استاندارد ملی ایران
۶-۱۰۸۲۵
چاپ اول
اردیبهشت ۱۳۹۲

جمهوری اسلامی ایران
Islamic Republic of Iran
سازمان ملی استاندارد ایران
Iranian National Standardization Organization

INSO
10825-6
1st. Edition
May.2013

فناوری اطلاعات – فنون امنیتی- احراز هویت هستار

قسمت ۶: سازوکارهای استفاده از انتقال دستی داده‌ها

Information technology – Security techniques — Entity authentication Part 6: Mechanisms using manual data transfer

ICS: 35.040

به نام خدا

آشنایی با سازمان ملی استاندارد ایران

مؤسسهٔ استاندارد و تحقیقات صنعتی ایران به موجب بند یک مادهٔ۳ قانون اصلاح قوانین و مقررات مؤسسهٔ استاندارد و تحقیقات صنعتی ایران، مصوب بهمن ماه ۱۳۷۱ تنها مرجع رسمی کشور است که وظیفه تعیین، تدوین و نشر استانداردهای ملی (رسمی) ایران را به عهده دارد.

نام موسسه استاندارد و تحقیقات صنعتی ایران به موجب یکصد و پنجاه و دومین جلسه شورای عالی اداری مورخ ۹۰/۶/۲۹ به سازمان ملی استاندارد ایران تغییر و طی نامه شماره ۲۰۶/۳۵۸۳۸ مورخ ۹۰/۷/۲۴ جهت اجرا ابلاغ شده است .

تدوین استاندارد در حوزه های مختلف در کمیسیون های فنی مرکب از کارشناسان سازمان ، صاحب نظران مراکز و مؤسسات علمی، پژوهشی، تولیدی و اقتصادی آگاه و مرتبط انجام می شود وکوششی همگام با مصالح ملی و با توجه به شرایط تولیدی، فناوری و تجاری است که از مشارکت آگاهانه و منصفانهٔ صاحبان حق و نفع، شامل تولیدکنندگان، مصرف‌کنندگان، صادرکنندگان و وارد کنندگان، مراکز علمی و تخصصی، نهادها، سازمان های دولتی و غیر دولتی حاصل می شود .پیش نویس استانداردهای ملی ایران برای نظرخواهی به مراجع ذی نفع و اعضای کمیسیون های فنی مربوط ارسال می‌شود و پس از دریافت نظرها و پیشنهادها در کمیتهٔ ملی مرتبط با آن رشته طرح و در صورت تصویب به عنوان استاندارد ملی (رسمی) ایران چاپ و منتشر می شود.

پیش نویس استانداردهایی که مؤسسات و سازمان های علاقه مند و ذی صلاح نیز با رعایت ضوابط تعیین شده تهیه می کنند درکمیتهٔ ملی طرح و بررسی و درصورت تصویب، به عنوان استاندارد ملی ایران چاپ و منتشر می شود .بدین ترتیب، استانداردهایی ملی تلقی می شوند که بر اساس مفاد نوشته شده در استاندارد ملی ایران شمارهٔ ۵ تدوین و در کمیتهٔ ملی استاندارد مربوط که سازمان ملی استاندارد ایران تشکیل می‌دهد به تصویب رسیده باشد.

سازمان ملی استاندارد ایران از اعضای اصلی سازمان بین المللی استاندارد (ISO)[1] ،کمیسیون بین المللی الکتروتکنیک (IEC)[2] و سازمان بین المللی اندازه شناسی قانونی (OIML)[3] است و به عنوان تنها رابط[4] کمیسیون کدکس غذایی (CAC)[5] در کشور فعالیت می کند. در تدوین استانداردهای ملی ایران ضمن توجه به شرایط کلی و نیازمندی های خاص کشور ، از آخرین پیشرفت های علمی، فنی و صنعتی جهان و استانداردهای بین‌المللی بهره‌گیری می‌شود .

سازمان ملی استاندارد ایران می تواند با رعایت موازین پیش بینی شده در قانون، برای حمایت از مصرف کنندگان، حفظ سلامت و ایمنی فردی و عمومی، حصول اطمینان از کیفیت محصولات و ملاحظات زیست محیطی و اقتصادی، اجرای بعضی از استانداردهای ملی ایران را برای محصولات تولیدی داخل کشور و/یا اقلام وارداتی، با تصویب شورای عالی استاندارد، اجباری نماید. سازمان می تواند به منظور حفظ بازارهای بین المللی برای محصولات کشور، اجرای استاندارد کالاهای صادراتی و درجه‌بندی آن را اجباری نماید. همچنین برای اطمینان بخشیدن به استفاده کنندگان از خدمات سازمان ها و مؤسسات فعال در زمینهٔ مشاوره، آموزش، بازرسی، ممیزی و صدورگواهی سیستم های مدیریت کیفیت و مدیریت زیست‌محیطی، آزمایشگاه ها و مراکز کالیبراسیون (واسنجی) وسایل سنجش، سازمان ملی استاندارد ایران این گونه سازمان ها و مؤسسات را بر اساس ضوابط نظام تأیید صلاحیت ایران ارزیابی می کند و در صورت احراز شرایط لازم، گواهینامهٔ تأیید صلاحیت به آن ها اعطا و بر عملکرد آن ها نظارت می کند. ترویج دستگاه بین المللی یکاها، کالیبراسیون (واسنجی) وسایل سنجش، تعیین عیار فلزات گرانبها و انجام تحقیقات کاربردی برای ارتقای سطح استانداردهای ملی ایران از دیگر وظایف این سازمان است.

1- International Organization for Standardization
2 - International Electrotechnical Commission
3- International Organization of Legal Metrology (Organisation Internationale de Metrologie Legale)
4 - Contact point
5 - Codex Alimentarius Commission

کمیسیون فنی تدوین استاندارد

« فناوری اطلاعات – فنون امنیتی- احراز هویت هستار - قسمت ۶: سازوکارهای استفاده از انتقال دستی داده‌ها »

رئیس :	**سمت و/ یا نمایندگی**
سعیدی، عذرا (کارشناسی ارشد مهندسی برق مخابرات)	کارشناس تدوین استاندارد سازمان فناوری اطلاعات
دبیر :	
میراسکندری، سید محمدرضا (کارشناسی مهندسی کامپیوتر نرم افزار)	مدیر کل خدمات ارزش افزوده سازمان فناوری اطلاعات
اعضا: (اسامی به ترتیب حروف الفبا)	
بختیاری، شیرین (کارشناسی مهندسی برق)	کارشناس تدوین استاندارد سازمان فناوری اطلاعات
سلطانی حقیقت، الهه (کارشناسی مهندسی برق مخابرات)	کارشناس تدوین استاندارد سازمان فناوری اطلاعات
فرهاد شیخ احمد، لیلا (کارشناسی ارشد مهندسی کامپیوتر نرم افزار)	کارشناس تدوین استاندارد سازمان فناوری اطلاعات
فولادیان، مجید (کارشناسی ارشد مهندسی برق مخابرات)	مشاور سازمان فناوری اطلاعات
فیاضی، مهدی (کارشناسی مهندسی برق مخابرات)	کارشناس مسئول تدوین استاندارد و امنیت شبکه سازمان فناوری اطلاعات
قسمتی، سیمین (کارشناسی ارشد فناوری اطلاعات)	کارشناس تدوین استاندارد سازمان فناوری اطلاعات
موجبی، محمود (کارشناسی ارشد مخابرات)	کارشناس تدوین استاندارد سازمان فناوری اطلاعات

میرزایی رضایی، طیبه (کارشناسی ارشد فیزیک)	رئیس اداره تدوین استانداردها و نظارت بر امنیت سرویس‌ها سازمان فناوری اطلاعات
ناظمی، اسلام (دکتری کامپیوتر)	استادیار دانشگاه شهید بهشتی
تصیری آسایش، حمیدرضا (کارشنایی ارشد مهندسی فناوری اطلاعات)	نماینده دانشگاه شهید بهشتی
نیسی مینایی، آصف (کارشناسی مهندسی فناوری اطلاعات)	نماینده دانشگاه شهید بهشتی
یعقوبی رفیع، کمال‌الدین (کارشناسی ارشد مهندسی فناوری اطلاعات)	نماینده دانشگاه شهید بهشتی

فهرست مندرجات

عنوان **صفحه**

پیش‌گفتار

استاندارد «فناوری اطلاعات - فنون امنیتی - احراز هویت هستار - سازوکارهای استفاده از انتقال دستی داده‌ها» که پیش‌نویس آن درکمیسیون‌های مربوط توسط سازمان فناوری اطلاعات تهیه و تدوین شده و در دویست و بیست و ششمین اجلاس کمیته ملی استاندارد رایانه و فرآوری داده مورخ ۹۱/۹/۲۱ مورد تصویب قرار گرفته است، اینک به استناد بند یک ماده ۳ قانون اصلاح قوانین و مقررات سازمان استاندارد و تحقیقات صنعتی ایران، مصوب بهمن ماه ۱۳۷۱، به‌عنوان استاندارد ملی ایران منتشر می‌شود.

برای حفظ همگامی و هماهنگی با تحولات و پیشرفت‌های ملی و جهانی در زمینه صنایع، علوم و خدمات، استانداردهای ملی ایران در مواقع لزوم تجدید نظر خواهد شد و هر پیشنهادی که برای اصلاح و تکمیل این استانداردها ارائه شود، هنگام تجدید نظر در کمیسیون فنی مربوط مورد توجه قرار خواهد گرفت . بنابراین، باید همواره از آخرین تجدیدنظر استانداردهای ملی استفاده کرد.

منبع و ماخذی که برای تهیه این استاندارد مورد استفاده قرار گرفته به شرح زیر است:

ISO/IEC 9798-6: 2010, Information technology — Security techniques — Entity authentication — Part 6: Mechanisms using manual data transfer . + Cor. 1:2009

مقدمه

در شبکه‌های وسایل ارتباطی، اغلب ضروری است که دو افزاره[1] بتوانند از طریق یک کانال، رویه احراز هویت یک هستار را انجام دهند. این کانال می‌تواند موضوع هر دو نوع حمله فعال و انفعالی باشد که حمله فعال می‌تواند شامل معرفی داده از طرف یک متخاصم سوم شخص یا دستکاری، حذف یا تکرار داده به طریق قانونی باشد که بر روی کانال فرستاده می‌شود.

در قسمت‌های دیگر استاندارد ISO/IEC 9798 سازوکارهای کاربردی از احراز هویت هستار در مواقعی که دو افزاره یک کلید خصوصی را به اشتراک گذاشته‌اند یا مواقعی که یک افزاره نسخه‌ای اصالت سنجی شده از کلید عمومی افزاره دیگر را دارد، مشخص می‌شود.

در این استاندارد، سازوکارهایی از احراز هویت هستار که در آنها هیچ‌گونه روابط تعیین کلید از قبل وجود ندارد و در واقع سازوکارهای دستی احراز هویت مشخص می‌شوند. در عوض احراز هویت هستار، با استفاده از انتقال دستی رشته کوتاهی از داده یک افزاره به افزاره دیگر یا با مقایسه دستی خروجی‌های رشته‌ای کوتاهی از دو افزاره، انجام می‌شود..

با توجه به هدف این استاندارد، معنای عبارت احراز هویت از معنای به‌کار رفته در قسمت‌های دیگر استاندارد ISO/IEC 9798 متفاوت است. به‌جای این‌که یک افزاره، هویت افزاره دیگر را بیازماید (یا برعکس) هر دو افزاره مالکیت تأیید کاربر را دارا هستند و رشته‌ای از داده را با هم در زمان اجرای سازوکار به اشتراک گذاشته‌اند. مطمئناً این رشته داده می‌تواند شامل شناساگر هویتی برای یک یا هر دو افزاره باشد.

همان‌طور که در پیوست‌های اطلاعاتی ب و پ توضیح داده شده است، سازوکار احراز هویت دستی می‌تواند به‌عنوان پایه‌ای برای استقرار کلید سری یا تبادل قابل اطمینان کلیدهای عمومی باشد. همچنین سازوکار تشخیص هویت دستی می‌تواند برای تبادل قابل اطمینان پارامترهای امنیتی عمومی یا سّری، از جمله بیانیه‌های سیاست امنیتی یا مهرهای زمانی، استفاده شود.

1- Device

فناوری اطلاعات – فنون امنیت- احراز هویت هستار –
قسمت ۶ : سازوکارهای استفاده از انتقال دستی داده‌ها

۱ هدف و دامنه کاربرد

هدف از تدوین این استاندارد، تعیین ویژگی‌های هشت سازوکار احراز هویت هستار مبتنی بر انتقال دستی داده‌ها بین افزاره‌های تعیین‌کننده هویت است. در این استاندارد، چگونگی استفاده از این سازوکارها برای حمایت از توابع مدیریت کلید نشان داده می‌شود و راهنماهایی به‌منظور انتخاب امن پارامترها برای این سازوکارها ارائه می‌شود. مقایسه‌ای از سطوح امنیتی و کارایی مهیا شده توسط این هشت سازوکار نیز ارائه می‌شود.

چنین سازوکارهایی، می‌توانند در شرایط متنوعی مناسب باشند. یکی از این کاربردها می‌تواند در شبکه‌های شخصی رخ دهد، جایی که مالک دو افزاره شخصی که امکان ارتباطات بی‌سیم دارند، می‌خواهد رویه احراز هویت هستار را به‌عنوان قسمتی از فرایند آماده‌سازی آنها برای استفاده در شبکه انجام دهد.

۲ مراجع الزامی

مدارک الزامی زیر حاوی مقرراتی است که در متن این استاندارد ملی ایران به آنها ارجاع داده شده است. بدین ترتیب آن مقررات جزئی از این استاندارد ملی ایران محسوب می شود.

