به نام خدا

تمرین چهارم امنیت داده و شبکه

نیم‌سال اول ۱۴۰۴-۱۴۰۳

دانشکده مهندسی کامپیوتر

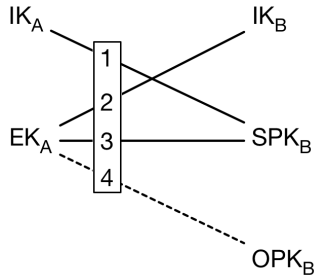
دانشگاه صنعتی شریف

موضوع **امنیت شبکه**

موعد تحویل **ساعت ۲۳:۵۹ چهارشنبه ۱۲ دی ۱۴۰۳**

طراحی تمرین توسط **حسین نقدبیشی - رضا سعیدی**

۱. پروتکل سیگنال از دو بخش مهم X3DH و Double Ratchet تشکیل شده است. درمورد آن‌ها تحقیق کرده و سپس به سوالات زیر پاسخ دهید.



شکل ۱: تبادل کلید X3DH

آزاده (A) و بهرام (B) هر دو از کاربران پیام‌رسان سیگنال هستند و در آن از قبل حساب کاربری دارند. آزاده می‌خواهد برای اولین بار به بهرام در پیام‌رسان سیگنال پیام بدهد. با توجه به مستندات موجود برای فرآیند ساخت کلید در سایت signal.org موارد زیر را با فرض معتمد بودن کارگزار [۱] توضیح دهید.

آ. به صورت مختصر و با ترتیبی درست، کلیهٔ محاسبات آزاده و بهرام، و تبادلات بین آزاده و بهرام تا رسیدن به کلید مشترک را توضیح دهید.

ب. کدام یک از اجزاء می‌توانند در بازه‌های زمانی مختلف تعویض شوند؟ کدام یک از اجزاء باید همواره ثابت بمانند، چرا؟

ج. کدام اجزاء تأیید اصالت دوجانبه [۲] را تضمین می‌کنند؟ اگر کارگزار معتمد نباشد، چه راهکاری برای اطمینان از عدم حملهٔ شخص میانی [۳] ارائه می‌دهید؟

د. کدام اجزاء پروتکل ویژگی رازمانی پیش‌سو [۴] و پس‌سو [۵] را تضمین می‌کنند؟ چگونگی آن را توضیح دهید.

ه. اگر چند پیام به مقصد خود نرسند، چه اتفاقی رخ می‌دهد؟

و. نقش SPK_B و اهمیت وجود آن در پروتکل را توضیح دهید. اگر کارگزار نتواند به‌اندازهٔ کافی OPK_B تأمین کند چه تغییری در امنیت مکالمهٔ دو شخص به وجود می‌آید؟

ز. آیا اگر تلفن همراه بهرام اکنون به اینترنت متصل نباشد، آیا لازم است آزاده منتظر آنلاین شدن بهرام برای تبادل کلید بماند؟

ح. یکی از ویژگی‌های پروتکل سیگنال انکارپذیری [۶] است. توضیح دهید که چگونه انکارپذیری تضادی با اطمینان از اصالت پیام‌ها ندارد. سپس نشان دهید که اولاً در مکالمهٔ بهرام با آزاده هر دو می‌توانند هر پیام دلخواه را انکار کنند [۷]، و دوم، سودابه بدون اینکه حتی یک پیام با آزاده رد و بدل کرده باشد، می‌تواند یک مجموعه مکالمهٔ جعلی با وی را بسازد. [۸] کاربرد انکارپذیری در دنیای واقعی چیست؟

۲. در پروتکل TLS 1.2 و نسخه‌های قبلی، کارگزار و کارخواه [۹] می‌توانستند برای استفاده از فشرده‌سازی توافق کنند. در صورت این توافق، از یک الگوریتم فشرده‌سازی بی‌اتلاف [۱۰] برای فشرده‌سازی قبل از انجام رمزنگاری استفاده می‌شد. پس از ارائهٔ حملاتی مانند CRIME این قابلیت در TLS 1.3 تعبیه نشد.

