دانشگاه صنعتي اميركبير

(پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر و فناوری اطلاعات

عنوان

آشنائی با پروتکل HTTPS

نگارش

سید محمد مهدی موسوی

استاد راهنما

دکتر رضا صفابخش

خرداد ۱۳۹۵

دانشگاه صنعتي اميرکبير

(پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر و فناوری اطلاعات

عنوان

آشنائی با پروتکل HTTPS

نگارش

سید محمد مهدی موسوی

استاد راهنما

دکتر رضا صفابخش

خرداد ۱۳۹۵

چکیده

بررسی راهکارهای امن کردن پروتکل HTTP هدف این مقاله است. در این اثر نحوه ی امن کردن پروتکل از طریق الگوریتم های رمز گذاری بررسی، و مزایا و معایب هر کدام از الگوریتم ها بررسی شده است.

فهرست مطالب

۱ مقدمه

امروزه امنیت شبکه یکی از مسائل مهم برای ادارات و شرکت های دولتی و سازمان های کوچک و بزرگ است. فضای مجازی به دلیل گسترگی و عدم کنترل پذیری رویکردی سیستماتیک را برای امنیت شبکه می‌طلبد. در بسیاری از صنایع، امنیت به شکل پیشرفته یک انتخاب نیست، بلکه یک ضرورت است. محافظت از اطلاعات اختصاصی، به منابع مالی نامحدود و عجیب نیاز ندارد بلکه با درکی از مسأله، خلق یک طرح امنیتی استراتژیک و تاکتیکی می‌تواند تمرینی آسان باشد در این اثر به بررسی راه کار های امنیت بخشید به پروتکل HTTP از طریق دو پروتکل لایه ی امنیت می‌پردازیم.

۲ مروری بر پروتکل HTTP[۱]

پروتکل HTTP یک از پروتکل های لایه ی کاربرد[۲] در شبکه‌های کامپیوتری است که توسط آقای تیم برنرس لی[۳] در نوامبر سال ۱۹۸۹ بنا نهاده شد.

پروتکل HTTP پروتکلی است که برای انتقال محتوای ابرمتن[۴] میان کاربران و رایانه های خدمات رسان[۵] مورد استفاده قرار می‌گیرد. این پروتکل داده را در قالب بسته ها [۶] مبادله می‌کند. در شکل زیر ساختار این پروتکل را مشاهده می‌کنید. بسته های این پروتکل شامل دو بخش سربرگ و بدنه می باشد. در سربرگ اطلاعات گیرنده، فرستنده، و ویژگی های بسته مانند اندازه‌ی بسته نوع ارتباط و مواردی از این دست قرار می‌گیرد. در بدنه اطلاعاتی که طرفین می‌خواهند با همدیگر مبادله کنند قرار می‌گیرد .

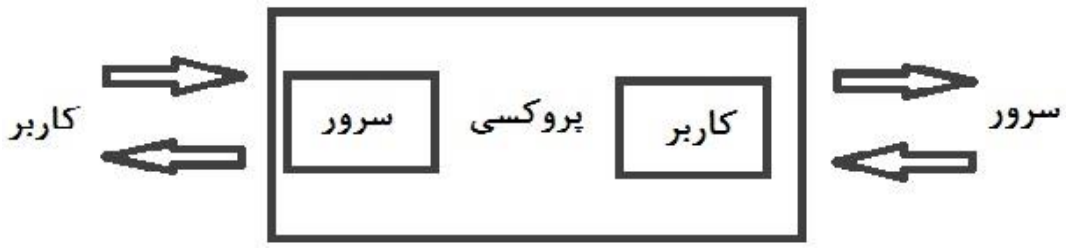
شکل ۱

[۱] Hypertext Transfer Protocol
[۲] Application
[۳] Tim Berners-Lee
[۴] Hyper Text
[۵] Computer Servers
[۶] Packets

حال که با این پروتکل بیشتر آشنا شدیم بهتر است ببینیم که چرا و چگونه امنیت این پروتکل تهدید می‌شود. به جهت همین منظور بهتر است آشنائی مختصری به تجهیزات میانی شبکه داشته‌باشیم. در ادامه با پروکسی‌ها به عنوان یک نمونه از عناصر میانی آشنا می‌شویم که بصورت گسترده ای مورد استفاده قرار می گیرد.

