit امنیت قابل مقایسه با یک کلید عمومی RSA 3072 bit را دارد.

.8

- دریافت تراکنش‌ها: کلید عمومی برای دریافت تراکنش‌های رمزنگاری شده استفاده می‌شود. این کلید به عنوان آدرسی که می‌توان آن را با هر کسی که می‌خواهد رمزنگاری را به صاحب کلید عمومی ارسال کند به اشتراک گذاشت. کلید عمومی با یک کلید خصوصی همراه است که برای باز کردن و دسترسی به موجودیات دریافتی استفاده می‌شود. این مکانیزم اطمینان می‌دهد که در حالی که هر کسی می‌تواند تراکنش‌ها را به کلید عمومی ارسال کند، فقط صاحب کلید خصوصی متناظر می‌تواند به موجودیات دسترسی پیدا کند.

- تأیید مالکیت: کلید عمومی برای تایید مالکیت رمزنگاری شده استفاده می‌شود. وقتی یک تراکنش انجام می‌شود، کلید عمومی برای تایید اینکه تراکنش توسط صاحب موجودیات امضا شده است استفاده می‌شود. این فرآیند به نام امضای دیجیتال شناخته می‌شود و شامل ایجاد یک امضای دیجیتال با استفاده از کلید خصوصی است. امضا می‌تواند توسط هر کسی با استفاده از کلید عمومی تأیید شود که نشان می‌دهد تراکنش واقعاً توسط صاحب موجودیات مجاز بوده است.

- ایجاد آدرس‌های بلاک‌چین: کلید عمومی اغلب برای ایجاد آدرس‌های بلاک‌چین استفاده می‌شود. این آدرس‌ها معمولاً نسخه‌ای از کلید عمومی هستند که به عنوان شناسه منحصر به فرد برای هر کیف پول در شبکه عمل می‌کند. این امکان را برای شناسایی و تعامل آسان در اکوسیستم بلاک‌چین فراهم می‌کند.

- ارتباط امن: علاوه بر تراکنش‌ها، کلید عمومی همچنین می‌تواند برای ارتباطات امن در شبکه بلاک‌چین استفاده شود. آن‌ها امکان ارتباط رمزگذاری شده را فراهم می‌کنند که فقط صاحب کلید خصوصی متناظر می‌تواند پیام‌ها را رمزگشایی کند.

- به طور خلاصه، کلید عمومی در تکنولوژی بلاک‌چین برای دریافت تراکنش‌ها، تأیید مالکیت رمزنگاری شده، ایجاد آدرس‌های بلاک‌چین منحصر به فرد و تسهیل ارتباطات امن استفاده می‌شود. آن‌ها بخش اصلی چارچوب رمزنگاری کلید عمومی (PKC) هستند که پایه عملکرد ارزهای رمزنگاری شده و تکنولوژی بلاک‌چین را تشکیل می‌دهد.

.9

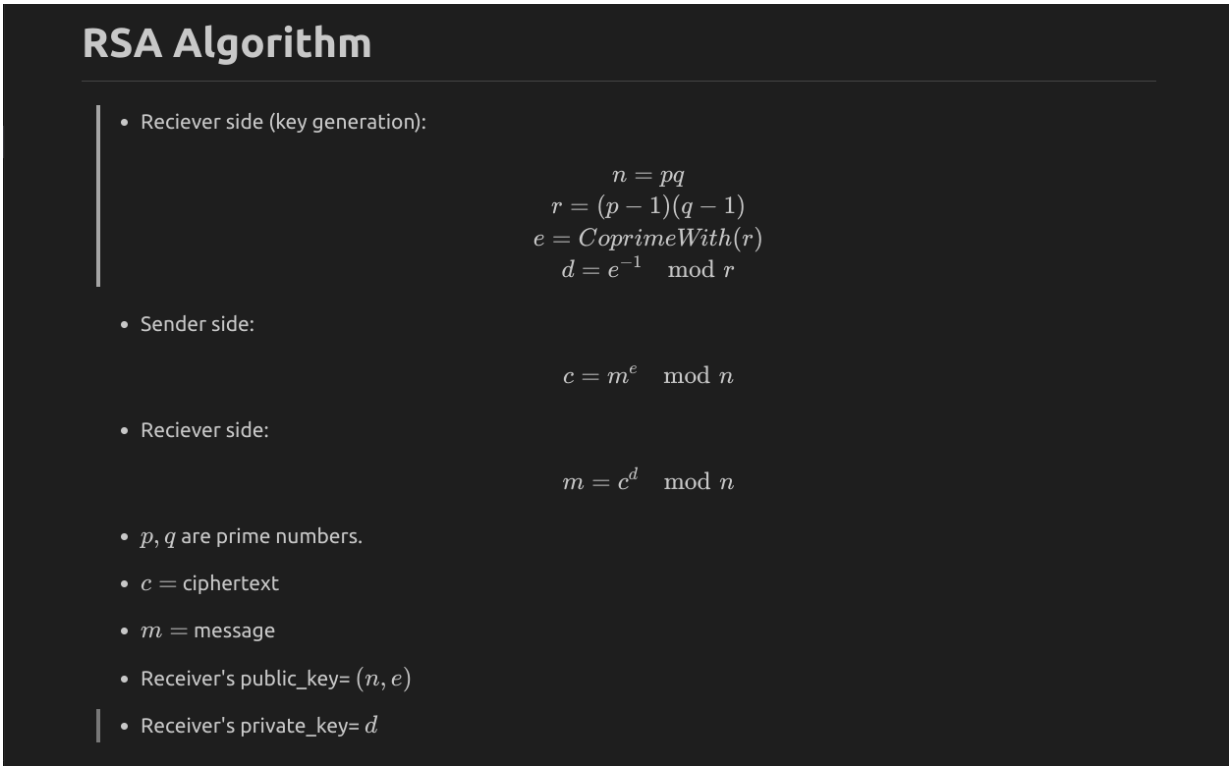
- مفهوم anonymity یا ناشناسی در زمینه تراکنش‌های مالی به معنای این است که هیچ کس نمی‌تواند تراکنش‌ها را به یک شخص خاص مرتبط کند. در بیت‌کوین، این مفهوم به دلیل طبیعت بلاک‌چین و رمزنگاری کلید عمومی و خصوصی امکان‌پذیر است. بلاک‌چین به طور مستقیم تراکنش‌ها را ثبت می‌کند ولی اطلاعات شخصی کاربران در آن ثبت نمی‌شود. این باعث می‌شود که تراکنش‌ها ناشناس باشند و فقط آدرس‌های بیت‌کوین ثبت شوند که متناظر با کلیدهای خصوصی کاربران هستند. این مکانیزم اطمینان می‌دهد که حتی در صورت تجزیه بلاک‌چین، هیچ اطلاعات شخصی در دسترس قرار نمی‌گیرد.

