امنیت شبکه – جلسه هشتم

رئوس مطالب

ادامه ابزارهای اصلی رمزنگاری

مرکز گواهی دیجیتال Certificate Authority

شبکه معتمدین PGP (Pretty Good Privacy)

امضا دیجیتال (Digital Signature)

توابع Hash

دستیابی به مکانیزمهای پایه امنیتی (CIA)به کمک ابزارهای رمزنگاری

Certificate Authority (CA) – مرکز گواهی دیجیتال

یک سوال مهم درباره عملکرد الگوریتمهای نامتقارن این است که چگونه از صحت کلیدهای عمومی که در اختیار ما قرارمی گیرند اطمینان حاصل کنیم؟ پاسخ این است که از یک نفر سوم (T) که به عنوان امین است و مقادیر صحیح کلیه کلیدهای عمومی را نزد خود نگه می دارد استفاده کنیم. این نفر سوم اصطلاحا مرکز گواهی دیجیتال Certificate Authority یا به اختصار CA نامیده می شود. هر کسی می باید در یکی از CA ها اطلاعات خود را ثبت کند که این اطلاعات شامل کلید عمومی خود ، هویت یا نام خود و نیز تاریخ انقضای اطلاعات ثبت شده میباشد. استاندارد X509 به طور مفصل در رابطه با CAها بحث می کند. یکی از مشهورترین و بزرگترین CAها VeriSign می باشد. برنامه نویسان در داخل برنامه خود لیستی از CAهای معتبر و کلید عمومی آنها را نگه میدارند. زمانیکه نیاز به تبادل اطلاعات بین A و B باشد، A تقاضای خود را که شامل دریافت کلید عمومی B است را با کلید عمومی CA که از پیش دراختیار دارد کد کرده به همراه کلید عمومی خود برای CA ارسال میکند. CA درخواست را دریافت کرده و با کلید خصوصی خود رمزگشایی می کند و متوجه می شود که A تقاضای دریافت کلید عمومی B را دارد، لذا کلید عمومی B را از پایگاه اطلاعاتی خود استخراج کرده و آن را با کلید عمومی A کد کرده برای A می فرستد. A اطلاعات ارسال شده را با کلید خصوصی

خود رمزگشایی کرده به کلید عمومی B به صورت امن دست میابد و می تواند از این کلید برای ایجاد یک ارتباط امن بین خود و B استفاده نماید.

شبکه ای از معتمدین یا روش PGP (Pretty Good Privacy)

استفاده از CAها پرهزینه است زیرا که مراکز CA جهت عضویت و ثبت کلید عمومی هزینه بالای سالیانه دریافت میکنند. یک روش ساده و کم هزینه بر مبنای اعتماد دوجانبه استوار شده است به دین ترتیب که اگر A و B به هم اعتماد داشته باشند و A وC نیز به هم اعتماد داشته باشند B و C نیز می توانند به هم اعتماد کنند و از آنجا که این رابطه اعتماد یک رابطه تعدی است به همین ترتیب بین امانت داران طرفین بسط داده می شود و یک شبکه از معتمدین Web of Trust ایجاد می کند. حال در این شبکه هرکسی که مقادیر کلید یا کلیدهای عمومی مطمئن در اختیار دارد می تواند آن را در اختیار بقیه قراردهد و چون همه به هم اطمینان دارند صحت اطلاعات تضمین شده است. روش PGP (Pretty Good Policy) از همین شبکه معتمدین برای بدست آوردن مطمئن کلیدهای عمومی استفاده می کند.

امضا دیجیتال Digital Signature

الگوریتم رمز نگاری نامتقارن و استفاده از کلیدهای عمومی وخصوصی امکان تشخیصی هویت را فراهم میکنند و زمینه ساز پیاده سازی مکانیزمی امن جهت تشخیص دیجیتالی هویت می گردند که اصطلاحا امضا دیجیتال (Digital Signature) نامیده می شود. امضا دیجیتال به لحاظ مفهومی بسیار شبیه امضا سنتی است ولی به لحاظ امنیت، امنیت امضا دیجیتال به لحاظ ریاضی تضمین شده است ولی امضا سنتی قابل جعل است.