۱.۲ پروکسی[۷]

پروکسی‌ها یکی از عناصر میانی در شبکه‌های‌کامپیوتری هستند که به فراوانی مورد استفاده قرار می-گیرند. این عنصر برای کاربر نقش رایانه‌ی خدمات‌رسان، و برای رایانه‌ی خدمات‌رسان نقش کاربر را بازی می‌کند.



شکل ۲

۲.۲ کاربرد های پروکسی

۱.۲.۲ حسابرسی [۸]

کاربران قبل از اینکه به شبکه‌ی جهانی اینترنت دسترسی پیدا کنند نیاز است که حویت آنها احراز شود. و دلیلی هم ندارد که هویت افراد و اعمالی که در فضای مجازی انجام داده اند مخفی بماند. به همین منظور پروکسی‌ها به عنوان عناصر میانی وظیفه دارند که فعالیت های کاربران را نگه‌داری کنند و در صورت تخلف، این امور را به مراجع مربوطه ارجاع دهند.

---

[۷] Proxy
[۸] Accounting

۲.۲.۲ آنلاین مخفی ماندن[۹]

پروکسی درمعنا یعنی "وکیل" و در تعریف پروکسی‌ها هم گفتیم که پروکسی‌ها عناصری هستند که برای کاربر نقش سرور و برای سرور نقش کاربر را بازی می‌کنند. پس به نظر می رسد که فعالیت هائی که کاربران انجام می‌دهد به اسم پروکسی‌ها ثبت می‌شود و هر کاری که کاربر انجام داده به مثابه این است که پروکسی انجام داده و از این طریق هویت شخص مخفی می‌ماند. لازم به ذکر است که مخفی ماندن برخط در فضای مجازی شبیه به یک شوخی بامزه است. حتی اگر صدها پروکسی و عنصر میانی استفاده کنید نمی‌توانید هویت خود را در فضای مجازی مخفی نگه‌دارید.[۱۰]

۳.۲ امنیت پروتکل HTTP

مزایای پروکسی‌ها را بررسی کردیم اما این عناصر مشکلاتی را برای ما ایجاد می‌کنند و از جمله‌ی این مشکلات، تهدید امنیت در شبکه و به تبع آن تهدید امنیت پروتکل HTTP است.

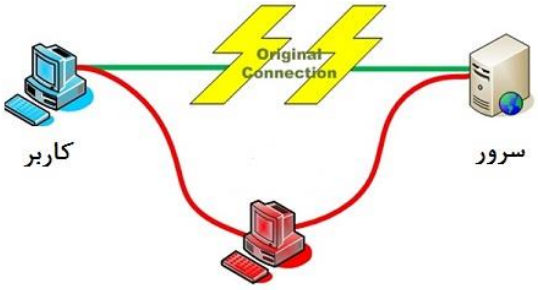
همانطور که در شکل زیر مشاهد می‌کنید دو حالت را بررسی می‌کنیم در حالت اول فرض می‌کنیم که هیچ عنصر میانی، میان کاربر و سرور قرار ندارد. در اینصورت هیچ مشکلی برای امنیت این پروتکل پیش نمی‌آید. اما رخ دادن چنین حالتی در شبکه غیر ممکن است و بطور قطع در شبکه عنصر میانی وجود دارد (حالت دوم) در اینصورت محتوای بسته‌ها توسط عناصر میانی قابل دسترسی و شنود است. و اگر اطلاعات مهم و محرمانه‌ای منتقل شود امنیت این اطلاعات غیر قابل تضمین است.

