گزارش پروژه ی دوم شبکه های کامپیوتری

زهرا سالاریان- 9731089

بخش سوالات تشریحی:

1. تلنت چیست و کاربرد آن را شرح دهید.

نلنت یک پروتکل شبکه است که برای دسترسی مجازی به رایانه و ایجاد کانال ارتباطی دو طرفه، مشارکتی و متنی بین دو ماشین مورد استفاده قرار می گیرد. این دستور برای ایجاد جلسات از راه دور از پروتکل کنترل انتقال/ پروتکل اینترنت (TCP / IP) پیروی می کند. در وب، پروتکل انتقال متن (HTTP) و پروتکل انتقال پرونده (FTP) به سادگی کاربران را قادر می سازد تا پرونده های خاصی را از رایانه ها از راه دور درخواست کنند و برخلاف بیشتر پروتکل‌های موجود مثل HTTP، علاوه بر نقل‌وانتقال، امکان دسترسی به فایل‌های مختلف در Host مقصد را نیز فراهم می سازد تا کلاینت بتواند فایل‌ها را ادیت و برنامه‌های گوناگون را در Host به طور مستقیم اجرا و کنترل کند. پورت پیش فرض Telnet مقدار 23 می باشد.

2. امنیت تلنت چگونه است؟

Telnet یک پروتکل امن نیست و رمزگذاری نشده است. با نظارت بر اتصال کاربر، هرکسی می تواند به نام کاربری، رمز ورود و سایر اطلاعات شخصی وی که در Telnet در متن ساده تایپ می شود، دسترسی پیدا کند. با استفاده از این اطلاعات می توان به دستگاه کاربر دسترسی پیدا کرد. علت این امر این است که در زمانی که این پروتکل طراحی شد بیشتر کاربرهای شبکه، افراد نظامی، موسسات آکادمیک، تسهیلات تحقیق دولتی و خصوصی و به طور کلی افراد مطمئنی بودند و مسئله‌ی امنیت به اندازه ی امروزه مطرح نبود. اما امروز برای تامین امنیت از پروتکل SSL استفاده می شود.

3. Transport Layer Security یا TLS ، یک پروتکل امنیتی است که به طور گسترده ای تصویب شده و برای تسهیل حریم خصوصی و امنیت داده ها برای ارتباطات از طریق اینترنت طراحی شده است. یک مورد اصلی استفاده از TLS رمزگذاری ارتباطات بین برنامه های وب و سرورها است، مانند مرورگرهای وب که یک وب سایت را بارگیری می کنند. ازTLS همچنین می تواند برای رمزگذاری ارتباطات دیگر مانند ایمیل، پیام و صوت استفاده شود.

سه مولفه اصلی برای دستیابی به پروتکل TLS وجود دارد: رمزگذاری، احراز هویت و یکپارچگی.
رمزگذاری: داده های منتقل شده از اشخاص ثالث را مخفی می کند.

احراز هویت: اطمینان حاصل می کند طرفین تبادل اطلاعات کسانی هستند که ادعا می کنند هستند.
یکپارچگی: تأیید می کند که داده ها جعل یا دستکاری نشده اند.

در طول Handshaking مربوط به TLS دستگاه کاربر و وب سرور:

مشخص می شود کدام نسخه ی TLS استفاده خواهد شد.
تصمیم گرفته می شود که از کدام مجموعه های رمزنگاری استفاده شود.
هویت سرور با استفاده از گواهینامه TLS سرور تأیید می شود.
پس از اتمام Handshaking، کلیدهای مربوطه را برای رمزگذاری پیام ها بین آنها ایجاد می کند.

مراحل اتصال پروتکل TLS:

Step 1: Client Hello (Client → Server):

با ارسال پیام "Hello" از سمت کلاینت به سرور، handshake آغاز می شود. پیام شامل نسخه TLS، مجموعه رمزهای پشتیبانی شده و یک رشته بایتی تصادفی است.

Step 2: Server Hello (Server → Client):

سرور در پاسخ به پیام "Hello" کلاینت، پیامی که شامل گواهی SSL سرور، مجموعه رمزگذاری شده انتخاب شده سرور و یک رشته تصادفی دیگر که توسط خود سرور تولید شده است را می فرستد.

Server's digital signature:

سرور از کلید خصوصی خود برای رمزگذاری تصادفی سرویس گیرنده استفاده می کند. این داده های رمزنگاری شده به عنوان امضای دیجیتالی سرور عمل می کند و نشان می دهد که سرور دارای کلید خصوصی است که با کلید عمومی گواهی SSL مطابقت دارد.