الگوریتم نامتقارن مورد استفاده در تولید امضا دیجیتال اصطلاحا الگوریتم کدگذاری کلید عمومی نامیده می شود. دراین الگوریتم از کلید خصوصی (فقط در اختیار کد کننده) برای کد کردن و از کلید عمومی (دراختیار همه) برای دیکد کردن استفاده می شود بدین معنی که در این روش تنها اطلاعات کد شده ای با کلید عمومی باز می شوند که اطلاعات با کلید خصوصی متناظر آن کد شده باشند به عبارت دیگر وقتی با کلید عمومی رمز را با موفقیت می گشایید میتوانید مطمئن شوید که اطلاعات به کمک کلید خصوصی متناظر آن کد شده بوده است که در اختیار طرف مد نظر شما بوده است. روش عملکرد امضا دیجیتال به شکل ذیل است:

۱. در سمت فرستنده اطلاعات، اطلاعات به کمک یک تابع Hash مناسب (در قسمت بعد مفصلا در رابطه با توابع Hash توضیح داده میشود) به یک رشته اطلاعات کوچک (Digest) تبدیل می شود.
۲. فرستنده به کمک کلید خصوصی خود مقدار Digestرا کد کرده و اطلاعات کد شده را که همان امضا دیجیتال است همراه اطلاعات ارسال میکند.
۳. در سمت گیرنده به کمک همان تابع Hash از روی اطلاعات مجددا Digest تولید میشود و مقدار آن با مقدار دیکد شده Digest (عملیات دیکد به وسیله کلید عمومی بر روی امضا دیجیتال انجام می شود) ارسال شده از طرف فرستنده تطبیق داده می شود. بدین ترتیب اگر تفاوتی نباشد اطلاعات صحیح و از طرف فرد مورد نظر منتقل شده است.

ادامه ابزارهای اصلی رمزنگاری

توابع Hash

یکی از ابزارهای مهم رمزنگاری توابع Hash هستند. این توابع دارای خصوصیات ذیل میباشند:

- توابعی هستند که اطلاعات با طول متغییر را دریافت کرده و یک اطلاعات با طول کوتاه که اصطلاحا Digest (خلاصه) نامیده می شود تولید میکنند.
- توابع عموما در زمان بسیار بسیار کمی عمل میکنند و سرعت اجرای آنها بسیار بالا است.
- به لحاظ عملیاتی طوری هستند که هیچ دوداده متفاوت Digest مشابه تولید نمیکند.
- اگر Digest را داشته باشیم به لحاظ عملیاتی امکان تولید خود داده اصلی از روی آن ممکن نیست (یعنی زمان انجام آن آنقدر زیاد است که عملا امکان پذیر نیست – همان اصطلاح امن بودن به لحاظ محسباتی Computationally Secure)

برخی از کابردهای متداول توابع Hash :

- یک کاربرد مهم Hashها در ذخیره کلمات عبور می باشد.
- از توابع Hash می توان برای پیاده سازی مکانیزم جامعیت اطلاعات (Integrity) استفاده کرد به این ترتیب که اطلاعات را به یک تابع Hash داد و یک Digest تولید کرد و

Digest را به همراه اطلاعات به صورت امن برای گیرنده ارسال کرد. حال اگر گیرنده اطلاعات را دریافت کرده با همان الگوریتم Hash مقدار Digest را تولید نماید می باید این مقدار با Digest ارسالی یکی باشد در غیر اینصورت اطلاعات تغییر کرده و جامعیت آنها به هم خورده است.