---

[۹] Stay Anonymous Online

[۱۰] بررسی جزئیات این مسئله در این اثر میسر نیست فقط در همین حد که به هیچ نحو نمی توان از پروکسی ها و عناصر میانی برای مخفی ماندن استفاده کرد. زیرا تمهیداتی اندیشیده شده است که می تواند فعالیت های افراد در فضای مجازی را، رد گیری کرد .

شکل ۳ـ الف_ زمانی که هیچ عنصر میانی، میان کاربر و سرور وجود ندارد

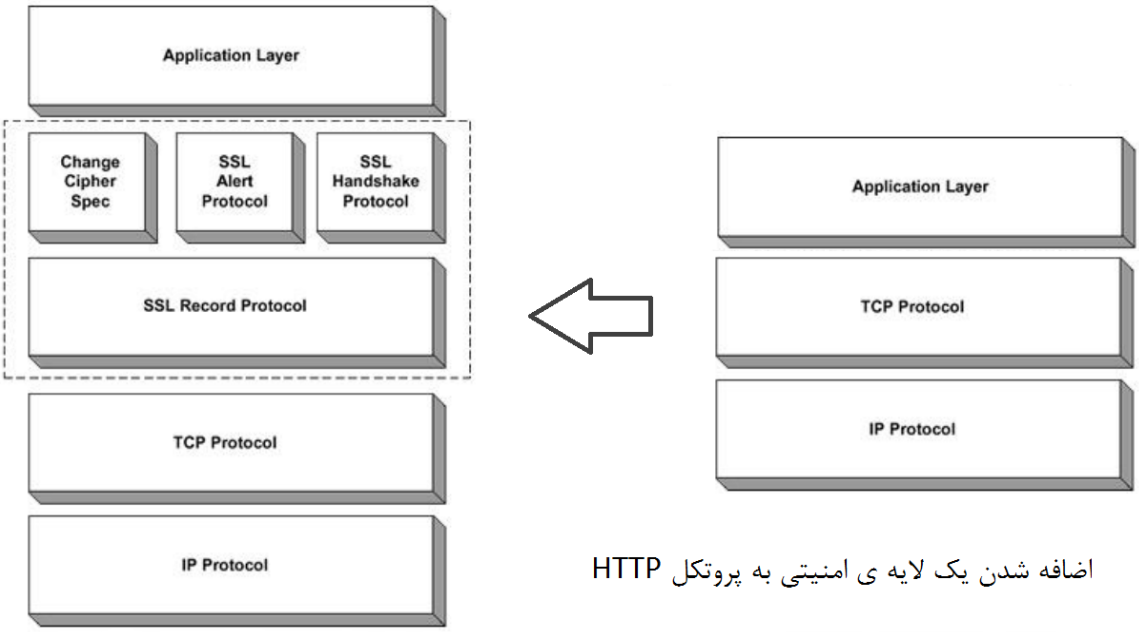


شکل ۳ـ ب_ زمانی که عنصر میانی، میان کاربر و سرور وجود دارد

در فصل آینده به راهکارها و بازنگری‌هائی در این پروتکل می‌پردازیم که این پروتکل را برای ما امن می‌کند.

۳ HTTPS[۱۱] راهکاری برای امن کردن پروتکل HTTP

در فصل قبل بررسی کردیم که عناصر میانی، در میان مسیر کاربر و سرور قرار می‌گیرند و پیام های مبادله شده را شنود می‌کنند. به نظر می رسد اگر پیام‌ها به گونه‌ای باشد که فقط گیرنده و فرستنده آن را بفهمند مشکل ما حل شده‌است. پس پروتکل HTTPS پروتکل HTTP را به اینصورت اصلاح می‌کند که پیام‌هائی که فرستنده و گیرنده با هم تبادل می‌کنند به گونه ای رمز گذاری شود، که فقط توسط خودشان قابل رمزگشائی باشد. این اصلاحیه با اضافه کردن یک لایه‌ی امنیتی[۱۲]، امنیت را فراهم می‌کند. این لایه از دو پروتکل TLS[۱۳] یا SSL[۱۴] استفاده می‌کند که اساس این دو پروتکل بر مبنای رمزگذاری اطلاعات است. این امر در شکل زیر قابل مشاهده است.