در صورتی که به مدرکی با ذکر تاریخ انتشار ارجاع داده شده باشد، اصلاحیه‌ها و تجدید نظرهای بعدی آن مورد نظر این استاندارد ملی نیست. در مورد مدارکی که بدون ذکر تاریخ انتشار به آن‌ها ارجاع داده شده است، همواره تاریخ تجدید نظر و اصلاحیه‌های بعدی آن‌ها مورد نظر است.
استفاده از مرجع زیر برای این استاندارد الزامی است:

۲-۱ استاندارد ملی شماره ۱-۹۷۹۸ : ۱۳۹۱[1]، فناوری اطلاعات – فنون امنیتی – احراز هویت هستار – قسمت ۱ : کلیات

1_ ISO/IEC 9798-1:2010

۳ اصطلاحات و تعاریف

در این استاندارد، علاوه بر اصطلاحات و تعاریف تعیین شده در استاندارد ملی شماره ۱-۹۷۹۸ :۱۳۹۱، اصطلاحات و تعاریف زیر نیز به‌کار می‌رود:

۱-۳

مقدار-وارسی[1]

رشته‌ای از بیت‌های محاسبه شده به‌عنوان خروجی یک تابع مقدار- وارسی، که از یک پیام‌ساز داده به دریافت‌کننده‌ی داده فرستاده شده و دریافت کننده‌ی داده را قادر می‌سازد که درستی آن را وارسی کند.

۲-۳

تابع مقدار-وارسی[2]

تابع f که رشته‌ای از بیت‌ها و کلید سرّی کوتاه، یعنی کلیدی که به‌راحتی می‌تواند وارد یا از افزاره کاربر خوانده شود، را به یک رشته با طول ثابت از بیت‌ها، یعنی یک مقدار- وارسی b بیت، که خاصیت‌های زیر را برآورده می‌نماید، نگاشت می‌کند:

- برای هر کلید k و هر رشته ورودی d، تابع $f(d, k)$ می‌تواند به‌صورت کارا محاسبه شود.
- پیدا کردن یک جفت از رشته‌های داده‌ای مجزای (d, d') که برای آن‌ها تعداد کلیدهایی که رابطه $f(d, k) = f(d', k)$ را برآورده می‌کنند بیشتر از کسر کوچکی از مجموعه کلیدهای ممکن باشد، از نظر محاسباتی غیر عملی باشد.

یادآوری- در عمل، یک کلید کوتاه به‌طور معمول شامل ۴ تا ۶ رقم و یا کاراکترهای الفبایی است.

۳-۳

احراز هویت منشاء داده[3]

تایید این که منشاء داده‌ی دریافت شده، همان است که ادعا شده است.

[ISO 7498-2]

۴-۳

مقدار- خلاصه[4]

رشته‌ای از بیت‌های محاسبه شده به‌عنوان خروجی یک تابع خلاصه که از یک پیام‌ساز داده به دریافت کننده‌ی داده فرستاده شده و دریافت کننده‌ی داده را قادر می‌سازد که درستی آن را وارسی کند.

1- Check-value
2- Check-Value Function
3- Data origin authentication
4- Digest-value

۵-۳

تابع خلاصه[1]

تابع d که رشته‌ای از بیت‌ها و کلید سرّی بلند را به یک رشته کوتاه و با طول ثابت از بیت‌ها، یعنی مقدار-خلاصه b بیتی، نگاشت می‌کند که به راحتی می‌تواند وارد یا از افزاره کاربر خوانده شود، و خاصیت‌های زیر را برآورده می‌کند:

- برای هر کلید k و هر رشته ورودی m، تابع $d(m, k)$ می‌تواند به‌صورت کارا محاسبه شود.
- پیدا کردن یک جفت از رشته‌های داده مجزای (m, m') که برای آن‌ها نسبت کلیدهایی که رابطه $d(m, k) = d(m', k)$ برآورده می‌کنند، بیشتر از $(2^{-b} + \varepsilon)$ است که در آن b، طول بیت مقدار-خلاصه و ε، ارزشی قابل صرف نظر در رابطه با 2^{-b} است، از نظر محاسباتی غیر عملی باشد.

یادآوری ۱- در عمل، اگر کلید k به اندازه یک مقدار درهم ساز معمولی رمزنگاشتی است، خاصیت تابع خلاصه دوم باید برآورده شود، به‌عنوان مثال، مقدار کلید k می‌تواند ۱۶۰ بیت باشد. این الزام از مرزهای پایینی نظری در طول کلیدی برای توابع درهم‌ساز جهانی، که یک کلاس عمومی از توابع خلاصه هستند، ناشی می‌شود. بحث‌های مفصل بیشتر این موضوع را می‌توانید در پیوست ج بیابید.

یادآوری۲- برای بحث‌های بیشتر در مورد کلید و طول‌های خلاصه، به پیوست‌های ت، ج و چ مراجعه کنید.

۶-۳

تابع-درهم ساز[2]

تابعی است که رشته‌هایی از بیت‌ها را به رشته‌هایی از بیت‌ها با طول ثابت می‌نگارد، و خواص زیر را برآورده می‌سازد:

- برای یک خروجی مفروض، یافتن ورودی‌ای که آن خروجی را نتیجه دهد از لحاظ محاسباتی غیر عملی باشد.
- برای یک ورودی مفروض، یافتن ورودی دیگری که همان خروجی را نتیجه دهد از لحاظ محاسباتی غیر عملی باشد.

یادآوری- عملی بودن از لحاظ محاسباتی، به الزامات خاص امنیتی و محیطی بستگی دارد.

۷-۳

گواهی احراز هویت دستی[3]

ترکیب یک کلید سرّی و یک مقدار-وارسی، تولید شده توسط یکی از دو افزاره درگیر در احراز هویت دستی، با این خاصیت که زمانی که به افزاره دیگر وارد شود، این جفت مقدارها را می‌توان برای تکمیل فرآیند احراز

1- Digest function
2- Hash-function
3- Manual Authentication Certificate

هویت دستی در زمان بعدی مورد استفاده قرار داد.

۸-۳

کد احراز هویت پیام (MAC[1])

رشته‌ای از بیت‌ها که خروجی یک الگوریتم MAC است.

[ISO/IEC 9797-1]

۹-۳

الگوریتم کد احراز هویت پیام[2]

الگوریتم MAC

الگوریتم برای محاسبه تابعی که رشته‌های بیت‌ها و یک کلید سرّی را به رشته‌های با طول ثابت از بیت‌ها می‌نگارد و خاصیت‌های زیر را برآورده می‌کند:

- برای هر کلید و هر رشته ورودی، تابع می‌تواند به‌صورت کارا محاسبه شود؛
- برای هر کلید ثابت و با توجه به هیچ دانش قبلی از کلید، محاسبه مقدار تابع در هر رشته ورودی جدید از نظر محاسباتی غیر ممکن است، حتی اگر دانشی از رشته‌های ورودی و مقادیر تابع مربوطه داده شود، که در آن مقدار رشته ورودی i ام ممکن است پس از مشاهده مقدار اولین i-1 مقادیر تابع، انتخاب شده باشد.

[ISO/IEC 9797-1]

۱۰-۳

احراز هویت دستی هستار[3]

فرایند دستیابی به احراز هویت هستار بین دو افزاره با استفاده از ترکیبی از تبادل پیام از طریق (به‌طور بالقوه ناامن) کانال ارتباطاتی و انتقال دستی مقدار محدودی از داده‌ها بین افزاره‌ها.

۱۱-۳

واسط ورودی ساده[4]

واسطی برای افزاره که به کاربر اجازه می‌دهد که اتمام موفق یا ناموفق یک رویه را به افزاره نشان دهد، به‌عنوان مثال می‌تواند به‌عنوان یک جفت از دکمه‌ها یا تنها یک دکمه که در یک بازه زمانی خاص فشرده شده است یا خیر، پیاده‌سازی شود.

1- Message Authentication Code
2- Message Authentication Code algorithm
3- Manual entity authentication
4- Simple input interface

۱۲-۳

واسط خروجی ساده[1]

واسطی برای افزاره که اجازه می‌دهد تا کاربر به افزاره اتمام موفقیت آمیز یا ناموفق یک رویه را نشان دهد، به‌عنوان مثال می‌تواند توسط چراغ‌های قرمز و سبز یا به‌عنوان نور واحد باشند که در راه‌های مختلف برای نشان دادن موفقیت یا عدم موفقیت روشن می‌شوند.

۴ نمادها و کوته‌نوشت‌ها[2]

A، B	برچسب‌هایی که برای دو افزاره‌ی درگیر در یک سازوکار احراز هویت دستی هستار، استفاده می‌شوند.
d	تابع خلاصه‌ای که در سازوکارهای ۳ و ۵ استفاده می‌شود که $d(D, k)$ نشان‌دهنده خلاصه‌ی محاسبه شده بر روی رشته داده‌ای D با استفاده از کلید k است.
D	رشته داده‌ای که مقدار آن به‌عنوان نتیجه انجام یک سازوکار احراز هویت هستار دستی بین افزاره‌های A و B برقرار می‌شود.
h	تابع درهم ساز که در سازوکارهای ۳ تا ۶ استفاده شده است.
I_A، I_B	شناساگرهای تشخیص A و B
K	کلید سرّی (کوتاه) استفاده شده با یک تابع مقدار- وارسی در سازوکارهای ۱ و ۲
k	کلید سرّی (بلند) استفاده شده در سازوکارهای ۳ تا ۶
K_A، K_{Ai}، K_B، K_{Bi}	کلیدهای MAC تصادفی استفاده شده در سازوکارهای ۷ و ۸
MAC	کد احراز هویت پیام
R	رشته بیت تصادفی (کوتاه) استفاده شده در سازوکارهای ۴، ۶، ۷ و ۸
$\|\|$	همان‌طور که در استاندارد ملی شماره ۱-۹۷۹۸ :۱۳۹۱ تعریف شد، $X\|\|Y$ به معنی نتیجه الحاق موارد داده‌ای X و Y به ترتیب مشخص شده است. در مواردی که دو یا چند مورد داده‌ای ورودی یک الگوریتم رمزنگاشتی، به‌عنوان بخشی از یک سازوکار احراز هویت هستند، این نتیجه به‌گونه‌ای ترتیب داده خواهد شد که بتوان منحصراً آن را به رشته‌های داده‌ای سازنده‌اش تجزیه نمود، تا به‌عبارت دیگر هیچ‌گونه احتمال ابهام در تفسیر وجود نداشته باشد. می‌توان به این ویژگی پایانی، بسته به برنامه

1- Simple output interface
2- Symbols and abbreviated terms

کاربردی، از راه‌های گوناگون دست یافت. به‌عنوان مثال، این ویژگی را می‌توان به دو طریق تضمین کرد: (الف) تثبیت طول هرکدام از زیررشته‌ها در کل دامنه استفاده از سازوکار؛ (ب) کدگذاری دنباله رشته‌های متوالی با استفاده از روشی که کدگشایی یکتایی را ضمانت کند، مثلاً استفاده از قوانین برجسته کدگذاری مطابق با استاندارد ISO/IEC 8825-1 [۱۰].

یادآوری- پیوست‌های ت و ث راهنماهایی در انتخاب‌های مقتضی برای طول‌های کلیدهای MAC و سری کوتاه فراهم می‌کنند.

۵ الزامات کلی

سازوکارهای مشخص شده در این مستند، علاوه بر الزامات مشخص شده در بندهای ۶، ۷ و ۸، دارای الزامات زیر هستند:

الف- هر دو افزاره‌ای که رویه احراز هویت هستار دستی را انجام می‌دهند، باید توسط یک پیوند ارتباطاتی به یکدیگر متصل باشند (برای مثال یک پیوند بی‌سیم یا اینترنت). هیچ فرض امنیتی در ارتباط با این پیوند در نظر گرفته نشده است؛ یعنی، سازوکارها برای عمل کردن امن حتّی در محیطی که یک حمله کننده می‌تواند داده‌های منتقل شده در پیوند را پایش کند، طراحی شده‌اند.

ب- هر دو افزاره‌ای که رویه احراز هویت هستار دستی را انجام می‌دهند، باید دارای یک واسط کاربر که قادر به ورود و خروج داده است، باشند.

پ- واسط ورود داده کاربر برای یک افزاره، باید حداقل بتواند موفقیت یا عدم موفقیت تکمیل شدن یک رویه را نشان دهد (برای مثال می‌تواند با استفاده از دو یا یک دکمه که در بازه زمانی معینی فشار داده شده یا نشده باشد، پیاده سازی شود)؛ به چنین وسیله ورود داده، واسط ورودی ساده اطلاق می‌شود. در مقابل، واسط ورود استاندارد باید ابزاری برای ورود رشته کوتاهی از نمادها مانند یک عدد، یک عدد در مبنای شانزده یا یک ورودی حرفی‌عددی فراهم کند. مگر این که به‌طور صریح غیر از این بیان شده باشد که در این‌صورت ضروری است که هر افزاره وسیله استاندارد ورود داده داشته باشد.

ت- واسط خروجی داده کاربر برای یک افزاره، باید حداقل بتواند موفقیت یا عدم موفقیت تکمیل شدن یک رویه را نشان دهد (برای مثال می‌تواند با استفاده چراغ‌های سبز و قرمز، پیاده‌سازی شود)؛ به چنین وسیله خروج داده، واسط خروجی ساده اطلاق می‌شود. در مقابل، واسط خروج استاندارد باید ابزاری برای خروج رشته کوتاهی از نمادها مانند یک نمایش عددی، در مبنای شانزده یا حرفی‌عددی فراهم کند. مگر این که صراحتاً غیر از این بیان شده باشد که در این‌صورت ضروری است که هر افزاره وسیله استاندارد خروج داده داشته باشد.

ث- برای سازوکارهای ۱ و ۲، دو افزاره‌ای که رویه احراز هویت هستار را انجام می‌دهند، باید بر روی

استفاده از یک تابع مقدار-وارسی مشخص توافق کرده باشند و باید وسایل پیاده‌سازی این تابع را داشته باشند.

یادآوری۱- راهنما، در مورد انتخاب‌های مقتضی برای توابع مقدار-وارسی و طول‌ها برای مقدار-وارسی‌ها و کلیدهای تصادفی برای استفاده در سازوکارهای ۱ و ۲، در پیوست ت فراهم شده است. یک ساختار برای تابع مقدار-وارسی مطلقاً امن مناسب برای استفاده در سازوکارهای ۱ و ۲، در پیوست ث آمده است.