[1] Server
[2] Mutual Authentication
[3] Person in the middle
[4] Forward Secrecy
[5] Backward Secrecy
[6] Repudiation
[7] Message Repudiation
[8] Participation Repudiation
[9] Client
[10] Lossless

یک الگوریتم فشرده‌سازی مانند $C(m)$ پیامی مانند m را می‌گیرد و با استفاده از کاهش تکرر[11] سعی می‌کند آن را کوتاه‌تر کند. یک الگوریتم فشرده‌سازی را بی‌اتلاف می‌گوییم اگر همواره معکوس‌پذیر باشد، به گونه‌ای که به ازای هر خروجی‌اش، بتوان ورودی یکتای m را پیدا کرد که منجر به ایجاد آن خروجی شده است. هم الگوریتم فشرده‌سازی و هم معکوسش باید از نظر محاسباتی بهینه باشند. یک الگوریتم فشرده‌سازی می‌تواند طول برخی ورودی‌ها را کمتر، و طول برخی ورودی‌ها را بیشتر کند، ولی الگوریتم فشرده‌سازی‌ای خوب است که پیام‌های مرسوم را عموماً کوتاه کند.

الگوریتم فشرده‌ساز بی‌اتلاف $C(m)$ را در نظر بگیرید که پیام‌های معتبر برای آن مجموعه رشته‌هایی هستند که بین یک تا پانزده 1 متوالی دارند و همیشه به تنها یک 0 ختم می‌شوند (به‌جز برای رشته 1های آخری که تک صفر انتهایی را ندارند). الگوریتم فشرده‌ساز $C(m)$ یک دنباله از مقادیر ۴ بیتی است که تعداد 1های هر رشته را نشان می‌دهد. برای مثال حالت زیر را ببینید که در آن طول خروجی $|C(m)| = 16$ کوتاه‌تر از طول پیام $|m| = 22$ است:

$$m = 1111111111011111010111$$
$$C(m) = 1010010100010011$$

اگر پیام m بیش از پانزده 1 متوالی داشته باشد (که برای پیام‌های معتبر مجاز نیست)، $C(m)$ پانزده 1 اول را به صورت 1111 نشان می‌دهد و بقیه‌ی رشته را انگار که آن پانزده 1 وجود نداشته باشند، پردازش می‌کند. برای مثال پیام $m = 111111111111111111110111$ به صورت $C(m) = 111101010011$ فشرده‌سازی می‌شود. در این حالت تابع فشرده‌ساز دیگر معکوس‌پذیر نیست زیرا بین پیام‌هایی با پانزده یکِ متوالی و منتهی به یک صفر، و پیام‌هایی با تعداد بیشتری یک تمایزی قائل نمی‌شود.

آ. فرض کنید که یک پیام ابتدا توسط تابع فشرده‌سازی بالا فشرده شده، و پس از پد شدن با یک شیوهٔ رمزنگاری بلوکی با اندازه بلوک ۸ بیت رمز می‌شود. توضیح دهید که یک مهاجم تنها با استفاده از مشاهدهٔ غیرفعالانهٔ تعداد بلوک‌های ارسال شده اولاً چه اطلاعاتی درمورد پیام m می‌تواند به دست بیاورد و دوماً چگونه با در نظر گرفتن دو پیام m با طول‌های یکسان و شرایطی خاص می‌تواند تشخیص دهد که کدام یک رمز شده است.

ب. شرایط را مشابه بخش قبل در نظر بگیرید با این تفاوت که حالا مهاجم توانایی افزودن بیت‌های دلخواه خود به ابتدای پیام m (که از آن هیچ اطلاعی ندارد) و رمز کردن آن به تعداد دلخواه به دست آورده است. یعنی مهاجم می‌تواند به ابتدای یک پیام ثابت نامعلوم، مانند m، چندین بار بیت‌های دلخواه خود را اضافه کرده و سپس قربانی مشابه شرایط بالا پیام تغییریافته را ابتدا فشرده، و سپس رمز و ارسال کند (مثلاً به علت نصب یک ویروس با دسترسی محدود روی دستگاه قربانی). راهی را پیدا کنید که در آن مهاجم بتواند تعداد 1های رشتهٔ اول m را بیابد.

ج. به صورت مختصر توضیح دهید که در حملهٔ CRIME مهاجم در چه‌شرایطی و چگونه می‌تواند کوکی‌های یک کاربر را با موفقیت بیابد.