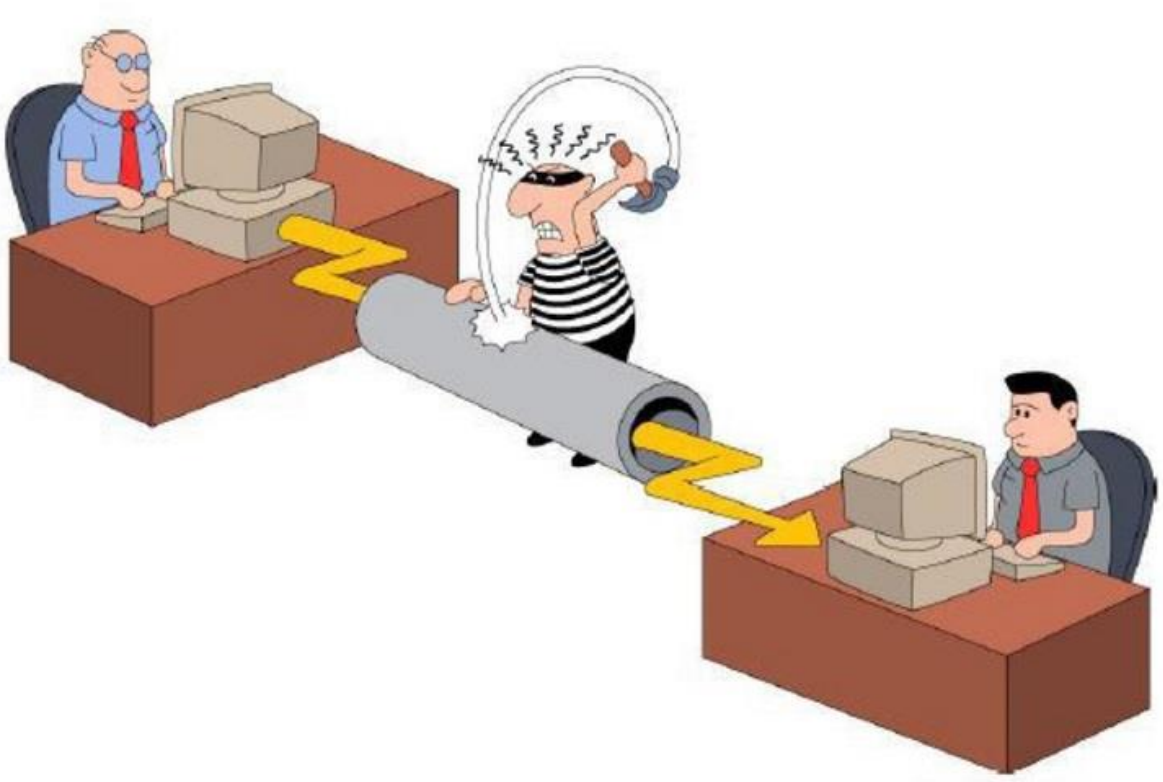


اضافه شدن یک لایه ی امنیتی به پروتکل HTTP

شکل ۴

[۱۱] Hypertext Transfer Protocol Secure
[۱۲] Secure Layer
[۱۳] Transport Layer Security
[۱۴] Secure Sockets Layer

همانطور که در شکل زیر مشاهده می‌کنید، این پروتکل مانند یک لایه حفاظتی، از نفوذ اشخاصی که می‌خواهند از اطلاعات سؤ استفاده کنند، جلوگیری می‌کند.