ج- برای سازوکارهای ۳ تا ۶، دو افزاره‌ای که رویه احراز هویت هستار را انجام می‌دهند، باید بر روی استفاده از یک تابع درهم‌ساز مشخص h توافق کرده باشند و باید وسایل پیاده‌سازی این تابع را داشته باشند.

یادآوری۲- راهنما در مورد انتخاب‌های مقتضی برای طول‌های بیت برای ورودی‌ها و خروجی‌های تابع درهم‌ساز برای استفاده در سازوکارهای ۳ تا ۶، در پیوست ت فراهم شده است.

چ- برای سازوکارهای ۳ و ۵، دو افزاره‌ای که رویه احراز هویت هستار را انجام می‌دهند، باید بر روی استفاده از یک تابع خلاصه مشخص d توافق کرده باشند و باید وسایل پیاده‌سازی این تابع را داشته باشند.

یادآوری ۳- راهنما در مورد طول‌های خلاصه‌ها برای استفاده در سازوکارهای ۳ و ۵ در پیوست ت فراهم شده است. ساخت‌ها[1] برای توابع خلاصه‌ای که از الگوریتم‌های MAC و توابع درهم‌ساز استفاده می‌کنند و مناسب برای استفاده در سازوکارهای ۳ و ۵ هستند، در پیوست چ آمده است.

ح- برای سازوکارهای ۷ و ۸، دو افزاره‌ای که رویه احراز هویت هستار را انجام می‌دهند، باید بر روی استفاده از یک الگوریتم MAC مشخص توافق کرده باشند و باید وسایل پیاده‌سازی این تابع را داشته باشند.

یادآوری ۴- راهنما در مورد انتخاب‌های مقتضی برای الگوریتم‌های MAC و طول‌ها برای MACها و کلیدهای تصادفی برای استفاده در سازوکارهای ۷ و ۸، در پیوست ت فراهم شده است.

خ- قبل از فراخوانی سازوکارهای ۱ تا ۸، دو افزاره‌ای که سازوکار را انجام می‌دهند، باید یک رشته داده‌ای D را ردوبدل کرده باشند (در ترکیب با یک مقدار-درهم‌ساز در سازوکارهای ۳ تا ۶). رشته داده‌ای D ممکن است توسط یک افزاره تولید و به افزاره دیگر فرستاده شود، یا ممکن است از الحاق داده‌های تولید شده از سوی دو افزاره تشکیل شود و توسط پیوند ارتباطی مشترک در هر دو جهت فرستاده

1- Constructions

شود.

د- یک کاربر انسانی با در اختیار داشتن هر دو افزاره باید با هر دو آنها کار کند، یا دو افزاره باید توسط دو کاربر که یک وسیله ارتباطی مورد اعتماد را به اشتراک گذاشته‌اند، عمل کنند.

ذ- باید اجرای کامل عملیات برای اطمینان از درستی پردازش این سازوکارها، به کاربران افزاره‌ها ارائه شود. نباید در حین انتقال دستی داده بین افزاره‌ها تاخیر قابل توجهی وجود داشته باشد. افزاره‌ها برای جلوگیری از حملات قطعی باید به‌صورت خودکار، همان‌طور که در مشخصات سازوکار نشان داده شد، متوقف شوند.

۶ سازوکارهای استفاده کننده از یک مقدار- وارسی کوتاه

۶-۱ کلیات

در این بند دو مورد از سازوکارهای احراز هویت دستی مبتنی بر استفاده از مقدارـوارسی مشخص می‌شوند. این دو سازوکار برای انواع مختلف افزاره‌ها مناسب هستند، به صورت خاص:

- سازوکار اول (سازوکار ۱) در مواردی که یک افزاره دارای واسط ورودی ساده و افزاره دیگر دارای واسط خروجی ساده است، مناسب است.
- سازوکار دوم (سازوکار ۲) در مواردی که هر دو افزاره دارای واسط وروردی ساده هستند، مناسب است.

یک واسط وروردی یا خروجی استاندارد می‌تواند یک واسط ساده را تقلید کند، پس اگر هر دو افزاره دارای واسط‌های ورودی و خروجی استاندارد باشند، در این صورت هر یک از سازوکارها می‌توانند قابل استفاده باشند.

هر دو سازوکار فوق به طریق عمومی که معرفی می‌شود، عمل می‌کنند. یک رشته داده D از یک افزاره به افزاره دیگر (یا الحاق داده انتقال داده شده در هر دو جهت) توسط پیوند ارتباطات مشترک فرستاده می‌شود. حال، سازوکار احراز هویت هستار به صورت دستی اجرا می‌شود. در نتیجه‌ی این سازوکار، هر دو افزاره، مطمئن خواهند بود که رشته داده‌ای D، مشابه همان مقدار افزاره دیگر را دارد.

۶-۲ سازوکار ۱ـ یک افزاره با ورودی ساده، یک افزاره با خروجی ساده

۶-۲-۱ الزامات مشخص

این سازوکار دارای الزامات مشخص زیر است:

الف- سازوکار مشخص شده در این زیربند در جایی که یک افزاره (افزاره A) دارای واسط ورودی ساده و دیگری دارای واسط خروجی ساده است، مناسب خواهد بود.

ب- افزاره A دارای منابعی برای تولید کلیدها خواهد بود.

۶-۲-۲ مشخصات داده‌های تبادل شده

باید تبادلات داده‌ای و عملیات زیر انجام شود (مطابق شکل ۱):

الف- هر دو افزاره یک سیگنال خروجی به‌عنوان تصدیق دریافت داده D و آمادگی برای سازوکار احراز هویت، ارسال می‌کنند. در هنگام مشاهده آمادگی افزاره‌ها، کاربر می‌تواند یک سیگنال به افزاره A وارد کند و به A اطلاع دهد که می‌تواند سازوکار را شروع کند.

ب- افزاره A باید یک کلید تصادفی K را تولید کند به‌طوری که K برای استفاده در تابع مقدار ـ وارسی به اشتراک گذاشته شده توسط دو مؤلفه، مناسب است. با استفاده از این کلید K، افزاره A یک مقدار ـ وارسی به‌عنوان تابعی از داده D محاسبه می‌کند. مقدار ـ وارسی و کلید K خروجی افزاره A با استفاده از واسط خروجی خواهند بود. کاربر باید مقدار ـ وارسی و کلید K را از واسط خروجی بخواند.

پ- کاربر باید مقدار ـ وارسی و کلید K به‌عنوان خروجی افزاره A را با استفاده از واسط ورودی افزاره B به آن افزاره، وارد کند. افزاره B از کلید K برای محاسبه مجدد مقدار ـ وارسی به‌عنوان تابعی از نسخه ذخیره شده داده D استفاده می‌کند. اگر هر دو مقدار ـ وارسی‌ها سازگار باشند، در این صورت افزاره B با استفاده از واسط خروجی ساده خود، یک سیگنال موفقیت تولید می‌کند، در غیر این صورت یک سیگنال شکست تولید می‌کند.

ت- کاربر می‌تواند نتیجه خروجی افزاره B (شکست یا موفقیت) را با استفاده واسط ورودی ساده افزاره A وارد آن افزاره کند.

شکل ۱- سازوکار احراز هویت دستی ۱

۶-۲-۳ گواهی‌های احراز هویت دستی

سازوکار ۱ احراز هویت دارای این خاصیت است که هیچ اطلاعات تشخیص هویتی روی کانال ناامن تبادل نمی‌شود. بنابراین، اگر مقادیر احراز هویت دستی (K و مقدار ـ وارسی) قبل از دریافت داده واقعی D، از افزاره A به افزاره B منتقل شوند، امنیت سازوکار تحت تأثیر قرار نمی‌گیرد. طبیعتاً این رویکرد تنها در موقعیت‌هایی قابل استفاده است که افزاره A داده D را تولید کند. اما در چنین موردی، سازوکار ۱ ابزاری را برای احراز هویت داده که بعد از مدتی دریافت می‌شود را پیشنهاد می‌دهد. این ابزار تشخیص هویت، گواهی احراز هویت دستی نام دارد. حال یک پروتکل برای احراز هویت داده با استفاده از گواهی احراز هویت دستی مشخص می‌شود. (با همان الزامات مشابه که در زیربند ۶-۲-۱ بیان شد.)

باید خاطرنشان کرد که این پروتکل، احراز هویت هستار را مهیا نمی‌سازد.

فرض کنید افزاره A دارای داده D است و باید در چند لحظه بعد به افزاره B فرستاده شود.

الف- افزاره A کلید تصادفی K را تولید می‌کند که در آن K برای استفاده در تابع مقدار- وارسی به اشـتراک گذاشته شده توسط دو افزاره، مناسب است. با استفاده از این کلید K، افزاره A یک مقدار-

وارسی به‌عنوان یک تابع از داده D محاسبه می‌کند. مقدار- وارسی و کلید K، خروجی کاربر توسط واسط خروجی افزاره A خواهند بود. کاربر، خروجی مقدار- وارسی و کلید K را می خواند.

ب- کاربر، خروجی مقدار- وارسی و کلید K از افزاره A را وارد واسط ورودی افزاره B می‌کند. مقدار- وارسی و کلید K در افزاره B ذخیره می‌شوند.

ج- وقتی که افزاره B در مدتی بعد داده D را دریافت کند، می‌تواند صحت هویت داده‌ها را با استفاده از مقادیر ذخیره شده K و مقدار- وارسی، مورد ارزیابی قرار دهد. افزاره B با استفاده از کلید K، مقدار- وارسی را به‌عنوان یک تابع از داده دریافتی D، مجدداً محاسبه می‌کند. اگر دو مقدار- وارسی سازگار باشند، افزاره B داده‌ها را قبول می‌کند و یک سیگنال موفقیت به‌عنوان خروجی به کاربر می فرستد. در غیر این صورت یک سیگنال شکست می‌دهد.

گواهی احراز هویت دستی، متشکل از K و مقدار- وارسی محاسبه شده به‌عنوان تابعی از K و D است.

یادآوری- مثالی از یک داده که می‌تواند در D شامل شود، یک کلید عمومی برای یک افزاره، هویت آن، دامنه سرویس و غیره است. در پیوست ب مثالی از اینکه چگونه با استفاده از گواهی‌های احراز هویت می‌توان یک کلید سرّی مشترک بین دو افزاره ایجاد کرد، ارائه شده است.

۶-۳ سازوکار ۲ - افزاره‌های با قابلیت‌های ورودی ساده

۶-۳-۱ الزامات مشخص

این سازوکار دارای الزامات مشخص زیر است:

الف- سازوکار مشخص شده در این زیربند برای موردی که هر دو افزاره (A و B) دارای یک واسط ورودی ساده باشند، مناسب است.

ب- یکی از افزاره‌ها (افزاره دارای برچسب A در زیر) باید ابزاری برای تولید کلیدها داشته باشد.

۶-۳-۲ مشخصات داده‌های رد و بدل شده

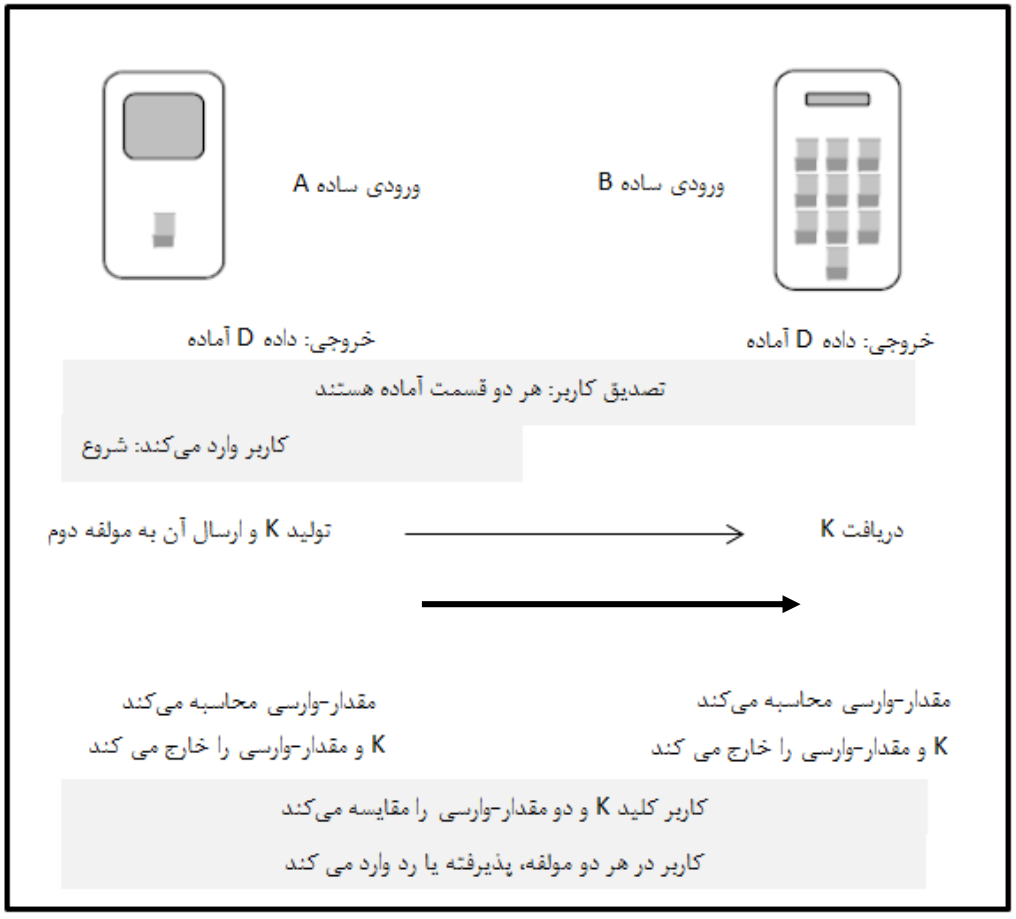
باید تبادل اطلاعات و عملیات زیر اتفاق بیافتد (مطابق شکل ۲):

الف- هر دو افزاره باید یک علامت برای اعلام این پیام داشته باشند که آنها داده‌های D را دریافت کرده‌اند و برای آغاز سازوکار احراز هویت آماده هستند. پس از مشاهده اینکه هر دو افزاره آماده هستند، کاربر باید یک علامت را برای افزاره A وارد کند تا به آن افزاره اعلام کند که سازوکار می‌تواند شروع شود.