- با این حال، برای حفظ ناشناسی در بیت‌کوین، کاربران باید از نقاط ورودی متفاوتی استفاده کنند، مانند خرید بیت‌کوین در معاملات خصوصی، دریافت بیت‌کوین به عنوان پاداش برای خدمات ارائه شده یا استخراج. تراکنش‌های بیت‌کوین بعدی می‌توانند ناشناس باشند زیرا هیچ اطلاعات شخصی در روشنگر بلاک‌چین ثبت نمی‌شوند. اما حفظ ناشناسی از این نقطه به بعد هیچ گونه ضمانتی ندارد: حتی اگر کاربر موفق به ارائه اطلاعات شخصی باشد، هنوز هویت واقعی او می‌تواند در طول تراکنش‌های بیت‌کوین در شبکه کشف شود.

- بیت‌کوین ناشناس اما عمومی است: هویت‌ها در پروتکل بیت‌کوین ثبت نمی‌شوند، اما هر تراکنش انجام شده با بیت‌کوین در روشنگر بلاک‌چین قابل مشاهده است. این ناشناسی مورد توجه و چالش برای نظارت مالی است. با افزایش پذیرش ارز، ممکن است نیاز به یک مسابقه فناوری بین ناشناس‌سازان و شناسایی کنندگان ایجاد شود، که در یک طرف تکنیک‌های پیشرفته‌تری برای ردیابی حرکت وجوه در بلاک‌چین بین افراد و بین کشورها و در طرف دیگر تکنیک‌های بهبود یافته برای مخفی کردن هویت و فعالیت فردی ارائه می‌شود.

به طور خلاصه، بیت‌کوین امکان حفظ ناشناسی را فراهم می‌کند، اما این ناشناسی باید با دقت مدیریت شود و ممکن است در آینده با توسعه تکنولوژی و نظارت مالی، چالش‌های امنیتی اضافی برای حفظ ناشناسی وجود داشته باشد.

.10



- الگوریتم rsa بصورت کلی از محاسبات ریاضی برای محرمانگی و اصالت استفاده می کند: به این صورت که با محاسبات ریاضی کلیدهای عمومی و خصوصی تشکیل می دهد که رمز کردن یک پیام با کلید عمومی با محاسبات ساده ریاضی امکان پذیر است ولی همان پیام رمزی را دیگر نمی توان با محاسبات ساده ریاضی و کلید عمومی بدست آورد و عملا عملیات رمز کردن بدون

داشتن کلید خصوصی یک عملیات یک طرفه یا به نوعی one-way function است و تنها راه برای بازگشت به پیام اصلی از پیام رمز شده داشتن کلید خصوصی است. (بصورت کلی trapdoor function است.)

- بنابرین محرمانگی برقرار است: چون هیچ کس به جز گیرنده کلید خصوصی را ندارد.
- از طرفی با استفاده از rsa با hash کردن پیام (h = (hash(message) و همین توضیحات گفته شده و بصورت عقبگرد (h' = s^e mod n = (h^d)^e mod n = h) می توان پیام را امضا کرد (امضای دیجیتال) و با این امضا می توان از اصالت پیام اطمینان پیدا کرد.

.11

- تولید کلید:
  - عملیات پیدا کردن باقی مانده یا مد گرفتن
  - پیدا کردن معکوس ضربی پیمانه ای یا modular multiplicative inverse که معمولا توسط الگوریتم Extended Euclidean algorithm یا با استفاده از نظریه اویلر حساب می شود.
  - پیدا کردن phi(n) که تابع totient اویلر نامیده میشود و تعداد اعداد کوچکتری که نسبت به n اول هستند را پیدا می کند.

.12

- در رمزنگاری deterministic یا قطعی الگوریتم رمز کردن یا encryption به ازای یک کلید و یک متن ثابت همیشه یک خروجی یا متن رمزی می دهد، مثل rsa.
- در رمزنگاری probabilistic یا احتمالی تابع encryption به ازای یک کلید و یک متن ثابت خروجی یا متن های رمزی متفاوتی می دهد. مثل Elgamal و Paillier.
- یک راه ساده برای تبدیل یک الگوریتم deterministic به probabilistic استفاده از یک رشته رندوم در ابتدای متن آشکار است. یک مثال از تکنیک پیشرفته تر این رویکرد OAEP است که دوباره یک trapdoor function است.