شکل ۵

۱.۳ بررسی الگوریتم های رمزگذاری اطلاعات

دو پروتکلی که لایه‌ی امنیت، برای برقراری امنیت استفاده می‌کند برمبنای رمزگذاری و رمز گشائی اطلاعات است. پس نیاز است بصورت مختصر الگوریتم های مورد استفاده را بررسی کنیم.

۲.۳ بررسی الگوریتم های ساده رمز گذاری

در شکل زیر یکی از این الگوریتم‌ها را بررسی می‌کنیم. در این الگوریتم برای رمزگذاری اطلاعات، هر حرف با سه حرف بعد از خودش جایگزین می‌شود. بطور مثال بجای T حرف W را قرار می‌دهیم و متن اصلی در شکل را به معادل کدشده تبدیل می‌کنیم و می‌بینیم که عبارتی که دارای مفهوم است، به یک عبارت غیرقابل فهم تبدیل شده است. بدیهی است که برای رمزگشائی لازم است که هرکاری که برای رمزگذاری انجام داده‌ایم، بصورت عکس انجام دهیم.

شکل ۶

۱.۲.۳ بررسی مشکلات الگوریتم های ساده

۱. به دلیل اینکه رمزگذاری اطلاعات توسط الگوریتم های ساده صورت می‌گرفت، ممکن بود با کمی بازی با متن کد شده، الگوریتم به کار رفته برای رمز گذاری حدس زده می‌شد و به کل امنیت اطلاعات به خطر می‌افتاد.

۲. فرض کنید که صد شرکت برای حفظ اطلاعات خود از یک الگوریتم استفاده می‌کردند در اینصورت اگر یکی از این شرکت‌ها سهل‌انگاری کند و الگوریتم، به دست افراد سؤاستفاده گر بیفتد، قطعا امنیت سایر شرکت‌ها به خطر می‌افتد.

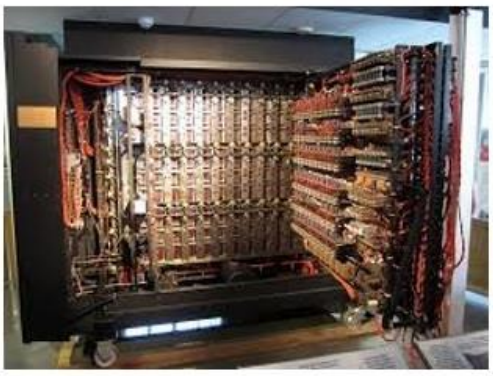
۳. چون عمل رمزگذاری توسط انسان انجام میشد مدت‌زمانی که طول می‌کشید که رمزگذاری انجام شود زیاد بود و همچنین این عمل به دلیل تعداد زیاد حروف که باید با معادل کد شده جایگزین می شدند، بسیار طاقت فرسا بود.

۳.۳ اختراع کامپیوتر ها

با اختراع کامپیوترها تحولی در رمزگذاری ایجاد شد. کامپیوترها مزیت هائی نسبت به ما داشتند که از جمله‌ی آن می‌توان به سرعت بالا، قابلیت اجرای الگوریتم های پیچیده‌تر، و همچنین عدم اشتباه در محاسبات اشاره کرد. جالب است که بدانید که کامپیوترهای برنامه پذیر بخاطر رمزگشائی اطلاعات اختراع شدند.

در جنگ جهانی دوم آلمان‌ها قبل از رد و بدل کردن پیام‌ها، آنها را رمزگذاری می‌کردند و کشور انگلستان و سایر کشورها که پیام ها را شنود می‌کردند با یکسری پیام‌های غیرقابل فهم مواجه می‌شدند به جهت همین موضوع یک تیم برای شکستن کدها استخدام شدند که این تیم به سرپرستی آلن تورینگ[۱۵] ماشینی اختراع کرد که قادر بود این رمزگذاری را بشکند. این کامپیوترها، نسل اولیه کامپیوترهای برنامه پذیر بودند. با این کار تورینگ؛ جنگ جهانی دوم حدود دو سال زودتر به پایان رسید و جان میلیون‌ها انسان بی‌گناه حفظ شد .