ب- افزاره A باید یک کلید تصادفی K را تولید کند، که در آن K برای استفاده همراه با تابع مقدار وارسی مناسب است که توسط دو مؤلفه به اشتراک گذاشته شده است. با استفاده از این کلید K، افزاره A باید یک مقدار وارسی را به‌عنوان تابعی از داده‌ها D محاسبه کند. مقدار وارسی و کلید K باید در مرحله بعد از طریق واسط افزاره A تبدیل به خروجی شوند. افزاره A همچنین باید کلید K را برای افزاره B از طریق پیوند ارتباطی مشترک ارسال کند.

پ- افزاره B باید از کلید K برای محاسبه مقدار وارسی به‌عنوان تابعی از نسخه ذخیره شده خود از داده‌ها D استفاده کند، و کلید K و مقدار وارسی محاسبه شده را به‌عنوان خروجی نمایش دهد.

ت- کاربر باید دو مقدار وارسی و دو کلید خروجی را مقایسه کند. اگر دو مقدار باهم مطابقت داشته باشند، کاربر باید سیگنال تصدیق را برروی هردو افزاره وارد کند. اگر مقادیر وارسی یا مقادیر کلیدی متفاوت از هم باشند، پس سازوکار با شکست مواجه شده و کاربر باید یک سیگنال عدم تصدیق را برروی افزاره‌ها وارد کند. افزاره باید فقدان یک سیگنال تصدیق را به‌عنوان سیگنال شکست تفسیر کند. (این امر مستلزم پیاده‌سازی یک سازوکار زمان انقضا خواهد بود.)



شکل۲- سازوکار احراز هویت دستی ۲

۷ سازوکارهای به‌کارگیرنده یک انتقال دستی یک خلاصه- مقدار یا کلید کوتاه

۷-۱ کلیات

در این بند چهار سازوکار احراز هویت دستی تعیین می‌شوند که با انتقال دستی یک مقدار دارای خلاصه

کوتاه یا یک کلید کوتاه سروکار دارند. این چهار سازوکار برای انواع مختلف افزاره‌ها مناسب هستند. به‌طور خاص،

- دو سازوکار اول (سازوکارهای ۳ و ۴) برای شرایطی مناسب هستند که در آنها یک افزاره دارای یک واسط ورودی ساده و افزاره دیگر دارای یک واسط خروجی ساده است و
- دو سازوکار دوم (سازوکارهای ۵ و ۶) برای شرایطی مناسب هستند که در آنها هر دو افزاره دارای یک واسط ورودی ساده هستند.

یک واسط ورودی یا خروجی ساده می‌تواند با یک واسط ساده برابری کند، و از این‌رو هردو افزاره دارای واسط‌های ورودی و خروجی استاندارد هستند، پس هر کدام از سازوکارها قابل استفاده هستند.
تمام سازوکارها با این شیوه به‌کار گرفته می‌شوند. یک رشته داده‌ای D و یک مقدار درهم‌ساز از یک افزاره به افزاره دیگر از طریق پیوند ارتباطی مشترک منتقل می‌شوند. (D ممکن است به تناوب از الحاق داده‌های انتقال یافته در هردو افزاره شکل گیرد.) سازوکار احراز هویت هستار دستی در مرحله بعد اجرا می‌شود. به‌عنوان نتیجه‌ای از سازوکارها، برای هر دو ستگاه این اطمینان حاصل می‌شود که رشته داده‌ای D که آنها پردازش می‌کنند با مقدار محفوظ توسط افزاره دیگر، یکسان است.

۷-۲ سازوکار ۳- یک افزاره با ورودی ساده، یک افزاره با خروجی ساده

۷-۲-۱ الزامات مشخص

این سازوکار دارای الزامات مشخص زیر است:

الف- سازوکار معین در این زیربند برای شرایطی مناسب است که یک افزاره (افزاره A) دارای یک واسط ورودی ساده است و افزاره دیگر (افزاره B) دارای یک واسط خروجی ساده است.

ب- افزاره A باید دارای وسایلی برای تولید کلیدهای تصادفی (بلند) باشد.

۷-۲-۲ مشخصه داده‌های مبادله شده

تبادلات داده‌ای و عملیات زیر باید رخ دهند. (مطابق شکل ۳) توجه داشته باشید که مراحل الف- و ب- ممکن است همچون مراحل ت و ث به‌طور موازی رخ دهند.

الف- افزاره‌های A و B باید موقتاً برسر رشته داده‌ای D به توافق برسند. این مهم، برای مثال، از طریق تبادل پیام‌های محافظت نشده توسط پیوند ارتباطی مشترک، محقق می‌شود.

ب- افزاره A باید یک کلید تصادفی k را تولید و آن را به صورت سری حفظ کند، این کلید باید برای استفاده به‌عنوان کلیدی با تابع خلاصه مورد توافق d مناسب باشد. افزاره A باید $h(k)$ را محاسبه و این مقدار درهم‌ساز را برای افزاره B توسط برخی وسایل (امن بودن ضروری نیست.) ارسال کند، برای مثال از طریق پیوند ارتباطات مشترک.

پ- هردو افزاره باید یک سیگنال را از طریق واسط‌های خروجی خود برای اعلام این امر صادر کنند که مراحل الف و ب را تکمیل کرده‌اند و برای آغاز سازوکار احراز هویت آماده هستند. پس از مشاهده سیگنال‌ها، کاربر باید یک سیگنال را برای افزاره A ارسال کند، این کار از طریق واسط ورودی ساده خود برای اعلام امکان آغاز سازوکار به A انجام می‌شود.

ت- از طریق واسط خروجی استاندارد افزاره A، این افزاره باید یک مقدار - خلاصه کوتاه $d(D, k)$ را محاسبه کرده و در خروجی مقدار - خلاصه را نمایش دهد. کاربر باید مقدار - خلاصه کوتاه را از واسط خروجی استاندارد افزاره A خوانده، و آن را برای افزاره B با استفاده از واسط ورودی استاندارد متعلق به افزاره B، وارد کند.

ث- افزاره A باید کلید k را برای افزاره B از طریق پیوند ارتباطی مشترک ارسال کند. پس از دریافت k، افزاره B باید $h(k)$ را محاسبه و وارسی کند که آیا این مقدار با مقدار دریافت شده توسط افزاره B از افزاره A در مرحله ب برابر است یا خیر. اگر مقادیر درهم‌ساز با هم مطابقت داشتند، پس B به مرحله « و » می‌رود. در غیر این‌صورت، افزاره B باید یک سیگنال شکست را ارائه و سازوکاری را پیاده‌سازی کند که به موجب آن، آغاز یک نمونه جدید از سازوکار را برای یک دوره زمانی کوتاه مدت، نخواهد پذیرفت.

ج- افزاره B باید از کلید k و نسخه ذخیره شده آن از داده‌های D برای محاسبه مجدد مقدار - خلاصه کوتاه $d(D, k)$ استفاده کند. اگر مقدار - خلاصه برابر با مقدار دریافت شده توسط افزاره B در مرحله د باشد، پس افزاره B باید یک سیگنال موفقیت را برای کاربر از طریق واسط خروجی ساده خود صادر کند. در غیر این‌صورت، افزاره B باید یک سیگنال شکست را ارائه و سازوکاری را پیاده‌سازی کند که به موجب آن آغاز یک نمونه جدید از سازوکار را برای یک دوره زمانی کوتاه مدت، نخواهد پذیرفت.

چ- کاربر باید خروجی منتج توسط افزاره B، یعنی موفقیت یا شکست را برای افزاره A از طریق واسط ورودی ساده، وارد کند. افزاره A باید فقدان یک سیگنال تصدیق را به‌عنوان سیگنال شکست تفسیر کند. (این امر مستلزم پیاده‌سازی یک سازوکار زمان انقضا خواهد بود.)

یادآوری۱- سازوکار تأخیر زمانی مورد استفاده در مراحل ت و ج مانع از بروز حمله‌ی مردی- در- میان می‌شود که در آن یک حمله‌کننده سعی در تغییر ظاهر به‌عنوان افزاره A برای افزاره B دارد، البته بلافاصله پس از آنکه افزاره B اجرا را رها می‌کند، درحالی که افزاره A همچنان منتظر سیگنال تصدیق در مرحله چ است.

یادآوری۲- در این سازوکار افزاره B به افزاره A اعتماد می‌کند چراکه افزاره A کلید تصادفی k را تولید و در نتیجه مقدار خلاصه $d(D, k)$ را از پیش تعیین می‌کند. اگر افزاره B دارای توانایی تولید کلیدهای تصادفی باشد پس لزومی ندارد افزاره‌های A و B به یکدیگر اعتماد کنند. با این حال، این امر به‌طور بالقوه سازوکار را به دلیل افزایش ارتباط در شبکه و الزامات

همگام‌سازی، پیچیده‌تر می‌کند. این مشاهد برای سازوکارهای ۴ تا ۶ نیز اعمال می‌شود.

شکل ۳- سازوکار احراز هویت دستی ۳

۳-۷ سازوکار ۴- یک افزاره با ورودی ساده، یک افزاره با خروجی ساده

۱-۳-۷ الزامات مشخص

این سازوکار دارای الزامات مشخص زیر است:

الف- سازوکار تعیین شده در این زیربند برای شرایطی مناسب است که در آن یک افزاره (افزاره A) دارای یک واسط ورودی ساده و افزاره دیگر (افزاره B) دارای یک واسط خروجی ساده است.

ب- افزاره A باید دارای وسایلی برای تولید کلیدهای تصادفی (بلند) و جریان‌های بیتی کوتاه تصادفی باشد.

۷-۳-۲ مشخصه داده‌های مبادله شده

تبادلات و عملیات داده‌ای زیر باید رخ دهند. (مطابق شکل ۴) توجه داشته باشید که مراحل الف- و ب- ممکن است، همچون مراحل پ و ت به‌طور موازی رخ دهند.

الف- افزاره‌های A و B باید موقتاً بر سر رشته داده‌ای D به توافق برسند. این مهم، برای مثال، از طریق تبادل پیام‌های ارسالی محافظت نشده توسط پیوند داده‌ای مشترک، محقق می‌شود.

ب- افزاره A باید یک کلید تصادفی (بلند) k و یک رشته بیت تصادفی (کوتاه) R را تولید و آن‌ها را به‌صورت سرّی حفظ کند افزاره A باید $h(D||k||R)$ را محاسبه و این مقدار درهم ساز را برای افزاره B توسط برخی وسایل (امن بودن ضروری نیست.) ارسال کند، برای مثال از طریق پیوند ارتباطات مشترک.

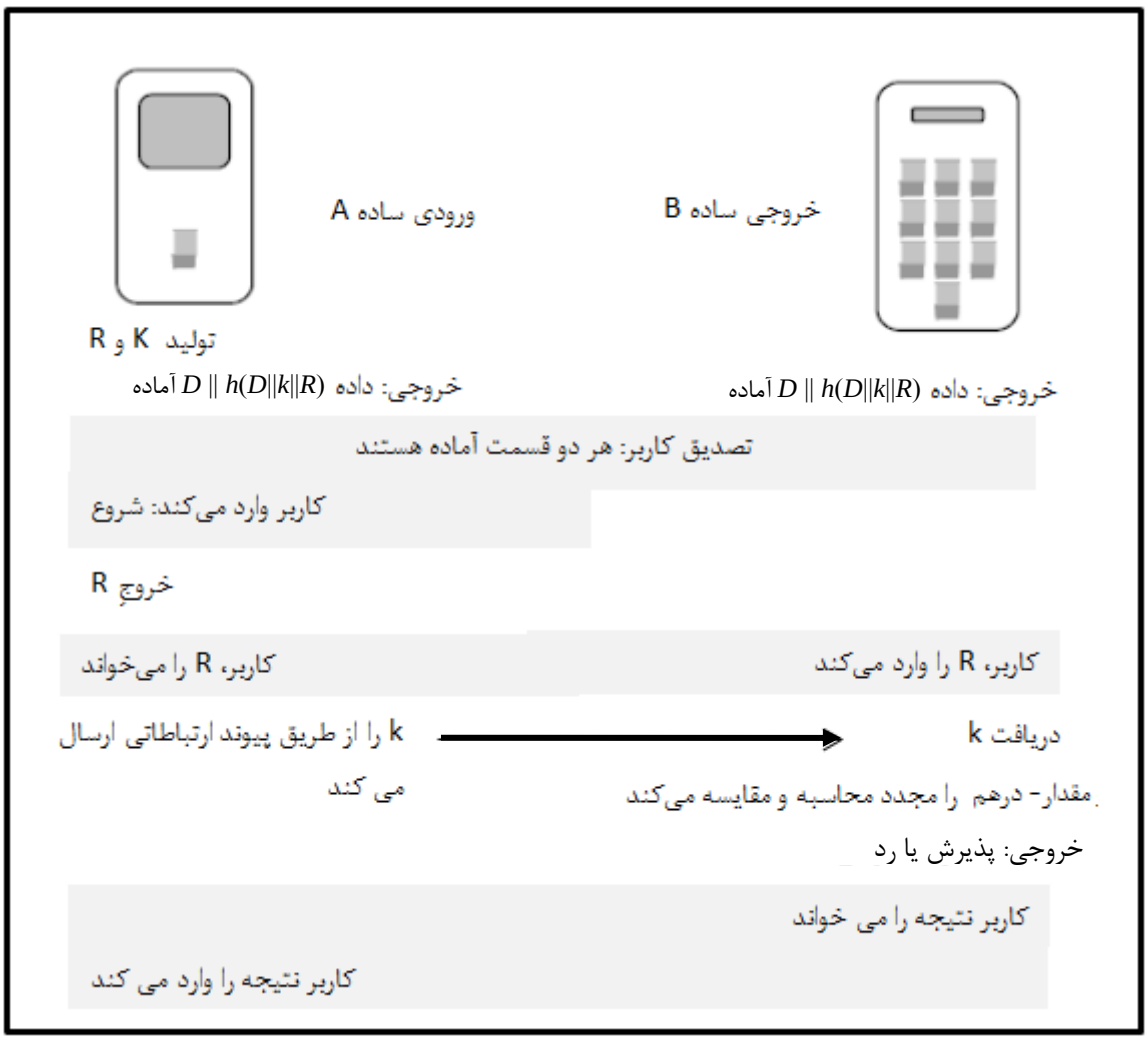
پ- هر دو افزاره باید یک سیگنال را از طریق واسط‌های خروجی خود برای اعلام این امر صادر کنند که مراحل الف و ب را تکمیل کرده‌اند و برای آغاز سازوکار احراز هویت آماده هستند. پس از مشاهده سیگنال‌ها، کاربر باید یک سیگنال را برای افزاره A ارسال کند، این کار از طریق واسط ورودی ساده خود برای اعلام امکان آغاز سازوکار به A انجام می‌شود.