Digital signature confirmed:

کلاینت امضای دیجیتال سرور را با کلید عمومی رمزگشایی می کند و تأیید می کند که سرور کلید خصوصی را کنترل می کند و شخصی است که خودش می گوید.

Client and server calculate the premaster secret:

به جای این که کلاینت premaster secret را تولید کند و آن را به سرور ارسال کند ، مانند یک RSA handshaking، کلاینتو سرور از پارامترهای DH که رد و بدل کرده اند برای محاسبه یک راز premaster secret جداگانه استفاده می کنند.

Session keys created:

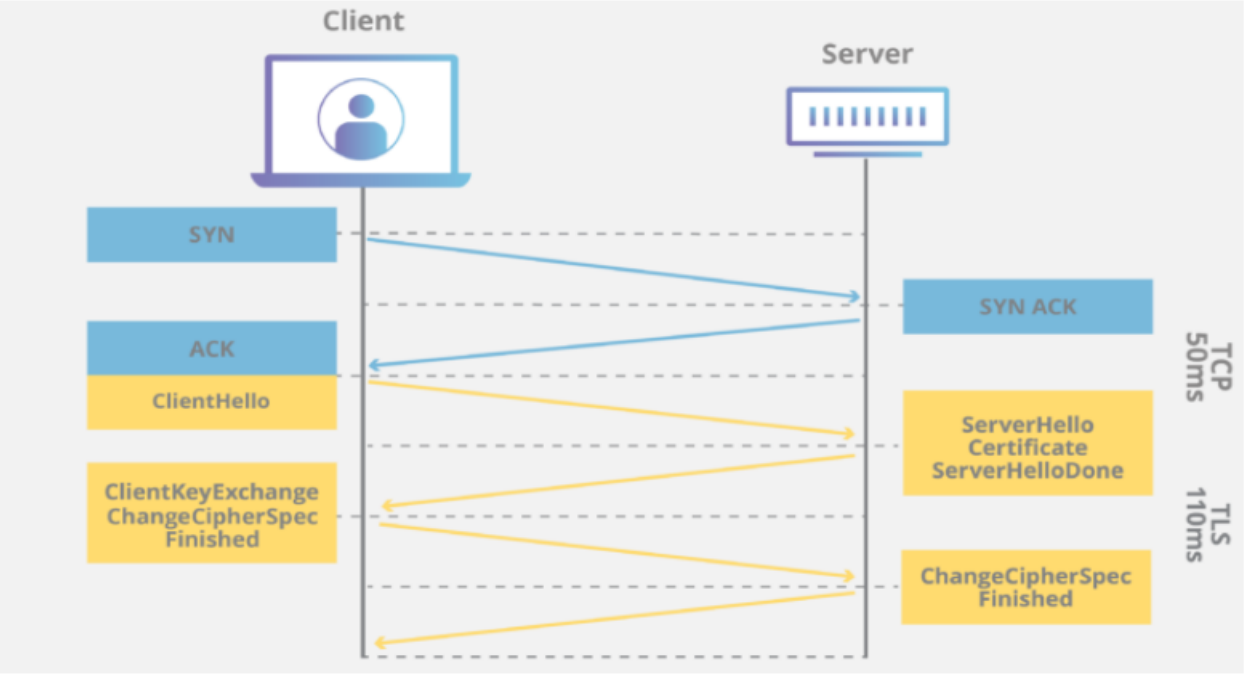
اکنون، سرویس گیرنده و سرور، کلیدهای جلسه را از طریق premaster secret ، کلاینت تصادفی و سرور تصادفی محاسبه می کنند.

Client is ready:

کلاینت یک پیام پایان که با کلید سشن رمزنگاری شده است را میفرستد.

Server is ready:

سرور یک پیام پایان که با کلید سشن رمزنگاری شده است را میفرستد.
و به این ترتیب ارتباط امن حاصل شده است.