شکل ۷ـ آلن تورینگ



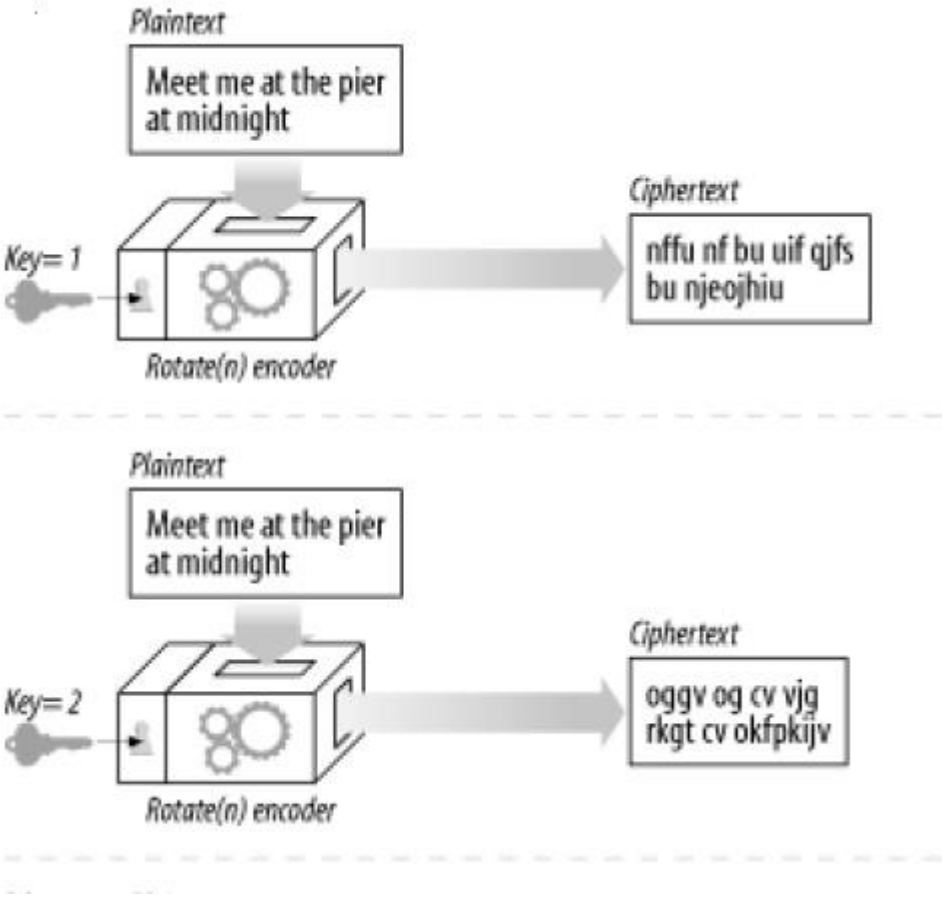
شکل ۸ ـ ماشین تورینگ

۴.۳ الگوریتم های رمزگذاری بر اساس کلید

به الگوریتم قبلی باز گردیم، به جای اینکه هرحرف را با سه حرف بعدی جایگزین کنیم هرحرف را با kتا حرف بعدی جایگزین می‌کنیم و مطابق شکل زیر الگوریتم یک کلید به عنوان ورودی دریافت می‌کند و اطلاعات را کد می‌کند. نکته‌ی قابل توجه این است که فقط سری ماندن کلید اهمیت دارد و اگر

[۱۵] Alan Turing

الگوریتم لو برود با مخفی ماندن کلید هیچ مشکلی برای امنیت اطلاعات پیش نمی‌آید. در شکل زیر مشاهده می‌کنید که خروجی به ازای هرکلید با دیگر خروجی‌ها تفاوت دارد.