ت- افزاره A باید در خروجی خود رشته بیت تصادفی کوتاه R را از طریق واسط استاندارد خروجی خود نمایش دهد. کاربر باید رشته بیت تصادفی کوتاه R را از واسط خروجی استاندارد افزاره A بخواند و آن را برای افزاره B با استفاده از واسط ورودی استاندارد افزاره B وارد نماید.

ث- افزاره A باید کلید k را برای افزاره B از طریق پیوند ارتباطی مشترک ارسال کند.

ج- پس از دریافت R و k در مراحل پ و ت، افزاره B باید از آنها برای محاسبه مجدد $h(D||k||R)$ به‌عنوان یکی از توابع نسخه ذخیره شده خود از داده‌های D استفاده کند. اگر مقدار درهم‌ساز برابر با مقدار دریافت شده توسط افزاره B از افزاره A در مرحله ب باشد، پس افزاره B باید یک سیگنال موفقیت را برای کاربر از طریق واسط خروجی ساده خود صادر کند. در غیر این‌صورت، افزاره B باید یک سیگنال شکست را ارائه دهد.

چ- کاربر باید خروجی منتج را توسط افزاره B، یعنی موفقیت یا شکست، برای افزاره A از طریق واسط ورودی ساده، وارد کند. افزاره A باید فقدان یک سیگنال تصدیق را به‌عنوان سیگنال شکست تفسیر کند. (این امر مستلزم پیاده‌سازی یک سازوکار زمان انقضا خواهد بود.)



شکل ۴- سازوکار احراز هویت دستی ۴

۴-۷ سازوکار ۵- سازوکارهای دارای قابلیت‌های ورودی خاص

۴-۷-۱ الزامات مشخص

این سازوکار دارای الزامات مشخص زیر است:

الف- سازوکار تعیین شده در این زیربند برای شرایطی مناسب است که در آن هر دو افزاره (A و B) دارای یک واسط ورودی ساده هستند.

ب- یکی از افزاره‌ها (افزاره دارای برچسب B در زیر) باید دارای وسایلی برای تولید کلیدهای تصادفی (بلند) باشد.

۷-۴-۲ مشخصه داده‌های مبادله شده

تبادلات و عملیات داده‌ای زیر باید رخ دهند. (مطابق شکل ۵) توجه داشته باشید که مراحل الف- و ب- ممکن است مانند مراحل پ و ت، به‌طور موازی رخ دهند.

الف- افزاره‌های A و B باید موقتاً بر سر رشته داده‌ای D به توافق برسند. این مهم، برای مثال، از طریق تبادل پیام‌های ارسالی محافظت نشده توسط پیوند داده‌ای مشترک محقق می‌شود.

ب- افزاره A باید یک کلید تصادفی k را تولید و آن را به صورت سرّی حفظ کند، این کلید باید برای استفاده به‌عنوان کلیدی با تابع خلاصه مورد توافق d مناسب باشد. افزاره A باید $h(k)$ را محاسبه و این مقدار درهم‌ساز را برای افزاره B توسط برخی وسایل (امن بودن ضروری نیست.) ارسال کند، برای مثال از طریق پیوند ارتباطات مشترک.

پ- هردو افزاره باید یک سیگنال را از طریق واسط‌های خروجی خود برای اعلام این امر صادر کنند که مراحل الف و ب را تکمیل کرده‌اند و برای آغاز سازوکار احراز هویت آماده هستند. پس از مشاهده سیگنال‌ها، کاربر باید یک سیگنال را برای افزاره A ارسال کند، این کار از طریق واسط ورودی استاندارد خود برای اعلام امکان آغاز سازوکار به A انجام می‌شود.

ت- افزاره A باید کلید k را برای افزاره B از طریق پیوند ارتباطی مشترک ارسال کند.

ث- افزاره A باید یک مقدار - خلاصه کوتاه $d(D, k)$ را محاسبه کرده و مقدار - خلاصه کوتاه را از طریق واسط خروجی استاندارد خود نمایش دهد.

ج- پس از دریافت کلید k در مرحله د، افزاره B باید مقدار درهم‌ساز $h(k)$ و مقدار - خلاصه کوتاه $d(D, k)$ را به‌عنوان تابعی از نسخه ذخیره شده‌ی خود از D مجدداً محاسبه کند. اگر مقدار درهم‌ساز برابر با مقدار دریافت شده توسط افزاره B از افزاره A در مرحله ب باشد، پس افزاره B باید مقدار - خلاصه کوتاه را از طریق واسط خروجی استاندارد خود صادر کند. در غیر این‌صورت، افزاره B باید یک سیگنال شکست را ارائه داده و سازوکاری را پیاده‌سازی کند که به موجب آن آغاز یک نمونه جدید از سازوکار را برای یک دوره زمانی کوتاه مدت نخواهد پذیرفت.

ح- کاربر باید دو مقدار - خلاصه کوتاه را از طریق واسط‌های خروجی استاندارد افزاره‌های A و B در مراحل ث و ج باهم مقایسه کند. اگر دو مقدار یکسان باشند، کاربر باید سیگنال تصدیق را برروی هردو افزاره، از طریق واسط‌های ورودی ساده آنها، وارد کند. در غیر این‌صورت، سازوکار با شکست مواجه شده و کاربر باید یک سیگنال عدم تصدیق را برروی افزاره‌ها وارد کند. افزاره‌ها باید فقدان یک سیگنال تصدیق را به‌عنوان سیگنال شکست تفسیر کنند؛ این امر مستلزم پیاده‌سازی یک سازوکار زمان انقضا خواهد بود.

یادآوری ۱ - سازوکار تأخیر زمانی مورد استفاده در مرحله « و » مانع از بروز حمله‌ی یک مرد-در-میان[1] می‌شود که در آن یک حمله‌کننده سعی در تغییر ظاهر به‌عنوان افزاره A برای افزاره B دارد البته بلافاصله پس از آنکه افزاره B اجرا را رها می‌کند، درحالی که افزاره A همچنان منتظر سیگنال تصدیق در مرحله « ه » است.

شکل ۵- سازوکار احراز هویت دستی ۵

۵-۷ سازکار ۶- افزاره‌هایی با قابلیت‌های ساده

۵-۷-۱ الزامات مشخص

این سازوکار دارای الزامات مشخص زیر است:

الف- سازوکار تعیین شده در این زیربند برای شرایطی مناسب است که در آن هر دو افزاره (A و B) دارای یک واسط ورودی ساده هستند.

ب- یکی از افزاره‌ها (افزاره دارای برچسب B در زیر) باید دارای وسایلی برای تولید کلیدهای تصادفی (بلند) و رشته - بیت‌های تصادفی کوتاه باشد.

---

1 - Man-in-the-middle

۷-۵-۲ مشخصه داده‌های مبادله شده

تبادلات و عملیات داده‌ای زیر باید رخ دهند. (مطابق با شکل ۶) توجه داشته باشید که مراحل الف- و ب- ممکن است همچون مراحل ت و ث، به‌طور موازی رخ دهند.

الف- افزاره‌های A و B باید موقتاً برسر رشته داده‌ای D به توافق برسند. این مهم، برای مثال، از طریق تبادل پیام‌های ارسالی محافظت نشده توسط پیوند داده‌ای مشترک محقق می‌شود.

ب- افزاره A باید یک کلید تصادفی k را تولید و آن را به‌صورت سرّی حفظ کند، این کلید باید برای استفاده به‌عنوان کلیدی با تابع خلاصه مورد توافق d مناسب باشد. افزاره A باید $h(k)$ را محاسبه و این مقدار درهم‌ساز را برای افزاره B توسط برخی وسایل (امن بودن ضروری نیست.) ارسال کند، برای مثال از طریق پیوند ارتباطات مشترک.

پ- هردو افزاره باید یک سیگنال را از طریق واسط‌های خروجی خود برای اعلام این امر صادر کنند که مراحل الف و ب را تکمیل کرده‌اند و برای آغاز سازوکار احراز هویت آماده هستند. پس از مشاهده سیگنال‌ها، کاربر باید یک سیگنال را برای افزاره A ارسال کند، این کار از طریق واسط ورودی ساده خود برای اعلام امکان آغاز سازوکار به A انجام می‌شود.

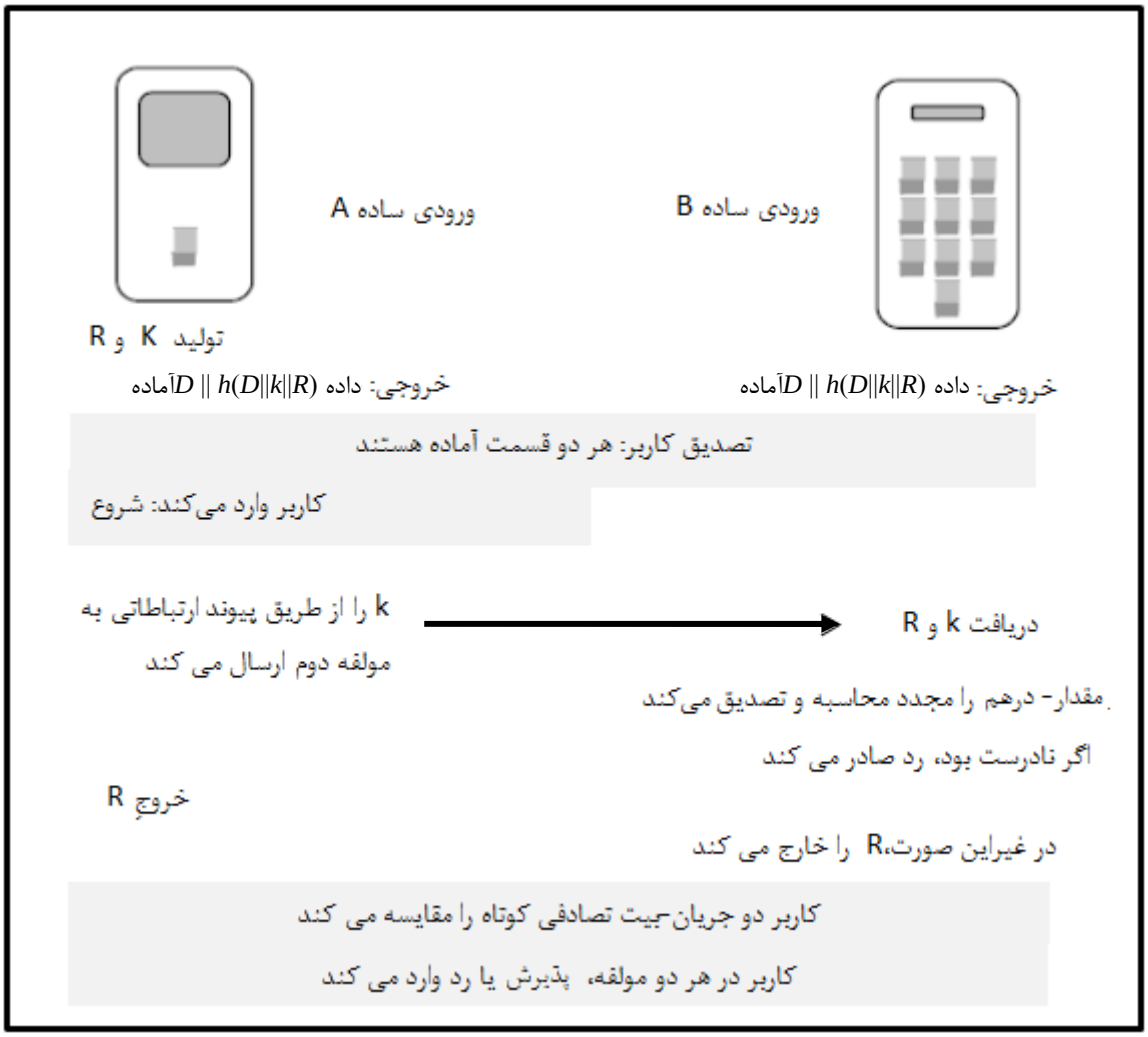
ت- افزاره A باید کلید k (بلند) و رشته - بیت تصادفی R (کوتاه) را از طریق پیوند ارتباطات مشترک به افزاره B ارسال کند.

ث- افزاره A باید در خروجی خود رشته بیتی تصادفی کوتاه R را از طریق واسط استاندارد خروجی خود نمایش دهد.

ج- در رسید k و R در گام پ افزاره B باید مقدار- درهم ساز $h(D||k||R)$ را به‌عنوان تابعی از نسخه ذخیره شده از داده D خود مجدداً محاسبه کند. اگر مقدار درهم‌ساز برابر با مقدار دریافت شده توسط افزاره B از افزاره A در مرحله ب باشد، پس افزاره B باید رشته بیتی تصادفی کوتاه R را از طریق واسط خروجی ساده خود صادر کند. در غیر این‌صورت، افزاره B باید یک سیگنال شکست را ارائه داده و سازوکاری را پیاده‌سازی کند که به موجب آن آغاز یک نمونه جدید از سازوکار را برای یک دوره زمانی کوتاه مدت نخواهد پذیرفت.