[11] Redundancy

۳. برای حل این سؤال نیازمند یک سیستم‌عامل با هستهٔ لینوکس و مجهز به بسته‌های nftables و netcat هستید. پاسخ هر مرحلهٔ حاوی قواعد، باید در قالب یک فایل جداگانهٔ nft. باشد. از هر مرحله یک یا چند نماگرفت[۱۲] که نشان‌دهندهٔ اجرای صحیح قواعد دیوارهٔ آتش باشند تهیه کنید و همراه فایل قواعد با پوشه‌بندی مناسب در یک فایل زیپ قرار دهید.

آ. یکی از مزیت‌های nftables نسبت به iptables، atomic بودن آن است. اهمیت این ویژگی را با ذکر یک مثال بیان کنید.

ب. برنامهٔ netcat را با استفاده از nc -l -k 127.0.0.1 8001 اجرا کنید تا بر روی درگاه ۸۰۰۱ بسته‌های TCP دریافت شوند. قواعدی بنویسید که اعمال آن‌ها باعث شود که یک کارخواه بتواند با استفاده از اجرای nc 127.0.0.1 8000 یک مکالمه دو طرفه با کارساز داشته باشد (هم پیام‌های کارخواه به کارساز برسد و هم برعکس).

> **راهنمایی**
>
> یکی از راهکارها این است که از Payload Statement استفاده کنید.

ج. این بار برای بسته‌های UDP برنامهٔ netcat را همزمان با nc -u -l -k 8002 و nc -u -l -k 8001 اجرا کنید. دیوارهٔ آتش باید هر بسته‌ای با مقصد درگاه‌های ۸۰۰۰ تا ۹۰۰۰ UDP را با توزیعی همواره ثابت، به یکی از این دو درگاه ارسال کند. نشان دادن ارتباط یک طرفه (فقط ارسال موفق از سمت کارخواه به کارساز) کافی است. در نماگرفت خود نشان دهید که برخی از بسته‌ها با موفقیت به درگاه ۸۰۰۱ و برخی به درگاه ۸۰۰۲ می‌رسند.

> **راهنمایی**
>
> از قابلیت Jenkins Hash در nftables استفاده کنید.

د. ابتدا تفاوت بین دیواره‌های آتش حالت‌دار[۱۳] و بی‌حالت[۱۴] را مختصراً توضیح دهید و سپس یک مجموعه قواعد برای یک کارگزار را بنویسید که فقط خدمات وب ارائه می‌دهد و مدیریت آن نیز با SSH رخ می‌دهد. تمامی بسته‌هایی که به درگاه ۴۴۳ و ۸۰ سرور می‌رسند، مگر آنکه از نظر conntrack لینوکس نامعتبر باشند، باید پذیرفته شوند. درگاه ۲۲ مربوط به SSH نیز تنها زمانی باز می‌شود که قبل از آن Port Knocking به درگاه‌های ۱۳۳۷ و ۱۳۳۸ رخ داده باشد.

[12] Screenshot
[13] Stateful
[14] Stateless

۴. چهار فایل حاوی ترافیک ضبط شده برای هر نشست کاربر به پیوست این تمرین ارائه شده است. با توجه با این فایل‌ها به سوالات زیر پاسخ دهید. درپاسخ هر بخش تصویر اجرای تمامی مراحل آورده شود.

> **راهنمایی**
>
> برای مشاهده محتوای این فایل‌ها میتوانید از ابزار Wireshark استفاده کنید.

I. ترافیک HTTP

آ. نام سه دامنه و آدرس آی‌پی آن‌ها را که توسط آدرس 192.168.0.100 مشاهده شده را بدست آورید.

ب. سه کوئری جستجو به همراه دامنهٔ سایت هر کوئری، کاربر 192.168.0.100 را ارائه دهید.

II. ترافیک FTP

آ. نام کاربری و کلمه عبور کاربر برای اتصال به سرور FTP چیست؟

ب. تمام فایل‌هایی که کاربر از این سرور دانلود کرده است را لیست کنید.

ج. مسیر کامل دو فایل در سرور FTP (در پوشه‌های مختلف) که دانلود نشده‌اند را لیست کنید.

III. ترافیک POP

آ. نام کاربری و کلمه عبور POP چیست؟

ب. چند ایمیل در صندوق پستی کاربر وجود دارد؟

ج. محتویات از، به، موضوع و تاریخ را برای یک ایمیل دلخواه بنویسید.

د. این شخص از چه برنامهٔ ایمیل و سیستم‌عاملی برای ارسال و دریافت ایمیل استفاده می‌کند؟