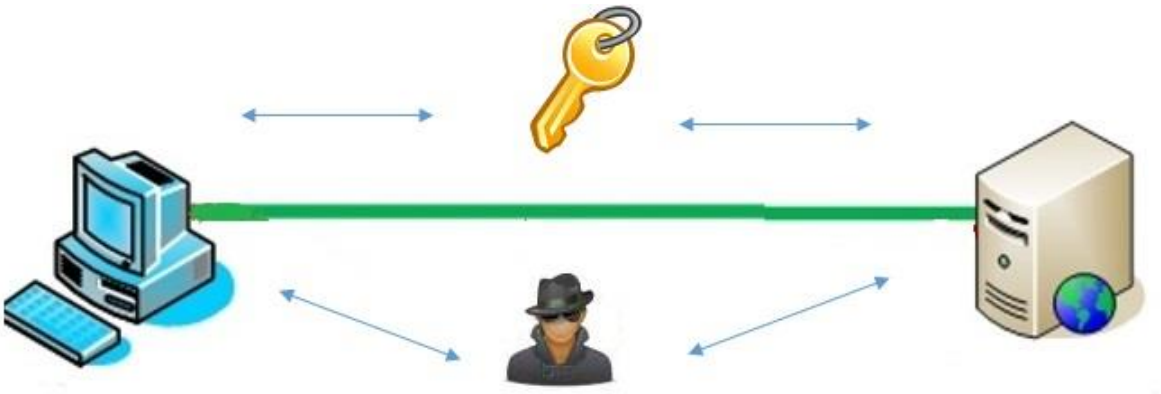


شکل ۹

۱.۴.۳ الگوریتم رمز گذاری کلید عمومی[۱۶]

این الگوریتم برای رمزگذاری و رمزگشائی اطلاعات از یک کلید عمومی استفاده می‌کند. به عبارت دیگر فرستنده و گیرنده برای رمزگذاری و رمزگشائی فقط از یک کلید یکسان استفاده می کنند. مشکلی که این الگوریتم دارد این است که فرستنده و گیرنده نیاز دارند به نحوی کلید را با یکدیگر مبادله کنند. اگر در میان مسیر کلید در دسترس دیگران قرار بگیرد، همه‌ی کارهای انجام شده بی‌فایده می‌شود و مانند این است که عمل کدگذاری اطلاعات صورت نگرفته باشد. در شکل زیر این امر به وضوح قابل مشاهده است.

[۱۶] Public Key



شکل۱۰

۲.۴.۳ الگوریتم کد گذاری بر اساس کلید اختصاصی[۱۷]

در الگوریتم کدگذاری بر اساس کلید عمومی مشکل، تبادل کلید عمومی بود. در این روش، الگوریتم رمزگذاری را با کلید عمومی انجام می دهد و رمزگشائی را فقط با کلید اختصاصی انجام می شود. حال همه از کلید عمومی اطلاع دارند اما فقط فرستنده و گیرنده از کلید اختصاصی خبر دارند و می‌توانند رمزگشائی را انجام دهند.

۳.۴.۳ مبادله ی کلید اختصاصی

حال که قرار است کلید اختصاصی فقط بصورت محرمانه در اختیار گیرنده و فرستنده باشد لازم است به گونه‌ای مبادله شود که فقط در سمت خودشان قابل مشاهده باشد. برای این منظور سرور یک الگوریتم در نظر می‌گیرد که با یک کلید عمومی رمزگذاری را انجام می‌دهد و فقط با کلید اختصاصی رمزگشائی صورت می‌پذیرد، سپس کلید عمومی را برای کاربر ارسال می‌کند. کاربر پس از دریافت، کلید اختصاصی خود را رمزگذاری می‌کند و برای سرور ارسال می‌کند. سرور پس از دریافت کلید آن را رمزگشائی کرده و از این پس از این کلید برای رمزگشائی استفاده می‌کند. در این روش مطمئن هستیم که این کلید محرمانه است و فقط فرستنده و گیرنده از آن خبر دارند.