چ- کاربر باید دو رشته بیتی تصادفی کوتاه صادر شده از طریق واسط‌های خروجی استاندارد افزاره‌های A و B در مراحل ث و ج را باهم مقایسه کند. اگر دو مقدار یکسان باشند، کاربر باید سیگنال تصدیق را برروی هردو افزاره، از طریق واسط‌های ورودی ساده آنها، وارد کند. در غیر این‌صورت، سازوکار با شکست مواجه شده و کاربر باید یک سیگنال عدم تصدیق را برروی افزاره‌ها وارد کند. افزاره‌ها باید فقدان یک سیگنال تصدیق را به‌عنوان سیگنال شکست تفسیر کنند؛ این امر مستلزم پیاده‌سازی یک سازوکار زمان انقضا خواهد بود.

یادآوری- سازوکار تأخیر زمانی مورد استفاده در مراحل « ن » و « و » مانع از بروز حمله‌ی یک مرد-در-وسط می‌شود که در آن یک حمله‌کننده سعی در تغییر ظاهر به‌عنوان افزاره A برای افزاره B دارد، البته بلافاصله پس از آنکه افزاره B اجرا را رها می کند، درحالی که افزاره A همچنان منتظر سیگنال تصدیق در مرحله « ه » است.



شکل ۶- سازوکار احراز هویت دستی ۶

۸ سازوکارهای استفاده کننده از یک MAC

۸-۱ کلیات

در این بند دو مورد از سازوکارهای احراز هویت دستی مبتنی بر استفاده از کد احراز هویت پیام (MAC) مشخص می‌شوند. این دو سازوکار برای انواع مختلف افزاره‌ها مناسب هستند، به‌صورت خاص:

- سازوکار اول (سازوکار ۷) در مواردی که هر دو افزاره دارای واسط خروجی ساده هستند، مناسب است و
- سازوکار دوم (سازوکار ۸) در مواردی که یک افزاره دارای واسط ورودی ساده و افزاره دیگر دارای واسط خروجی ساده، مناسب است.

یک واسط ورودی یا خروجی استاندارد می‌تواند یک واسط ساده را تقلید کند، پس اگر هر دو افزاره دارای واسط‌های ورودی و خروجی استاندارد باشند در این صورت هر یک از سازوکارها می‌تواند قابل استفاده باشد.
هر دو سازوکار به طریق عمومی که معرفی می‌شوند، عمل می‌کنند. یک رشته داده D از یک افزاره به افزاره دیگر (یا الحاق داده انتقال داده شده در هر دو جهت) توسط پیوند ارتباطات مشترک فرستاده می‌شود. حال، سازوکار احراز هویت هستار به صورت دستی اجرا می‌شود. در نتیجهٔ این سازوکار، هر دو افزاره، مطمئن خواهند بود که رشته داده‌ای D، مشابه همان مقدار افزاره دیگر را دارد.

۸-۲ سازوکار ۷- افزاره‌های دارای قابلیت‌های ساده خروجی

۸-۲-۱ کلیات

این سازوکار دارای دو گونه (۷ الف و ۷ ب) است. سازوکار ۷ الف، تعیین شده در بند ۸-۲-۳، نیازمند تعاملات کمتری بین دو افزاره است، در حالی که ۷ب، تعیین شده در قسمت ۸-۲-۴، نیازمند تعامل کمتر کاربر به صورت دستی است.

۸-۲-۲ الزامات مشخص

این سازوکار دارای الزامات مشخص زیر است:

الف- دو گونه سازوکار تعیین شده در این زیربند برای شرایطی مناسب هستند که هر دو افزاره (A و B) دارای یک واسط خروجی ساده باشند.

ب- هر دو افزاره باید دارای وسایلی برای تولید کلیدهای تصادفی MAC باشند، و کاربر باید دارای وسایلی برای تولید رشته بیت‌های تصادفی کوتاه باشد.

پ- قبل از شروع سازوکار، هر دو افزاره باید از هویت هم‌دیگر آگاه باشند.

یادآوری- اگر کاربر انتخاب‌های ضعیفی را برای رشته بیت تصادفی انجام دهد، برای مثال کاربر همیشه مقدار یکسانی را انتخاب می‌کند، پس احتمال خطر یک حمله موفق بر سازوکار به شدت افزایش می‌یابد.

۸-۲-۳ مشخصه داده‌های مبادله شونده در سازوکار ۷ الف

تبادلات و عملیات زیر باید رخ دهند(مطابق شکل ۷). توجه داشته باشید که مراحل ب و پ ممکن است، همچون مراحل ت، ث و ج، چ به‌صورت موازی رخ دهند.

الف- هر دو افزاره باید از طریق واسط‌های خروجی ساده‌ی مربوطه‌ی خود، دارای خروجی باشند، در واقع یک سیگنال برای اعلام این امر که آنها داده‌های D را دریافت کرده‌اند و برای آغاز سازوکار احراز هویت آماده هستند. پس از مشاهده اعلام آمادگی هر دو افزاره، کاربر باید یک رشته بیتی کوتاه تصادفی R را تولید کند. کاربر باید یک رشته بیت تصادفی R را وارد هر دو افزاره کند و سپس یک سیگنال را برای افزاره A وارد نموده تا به A اعلام کند که سازوکار می‌تواند شروع شود.

ب- افزاره A باید یک کلید تصادفی K_A را تولید کند که برای استفاده به‌عنوان یک کلید دارای تابع MAC مشترک بین دو افزاره مناسب است. با استفاده از K_A به‌عنوان کلید، افزاره A یک MAC (با

نشان MAC_A) را برروی آن رشته داده‌ای محاسبه می‌کند که متشکل از الحاق I_A (یک شناساگر برای A)، داده‌ها D و رشته بیت تصادفی R است. افزاره A باید MAC_A را برای افزاره B از طریق پیوند ارتباطی مشترک ارسال کند.

پ- افزاره B باید یک کلید تصادفی K_B را تولید کند که برای استفاده به‌عنوان یک کلید دارای تابع MAC مشترک بین دو افزاره مناسب است. با استفاده از K_B به‌عنوان کلید، افزاره B یک MAC (با نشان MAC_B) را برروی آن رشته داده‌ای محاسبه می‌کند که متشکل از الحاق I_B (یک شناساگر برای B)، داده‌ها D، و رشته بیت تصادفی R است. افزاره B باید MAC_B را برای افزاره A از طریق پیوند ارتباطی مشترک ارسال کند.

ت- وقتی افزاره A، MAC_B را دریافت کرده است (نه قبل از آن)، افزاره A باید K_A را برای افزاره B ارسال کند.

ث- پس از دریافت K_A، افزاره B تائید می‌کند که MAC_A برابر با یک مقدار MAC محاسبه شده با استفاده از مقادیر ذخیره شده‌ی R، D، I_A و مقدار دریافت شده K_A به‌عنوان کلید است. اگر رویه تائید به‌طور موفقیت‌آمیز انجام شود، افزاره B سیگنال موفقیت را در خروجی نمایش می‌دهد.

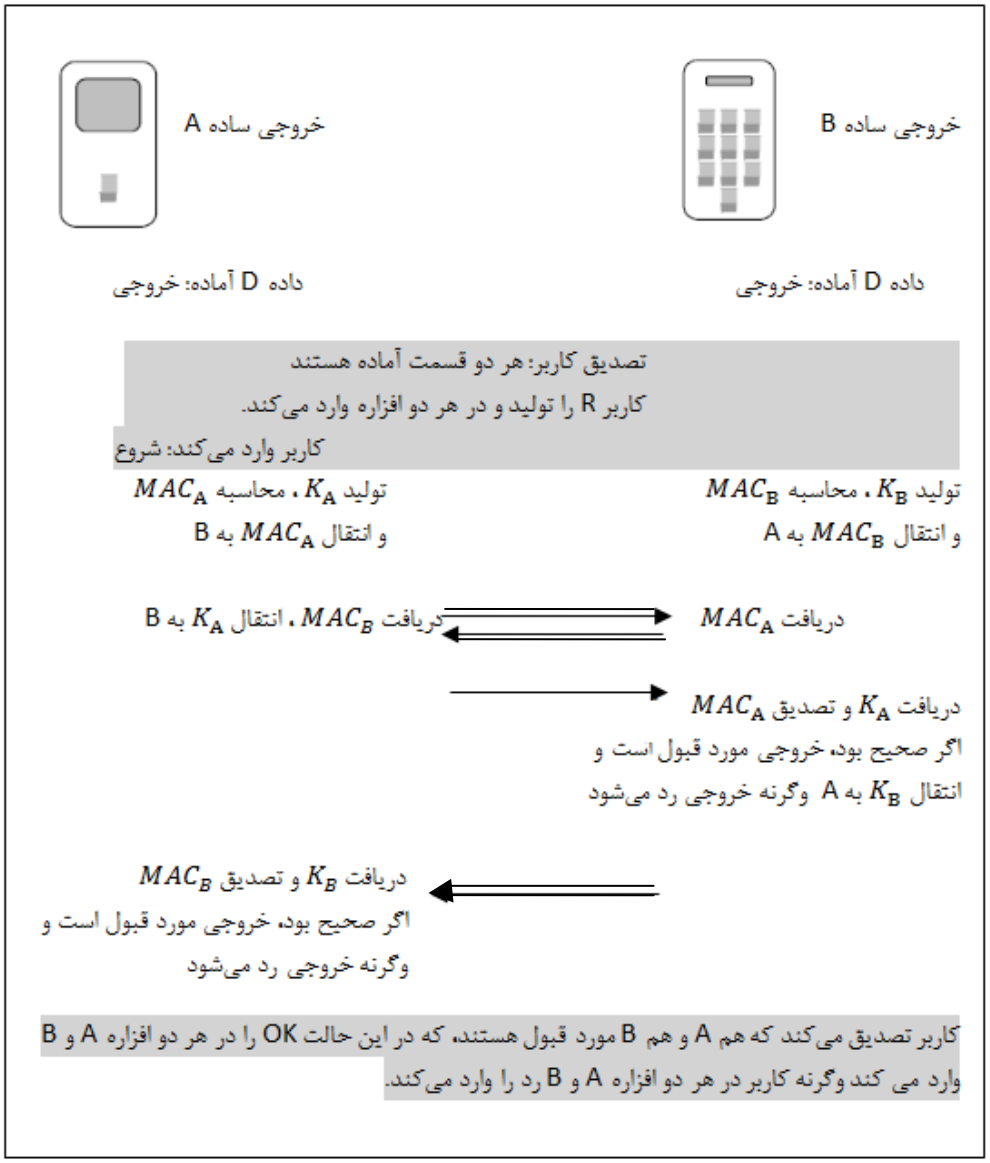
ج- وقتی افزاره B، MAC_A را دریافت کرده است (نه قبل از آن)، افزاره B باید K_B را برای افزاره A ارسال کند.

چ- پس از دریافت K_B، افزاره A تائید می‌کند که MAC_B برابر با یک مقدار MAC محاسبه شده با استفاده از مقادیر ذخیره شده‌ی R، D، I_B، و مقدار دریافت شده K_B به‌عنوان کلید است. اگر رویه تائید به‌طور موفقیت‌آمیز انجام شود، افزاره A سیگنال موفقیت را در خروجی نمایش می‌دهد.

ح- کاربر تائید می‌کند که هر دو افزاره یک سیگنال موفقیت ارائه داده‌اند و در این‌صورت، تاییدیه موفقیت را برای هر دو افزاره وارد می‌کند. اگر یکی از افزاره‌ها یا هردوی آنها سیگنال شکست را ارائه دهند، پس کاربر باید یک سیگنال شکست را برروی هر دو افزاره وارد کند. اگر کاربر در ورود یک سیگنال موفقیت برروی یک افزاره طی فاصله زمانی مشخص با شکست مواجه شود، این امر به‌عنوان شکست سازوکار باید تفسیر شود.

یادآوری- مرحله چ در این سازوکار مانع از بروز حمله جایگزینی[1] می‌شود که در آن حمله‌کننده سعی در جعل هویت به‌عنوان افزاره A برای افزاره B را دارد.

1 - Substitution attack



شکل ۷- سازوکار احراز هویت دستی ۷الف

۸-۲-۴ مشخصه داده‌های مبادله شونده در سازوکار ۷ب

تبادلات و عملیات زیر باید رخ دهند (مطابق شکل ۸).

الف- هر دو افزاره باید برای اعلام این امر دارای خروجی باشند که آنها داده‌های D را دریافت کرده‌اند و برای آغاز سازوکار احراز هویت آماده هستند. پس از مشاهده اعلام آمادگی هر دو افزاره، کاربر باید یک رشته بیتی تصادفی $R = (r_1,r_2,\ldots,r_n)$ را تولید کند که در آن r_i یک بیت و n تعداد بیت‌ها در R است. کاربر باید مقدار R را وارد هر دو افزاره کند و سپس یک سیگنال را برای افزاره A وارد نموده تا به A اعلام کند که سازوکار می‌تواند شروع شود.

ب- برای i که به‌طور متوالی با مقادیر 1,2,…,n مقداردهی شده، باید گام‌های ۱ الی ۵ اجرا شوند. (توجه داشته باشید که گام‌های ۱ و ۲ ممکن است به‌طور موازی اجرا شوند.)

۱- افزاره A باید یک کلید تصادفی K_{Ai} را تولید کند که برای استفاده به‌عنوان یک کلید دارای

تابع MAC مشترک بین دو افزاره مناسب است. با استفاده از K_{Ai} به‌عنوان کلید، افزاره A یک MAC (با نشان MAC_{Ai}) را بر روی آن رشته داده‌ای محاسبه می‌کند که متشکل از الحاق I_A (یک شناساگر برای A)، داده‌ها D و رشته بیت تصادفی ri است. افزاره A باید MAC_{Ai} را برای افزاره B از طریق پیوند ارتباطی مشترک ارسال کند.