- از کاربردهای دیگر توابع Hash استفاده از آنها در تولید اعداد نیمه تصادفی است که همانطور که قبلا اشاره شد نیاز به آنها در پیاده سازی مکانیزمهای امنیتی ضروری است.
- جهت امن سازی بیشتر توابع Hash معمولا از یک کلید ورودی به عنوان پارامتری در تابع Hash استفاده کرده و به کمک آن Digest را تولید میکنند.

دستیابی به مکانیزمهای پایه امنیتی (CIA)به کمک ابزارهای رمزنگاری

پس از توضیح هریک از چهار ابزار رمزنگاری در ذیل نحوه پیاده سازی هریک از مکانیزمهای پایه امنیتی (محرمانگی، جامعیت و تشخیص هویت Confidentiality Integrity and Authentication) آورده شده است:

الف) پیاده سازی محرمانگیConfidentiality:

۱. A یک کلید متقارن تولید کرده و به کمک کلید عمومی B آن را کد می کند.
۲. A کلید متقارن کد شده را برای B ارسال میکند.
۳. A پیامهای خود را با کلید متقارن کد کرده و برای Bمی فرستد
٤. فقط و فقط B است که پیام را میتواند بخواند زیرا کلید متقارن رمز نگاری را قبلا به صورت کد شده با کلید عمومی خود دریافت کرده و تنها اوست که میتواند کلید متقارن را به کمک کلید خصوصی خود رمز گشایی کرده به کمک آن اطلاعات مورد تبادل را رمزگشایی کند.

ب) پیاده سازی جامعیت Integrity:

۱. A پیام خود را Hashکرده و با کلید عمومی B کد میکند.
۲. A پیام را به همراه Digestکد شده برای B میفرستد

۳. B مقدار Digest را با کلید خصوصی خود دیکد کرده و از روی اطلاعات دریافتی مجددا Digest را محاسبه کرده و آن را با Digest دیکد شده مقایسه می کند اگر مساوی نباشند جامعیت اطلاعات برهم خورده است (اطلاعات تغییر کرده)

٤. یک واسط خرابکار (M) نمیتواند در اطلاعات مورد تبادل خدشه ایجاد کند زیرا نمیتواند Digest را دیکد کنده (کلید خصوصی B را ندارد) در نتیجه نمیتواند Digest جدیدی را باتوجه به اطلاعاتی که تغییر میدهد تولید کند و هر گونه تغییر در اطلاعات توسط B قابل تشخیص خواهد بود.

ج) پیاده سازی تشخیص هویت Authentication: (مشابه بحث امضا دیجیتال دربالا)

۱. A پیام خود را Hashکرده و Digest را با کلید خصوصی خود امضا دیجیتال میکند.
۲. A پیام و Digest امضا شده را برای B می فرستد.
۳. B امضا را تطابق می دهد زیرا کلید عمومی A را دارد و نیز میتواند با محاسبه مجددDigest صحت آن را تطابق دهد
٤. تنها کسی که میتوانسته آن Digest را امضا کند A بوده است زیرا فقط او کلید خصوصی را در اختیار داشته است.

د) پیاده سازی محرمانگی، جامعیت و تشخیص هویت (CIA) تواما:

۱. A یک کلید متقارن ایجاد کرده و آن را با کلید عمومی B کد میکند.
۲. Aکلید متقارن کد شده را برای B می فرستد.
۳. A ، Digest اطلاعات مورد ارسال را به کمک تابع Hashمناسب تولید کرده و آن را امضا دیجیتال می کند (با کلید خصوصی خود کد میکند)
٤. A کل پیام و Digest امضا شده را با کلید متقارن کد کرده برای B میفرستد.
٥. فقط B می تواند کلید متقارن را که با کلید عمومی خود کد شده دیکد کند زیرا فقط او کلید خصوصی را در اختیار دارد.
٦. B میتواند با محاسبه مجدد Digest صحت آن را چک کند
۷. B با کنترل امضا دیجیتال که برروی Digest زده شده می تواند هویت فرستنده (A) را تایید کند