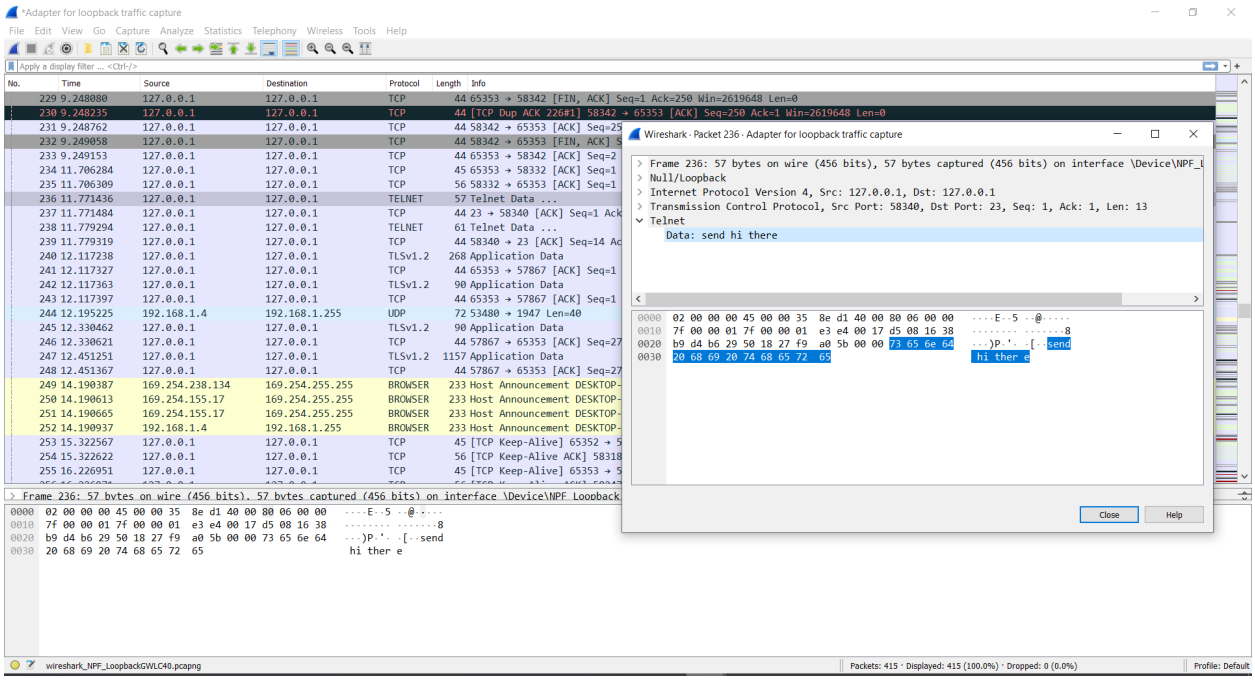


بررسی بخش وایرشارک و تفاوت send و send -e:

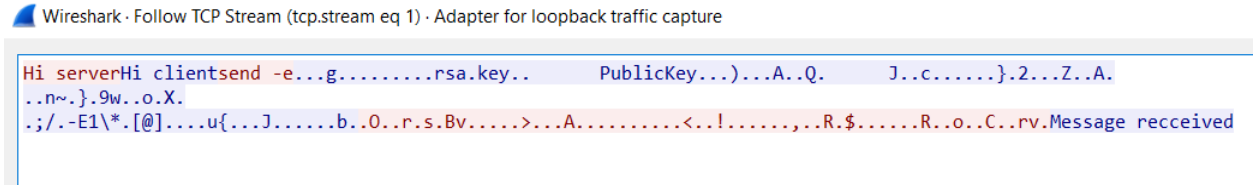
در این بخش از کد گذاری نامتقارن rsa (کلید عمومی و خصوصی) استفاده شده است.

اگر بر روی یکی از پکت‌های کانکشنمان دو بار کلیک کنیم تمام پکت‌های آن را نمایش می‌دهد و میتوانیم در قسمت Data داده ای که فرستاده شده است را ببینیم.

در شکل زیر پیامی که کد گذاری نشده است را از طریق telnet send hi there فرستاده ایم و همانطور که در شکل زیر مشاهده می شود متن این پیام کاملا قابل استخراج است:



حال اگر این بار دستور telnet send -e hi there را اجرا کنیم ابتدا کلاینت دستور send -e را استخراج می‌کند سپس برای سرور کلمه‌ی نیز send -e را می‌فرستد که به سرور می فهماند که کلاینت می‌خواهد یک پیام رمزنگاری شده ارسال کند. سپس در سمت سرور یک کلید عمومی ساخته شده و به کلاینت فرستاده می شود. کلاینت پیام را با کلید عمومی رمزنگاری کرده و می فرستد و در سمت سرور با کلید خصوصی آن را encode می کنیم. در شکل زیر می توانیم محتوای یک پیام رمزنگاری شده را مشاهده کنیم:



در شکل بالا handshaking اولیه، درخواست send -e توسط کلاینت به رنگ قرمز، PublicKey ارسال شده توسط سرور به کلاینت به رنگ آبی و در نهایت پیام ارسال شده به صورت رمزنگاری شده را داریم که توسط سرور به رنگ قرمز ارسال شده و در نهایت وقتی سرور آن را دریافت کرد پیام Message received را می فرستد.

عکس ها همچنین به طور مجزا ارسال شده اند.