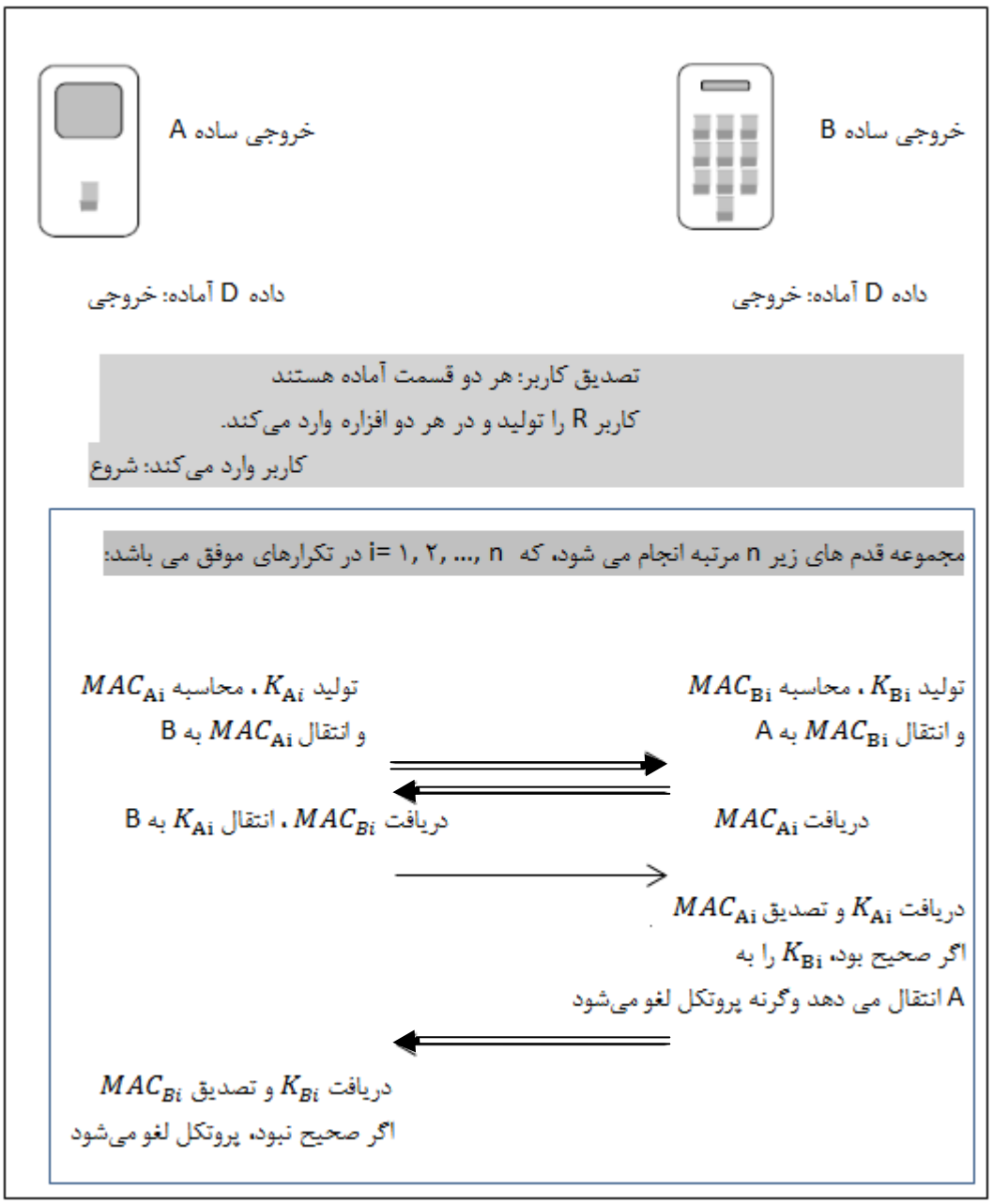
۲- افزاره B باید یک کلید تصادفی K_{Bi} را تولید کند که برای استفاده به‌عنوان یک کلید دارای تابع MAC مشترک بین دو افزاره مناسب است. با استفاده از K_{Bi} به‌عنوان کلید، افزاره B یک MAC (با نشان MAC_{Bi}) را بر روی آن رشته داده‌ای محاسبه می‌کند که متشکل از الحاق I_B (یک شناساگر برای B)، داده‌ها D، و رشته بیت تصادفی ri است. افزاره B باید MAC_{Bi} را برای افزاره A از طریق پیوند ارتباطی مشترک ارسال کند.

۳- پس از دریافت MAC_{Bi}، افزاره A همان K_{Ai} را برای افزاره B ارسال می‌کند.

۴- پس از دریافت MAC_{Ai} و K_{Ai} ، افزاره B تائید می‌کند که MAC_{Ai} برابر با یک مقدار MAC محاسبه شده با استفاده از مقادیر ذخیره شده‌ی ri، D، I_A، و کلید دریافت شده K_{Ai} است. اگر رویه تائید به‌طور موفقیت‌آمیز انجام شود، افزاره B همان K_{Bi} را برای افزاره A ارسال می‌کند؛ در غیر این‌صورت پروتکل نیمه‌تمام باقی می‌ماند.

۵- پس از دریافت K_{Bi} ، افزاره A تائید می‌کند که MAC_{Bi} برابر با یک مقدار MAC محاسبه شده با استفاده از مقادیر ذخیره شده‌ی ri، D، I_B و کلید دریافت شده K_{Bi} است. اگر رویه تائید به‌طور موفقیت‌آمیز انجام نشود، افزاره A پروتکل را نیمه‌تمام باقی می‌گذارد.

یادآوری- در صورتی که $i = n$، اگر رویه تائید در مراحل ۴ و ۵ موفقیت‌آمیز باشد، افزاره‌های A و B، به ترتیب می‌توانند یک سیگنال موفقیت را در خروجی نمایش دهند. در حالی که این امر از اجزا لاینفک پروتکل نیست ، اما می‌تواند برای دادن این سیگنال به کاربر افزاره مفید باشد که فرآیند به‌طور موفقیت‌آمیزی تکمیل شده است.



شکل ۸- سازوکار احراز هویت دستی ۷ب

۸-۳ سازوکار ۸- یک افزاره با ورودی ساده، یک افزاره با خروجی ساده

۸-۳-۱ کلیات

این سازوکار نیز دارای دو گونه (۸ الف و ۸ ب) است. سازوکار ۸الف، نیازمند تعاملات کمتری بین دو افزاره است ، در حالیکه ۸ب نیازمند تعامل کمتر کاربر به‌صورت دستی است.

۸-۳-۲ الزامات مشخص

این سازوکار دارای الزامات مشخص زیر است:

الف- دو متغیر سازوکار تعیین شده در این زیربند برای شرایطی مناسب هستند که در آن افزاره یک (A)

دارای یک واسط ورودی ساده و افزاره دیگر (B) دارای یک واسط خروجی ساده است.

ب- هردو افزاره باید دارای وسایلی برای تولید کلیدهای تصادفی MAC باشند.

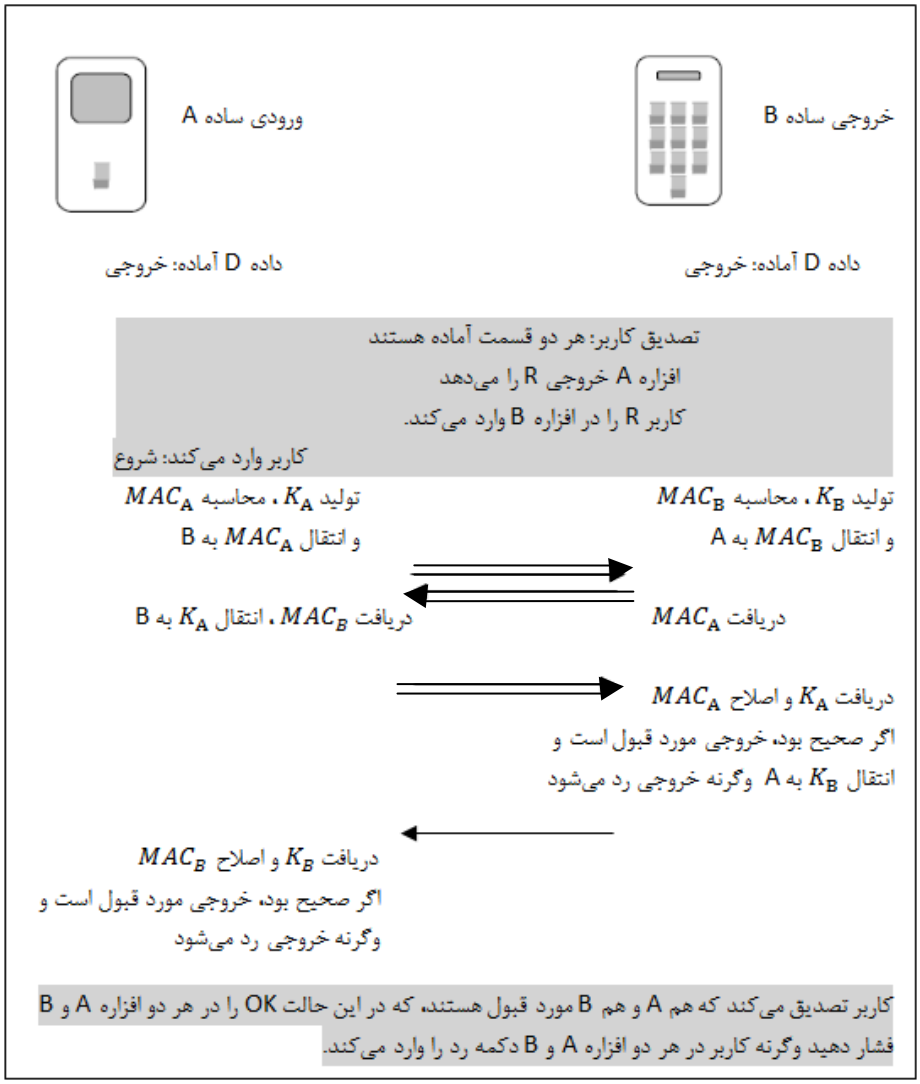
پ- قبل از شروع سازوکار، افزاره‌ها از شناساگر همدیگر آگاه هستند.

۸-۳-۳مشخصه داده تبدیل شده در سازوکار ۸الف

عملیات و تبدیلات داده دقیقاً مشابه سازوکار الف۷ است (همان‌طور که در بند ۸-۲-۳ شرح داده شده است.) فقط با این استثناء که:

ـ در گام الف) افزاره ارتباطی A رشته ـ بیت تصادفی را تولید و آن را به کاربر نشان می‌دهد که کاربر آن را به افزاره ارتباطی B کپی می‌کند. بنابراین در این سازوکار نیازی نیست که کاربر رشته ـ بیت تصادفی را تولید کند.

این سازوکار در شکل ۹ نمایش داده شده است.



شکل ۹- سازوکار احراز هویت دستی ۸الف

۸-۳-۴ مشخصه داده تبدیل شده در سازوکار ۸ب

عملیات و تبدیلات داده دقیقاً مشابه سازوکار ۷ب است (همان‌طور که در بند ۸-۲-۴ شرح داده شده است.) فقط با این استثنا که:

- در گام الف- افزاره ارتباطی A رشته ـ بیت تصادفی را تولید و آن را به کاربر نشان می‌دهد که کاربر آن را به افزاره B کپی می‌کند. بنابراین در این سازوکار نیازی نیست که کاربر رشته ـ بیت تصادفی را تولید کند.

پیوست الف

(اطلاعاتی)

پیمانه ASN.1

الف-۱ تعریف صوری

```
EntityAuthenticationMechanisms-6 {
   iso(1) standard(0) e-auth-mechanisms(9798) part6(6)
      asn1-module(0) object-identifiers(0) }
   DEFINITIONS EXPLICIT TAGS ::= BEGIN

-- EXPORTS All; --
-- IMPORTS None; --

OID ::= OBJECT IDENTIFIER -- alias
-- Synonyms --
is9798-6 OID ::= { iso(1) standard(0) e-auth-mechanisms(9798)
part6(6) }
mechanism OID ::= { is9798-6 mechanisms(1) }
-- Mechanisms using manual transfer of a short key and a short
check-value --
mdt-kc-siso OID ::= {mechanism mdt-kc-siso(1)}
mdt-kc-sisi OID ::= {mechanism mdt-kc-sisi(2)}

-- Mechanisms using manual transfer of a short digest-value or
a short key --
mdt-c-siso-one OID ::= {mechanism mdt-c-sisoone(3)}
mdt-c-siso-two OID ::= {mechanism mdt-c-sisotwo(4)}
mdt-c-sisi-one OID ::= {mechanism mdt-c-sisione(5)}
mdt-c-sisi-two OID ::= {mechanism mdt-c-sisitwo(6)}
-- Mechanisms using a MAC --
mac-k-soso OID ::= {mechanism mac-k-soso(7)}
mac-k-siso OID ::= {mechanism mac-k-siso(8)}
END -- EntityAuthenticationMechanisms-6 --
```

پیوست ب

(اطلاعاتی)

استفاده از پروتکل‌های احراز هویت دستی برای تبادل کلیدهای سرّی

ب-۱ کلیات

در این پیوست روش‌هایی را برای توانمند کردن وسایل ارتباطی برای به اشتراک‌گذاری یک کلید سرّی با استفاده از سازوکارهای احراز هویت دستی که در متن این استاندارد شرح داده شد.

ب-۲ توافق کلید احراز هویت شده Diffie-Hallman

رویه‌ی که در اینجا توضیح داده می‌شود، سازوکار توافق کلید احراز هویت شده Diffie-Hallman است که با استفاده از سازوکار احراز هویت دستی، احراز هویت می‌شود. سازوکار توافق کلید مطابق سازوکار توافق کلید ۴ موجود در استاندارد ملی ۳-۱۰۸۲۲ است (همچنین به پیوست ب-۵ استاندارد ملی ۳-۱۰۸۲۲ مراجعه شود). توصیف مفروض زیر، ساده‌سازی شده و برای توضیحات کامل، خواننده به استاندارد ملی ۳-۱۰۸۲۲ ارجاع داده می‌شود.