[۱۷] Private Key

۴ جمع بندی و نتیجه گیری

با توجه به اهمیت حفظ اطلاعات، روش‌های امنیت دادن به اطلاعات را بررسی کردیم. دیدیم که می-توان با استفاده از پروتکل HTTPS امنیت کامل را در ارتباطات ایجاد کنیم و دیدیم که این پروتکل حفاظت اطلاعات را تضمین می‌کند. باید به این نکته توجه کنیم که بدون دلیل نباید از پروتکل HTTPS استفاده کرد، زیرا این پروتکل برای ما هزینه هائی به‌همراه دارد. در مواردی که نیاز به امنیت است باید از این پروتکل استفاده کنیم. بطور مثال سیستم‌های مالی مثل بانک‌ها، سیستم‌های دفاعی، سیستم‌های اطلاعاتی، و هر سازمان یا ارگانی که اطلاعات مهمی را رد و بدل می‌کند، باید از این پروتکل استفاده کند. در مقابل اطلاعاتی که مهم نیستند مانند اطلاعات یک وبگاه خبری یا یک وبگاه تبلیغاتی؛ نیازی به استفاده از پروتکل HTTPS نیست و بهتر است که از پروتکل HTTP استفاده کنیم.

منابع و مراجع

[۱] Z. Durumeric, J. Kasten, M. Bailey, and J. A. Halderman, "Analysis of the HTTPS certificate ecosystem," presented at the Proceedings of the ۲۰۱۳ conference on Internet measurement conference, Barcelona, Spain, ۲۰۱۳.

[۲] A. Goldberg, R. Buff, and A. Schmitt, "A comparison of HTTP and HTTPS performance," *Computer Measurement Group, CMG*۹۸, ۱۹۹۸.

[۳] J. Clark and P. C. van Oorschot, "SoK: SSL and HTTPS: Revisiting past challenges and evaluating certificate trust model enhancements," in *Security and Privacy (SP),* ۲۰۱۳ *IEEE Symposium on*, ۲۰۱۳, pp. ۵۱۱-۵۲۵.

[۴] F. Callegati, W. Cerroni, and M. Ramilli, "Man-in-the-Middle Attack to the HTTPS Protocol," *IEEE Security and Privacy,* vol. ۷, pp. ۷۸-۸۱, ۲۰۰۹.

[۵] N. Heninger, Z. Durumeric, E. Wustrow, and J. A. Halderman, "Mining your Ps and Qs: Detection of widespread weak keys in network devices," in *Presented as part of the* ۲۱*st USENIX Security Symposium (USENIX Security* ۱۲*)*, ۲۰۱۲, pp. ۲۰۵-۲۲۰.

[۶] D. Naylor, A. Finamore, I. Leontiadis, Y. Grunenberger, M. Mellia, M. Munafò*, et al.*, "The cost of the S in HTTPS," in *Proceedings of the* ۱۰*th ACM International on Conference on emerging Networking Experiments and Technologies*, ۲۰۱۴, pp. ۱۳۳-۱۴۰.

[۷] I. Anshel, M. Anshel, and D. Goldfeld, "An algebraic method for public-key cryptography," *Mathematical Research Letters,* vol. ۶, pp. ۲۸۷-۲۹۲, ۱۹۹۹.

[۸] T. Chomsiri, "HTTPS Hacking Protection," in *Advanced Information Networking and Applications Workshops,* ۲۰۰۷*, AINAW'*۰۷*.* ۲۱*st International Conference on*, ۲۰۰۷, pp. ۵۹۰-۵۹۴.

[۹] D. Schatzmann, W. Mühlbauer, T. Spyropoulos, and X. Dimitropoulos, "Digging into HTTPS: flow-based classification of webmail traffic," in *Proceedings of the* ۱۰*th ACM SIGCOMM conference on Internet measurement*, ۲۰۱۰, pp. ۳۲۲-۳۲۷.