سازوکار توافق کلید احراز هویت شده Diffie-Hallman در قالب عمومی گروه G توضیح داده می‌شود (در اصطلاح شناسی ضربی بیان شده است.) و عنصر g در G که به اندازه کافی بزرگ است. گام‌های این رویه به قرار زیر است:

الف- افزاره A به‌صورت تصادفی و خصوصی (مخفیانه) عدد صحیح x را تولید می‌کند، g^x را محاسبه و آن را به افزاره B ارسال می‌کند.

ب- افزاره B به‌صورت تصادفی و خصوصی (مخفیانه) عدد صحیح y را تولید می‌کند، g^y را محاسبه و آن را به افزاره A ارسال می‌کند.

پ- افزاره A و B یکی از پروتکل‌های احراز هویت دستی را برای داده $D = (g^x \| g^y \| text)$ اجرا می‌کنند که در آن «متن» هر داده اضافی است. به‌عنوان مثال دیگر، احراز هویت دهنده‌ها، که وسایل ارتباطی بخواهند روی آن توافق کنند.

ت- اگر نتیجه پروتکل احراز هویت دستی موفقیت‌آمیز باشد، مؤلفه‌ها می‌توانند کلید Diffie-Hallman به اشتراک‌گذاری شده یعنی $S = g^{xy}$ را محاسبه کنند.

مؤلفه‌ها سپس می‌توانند کلیدهای رمزنگاشتی سرّی را با طول و فرمت مورد نیاز از کلید خصوصی Diffie-Hallman S به اشتراک‌گذاری شده استنتاج کنند.

ب-۳ کلید توافق احراز هویت شده با استفاده از گواهی احراز هویت دستی

ب-۳-۱ کلیات

رویه‌ی که در اینجا شرح داده می‌شود سازوکار کلید توافق Diffie-Hallman است، که در آن یکی از

کلیدهای عمومی Diffie-Hallman با استفاده از گواهی احراز هویت دستی احراز هویت می‌شود. (از این‌رو، این رویه به سازوکار ۱ اختصاص دارد و الزامات ذکر شده در زیربند ۶-۲-۱ باید برآورده شود.) افزاره B با استفاده از نسخه رمزبندی شده کلید مقدار- وارسی K مورد استفاده در سازوکار، به افزاره A احراز هویت می‌شود. به یاد داشته باشید این سازوکار نیازمند ۲ افزاره است تا بتواند یک سازوکار رمزبندی متقارن e را توافق و پیاده‌سازی کند که در آن $e_L(M)$ به معنی رمزبندی داده M با استفاده از کلید سرّی L است. فنون رمزبندی متقارن در استانداردهای ISO/IEC 18033-3 و ISO/IEC 18033 -4 استاندارد شده‌اند.

سازوکار توافق کلید Diffie-Hallman در قالب عمومی گروه G توضیح داده می‌شود (در اصطلاح شناسی multiplicative بیان شده است.) و یک المان g در G، به اندازه کافی بزرگ است. رویه ۲ مرحله دارد: مرحله ۱ و مرحله ۲.

در مرحله ۱، افزاره A کلید خصوصی Diffie-Hallman خود را تولید می‌کند، کلید عمومی متناظر را محاسبه می‌کند و یک گواهی احراز هویت دستی برای رشته داده‌ای شامل این کلید عمومی تولید می‌کند. گواهی به افزاره B انتقال داده می‌شود. در مرحله ۲، افزاره B کلید عمومی افزاره A را دریافت و مورد درستی آزمایی قرار می‌دهد و کلیدهای Diffie-Hallman مشترک و سرّی خود را تولید می‌کند. علاوه بر این، هر دو افزاره، راز Diffie-Hallman مشترک که از آن یک کلید رمزبندی سرّی استنتاج می‌شود را محاسبه می‌کنند. در نهایت، افزاره A نسخه رمزبندی‌شده کلید K که از افزاره B دریافت کرده است را به‌عنوان بخشی از فرآیند احراز هویت دستی، تایید می‌کند. به‌عنوان نتیجه، راز Diffie-Hallman به اشتراک گذاشته شده توسط دو افزاره، احراز هویت شده است.

ب-۳-۲ مرحله ۱

الف- افزاره A به‌صورت تصادفی و خصوصی (مخفیانه) یک عدد صحیح x را تولید و g^x را محاسبه می‌کند. بعد از آن، افزاره A یک گواهی احراز هویت دستی برروی رشته داده‌ای D متشکل از g^x و هر داده دیگر که نیاز است به‌صورت مطمئن به افزاره B انتقال یابد، ایجاد می‌کند. گواهی احراز هویت دستی (K، مقدار- وارسی) به‌صورت دستی به افزاره B منتقل می‌شود و افزاره B آن را ذخیره می‌کند. افزاره A، x و g^x و هر اقلام داده که درون D شامل می‌شود را ذخیره می‌کند.

ب-۳-۳ مرحله ۲ (آغاز شده توسط هر کدام از افزاره‌ها در مدتی بعد)

ب_ افزاره A، g^x و احتمالاً برخی از داده‌های دیگر را از طریق پیوند ارتباطات مشترک به افزاره B می‌فرستد. افزاره B احراز هویت g^x را بر مبنای گواهی احراز هویت دستی ذخیره شده، تایید می‌کند.

پ_ افزاره B به‌صورت تصادفی و خصوصی (مخفیانه) یک عدد صحیح y را تولید و g^y را محاسبه می‌کند. افزاره B راز مشترک Diffie-Hallman مانند $S = (g^x)^y$ را محاسبه می‌کند و S را برای رمز بندی کلید K، برای مثال، کلید گواهی احراز هویت دستی، استفاده می‌کند. افزاره B، کلید رمز بندی شده $e_L(K)$ و کلید عمومی Diffie-Hallman خودش، g^y را به افزاره A می‌فرستد.

ت_ افزاره A کپی راز مشترک مانند $S = (g^y)^x$ را محاسبه می‌کند. سپس $e_s(K)$ را رمزگشایی و درستی K را تایید می‌کند. اگر چنین باشد، افزاره A می‌تواند S را به‌عنوان احراز هویت شده قبول کند.

پس از آن، دو افزاره می‌توانند از کلید Diffie-Hallman سرّی مشترک، کلیدهای رمزگشایی با طول و فرمت مورد نیاز را استنتاج کنند.

یادآوری – در توصیف بالا، کلید K از گواهی احراز هویت دستی، در قدم‌های پ و ت مورد استفاده قرار گرفت. هر مقدار مناسب دیگر، به‌عنوان مثال، مقدار- وارسی یا برخی از مقدارهای خاص منظوره که بین طرف‌های مرحله ۱ به توافق رسیده است، می‌تواند مورد استفاده قرار گیرد.

ب-۴ بیش از دو مولفه

افزاره‌ای که از طریق آن بیش از دو افزاره می‌تواند بر روی یک کلید سرّی با استفاده از فنون احراز هویت دستی توافق کنند، در حال حاضر شرح داده می‌شود.

الف- یک افزاره به‌عنوان افزاره «ارشد» عمل می‌کند.

ب- افزاره ارشد با هر افزاره دیگری برای ایجاد یک کلید سرّی مشترک، سازوکار تشریح شده مطابق ب-۲ را اجرا می‌کند.

پ- سپس هر یک از افزاره‌ها می‌تواند کلید سرّی مشترک را تولید کند که پس از آن توزیع‌های رمز بندی شده و همچنین، احتمالاً برای باقی‌مانده افزاره‌ها از طریق افزاره ارشد، یکپارچگی-محافظت‌شده[1] خواهد بود.

اگرتعداد افزاره‌ها n باشد، افزاره ارشد نیاز به محاسبه n مورد exponentiation در گروه G را دارد. هر یک از افزاره‌های دیگر نیاز به محاسبه ۲ مورد exponentiation را دارد. بنابراین بهتر است افزاره ارشدی انتخاب شود که قدرت محاسباتی کافی برای انجام کار را داشته باشد.

1- integrity-protected

پیوست پ
(اطلاعاتی)
استفاده از پروتکل احراز هویت دستی برای تبادل کلیدهای عمومی

پ-۱ کلیات

در این پیوست روش‌هایی را برای توانمند کردن وسایل ارتباطی برای تبدیل مطمئن یک کلید عمومی با استفاده از سازوکارهای احراز هویت دستی که در متن اصلی این استاندارد تشریح شد، شرح داده می‌شود. زمینه تفسیر بین مرجع گواهینامه (CA) و یک کارخواه CA است. مرجع گواهینامه، نیاز دارد کلید عمومی خود را به‌طور مطمئن به کارخواه انتقال دهد و کارخواه کلید عمومی خود را به‌طور مطمئن به CA انتقال دهد. بسته به اینکه آیا کارخواه CA، کلیدهای خصوصی خود را تولید می‌کند یا آن‌ها از طریق تسهیل مدیریت کلید[1] تولید شده و سپس وارد افزاره می‌شوند، دو مورد مختلف شرح داده می‌شوند.

پ-۲ الزامات

مرجع گواهینامه باید به واسط خروجی استاندارد، برای مثال یک صفحه نمایش و یک واسط ورودی ساده برای ورود دستورات آن مجهز شود. کارخواه CA باید دارای یک واسط ورودی استاندارد و یک واسط خروجی استاندارد، برای مثال یک خروجی صوتی برای نمایش موفقیت یا عدم موفقیت فرایند، باشد.

یادآوری – برای مواردی که CA و کارخواه CA دارای انواع مختلفی از واسط‌های کاربری باشند، سازگاری رویه کار سرراستی است و فقط، نوع پروتکل احراز هویت دستی نیاز به تغییر دارد.

پ-۳ کلید خصوصی تولید شده در افزاره

رویه به‌صورت زیر عمل می‌کند:

الف- باید CA به‌طور قابل اطمینانی از شناساگر کارخواه CA مطلع باشد و این می‌تواند، برای مثال از طریق وارد کردن شناساگر باشد که توسط کاربر کارخواه به واسط ورودی وارد می‌شود حاصل شود. با این وجود، می‌تواند بخشی از خود پروتکل باشد. (به زیر مراجعه شود.)

ب- CA کلید عمومی خود P_{CA} را به کارخواه CA ارسال می‌کند، و کارخواه CA کلید عمومی خود P_M را به CA می فرستد. فرض می‌شود، این انتقال از طریق پیوند ارتباطاتی (نامطمئن) اتفاق می‌افتد. به همراه P_M، کارخواه CA می‌تواند هر نوع اطلاعات دیگری که تمایل دارد در احراز کلید عمومی شامل شود و از طریق CA تولید خواهد شد را ارسال کند. این موضوع می‌تواند برای مثال شامل شناساگر برای کارخواه باشد.

1- key management facility

پ- کارخواه و CA اکنون سازوکار احراز هویت دستی که در زیربند ۶-۲ مشخص شده را انجام می‌دهند تا تصدیق کنند کلیدهای عمومی مبادله شده درست هستند. کارخواه نقش افزاره B و CA نقش افزاره A را بازی می‌کند. داده D که در سازوکار احراز هویت دستی استفاده شده، شامل P_{CA} و P_M و هر نوع داده دیگر به‌وسیله کارخواه و CA تأمین و پشتیبانی می‌شود. این داده اضافی ممکن است شامل شناساگرهای یکتای CA و کارخواه باشد.

ت- اگر (و تنها اگر) کارخواه (افزاره B) یک شاخص موفقیت نشان دهد، کاربر به CA دستور می‌دهد تا یک گواهی کلید عمومی مناسب تولید کند. سپس این گواهی می‌تواند از طریق پیوند ارتباطی (احتمالاً حفاظت نشده) به کارخواه ارسال شود.

ث- کارخواه (افزاره B) اکنون، قبل از پذیرش گواهی، دو چیز را وارسی می‌کند، ابتدا کارخواه، امضای مورد استفاده کلید عمومی CA را تصدیق می‌کند (P_{CA})، دوم اینکه کارخواه تصدیق می‌کند که داده موجود در گواهی (شامل کلید عمومی P_M و شناساگر کاربر) همان داده مورد انتظار است و رویه کامل می‌شود.

پ-۴ کلید عمومی تولید شده در بیرون

اگر کلید عمومی کارخواه CA از طریق CA یا هر وسیله مورد اطمینان تولید کننده کلید، تولید شود، کلید عمومی باید به‌طور امنی به کارخواه منتقل شود.

در این جا یک رویه برای انتقال کلید عمومی از CA به کارخواه تشریح می‌شود. گام‌های رویه به‌صورت زیر است:

الف- کارخواه و CA، همان‌طور که در پیوست ب شرح داده شده است، یک کلید سرّی مشترک ایجاد می‌کنند.

ب- با استفاده از کلید سرّی ایجاد شده در گام الف، CA کلید خصوصی کارخواهِ رمزشده و به‌صورت یکپارچه حفاظت شده را به کارخواه می‌فرستد که در آنجا به‌صورت امن ذخیره می‌شود. همچنین، CA دوباره کلید عمومی خودش (P_{CA}) را به‌صورت یکپارچه حفاظت شده با استفاده از کلید سرّی ایجاد شده در گام الف به کارخواه می‌فرستد. فنون رمز بندی متقارن مطابق استاندارد ملی ۱۰۸۲۴-۳و استاندارد ISO/IEC 18033-4، استانداردسازی شده است.

پ) اکنون کارخواه هر اطلاعاتی که می‌خواهد در گواهی خود که با استفاده از کلید سرّی ایجاد شده در گام الف- به‌صورت یکپارچه حفاظت شده شامل شود را می‌فرستد.

ت- CA گواهی برای P_M،کلید عمومی کارخواه تولید می‌کند. سپس، این گواهی می‌تواند از طریق پیوند ارتباطاتی (احتمالاً حفاظت نشده) به کارخواه ارسال شود.

ث- اکنون، کارخواه قبل از پذیرش گواهی، دو چیز را وارسی می‌کند، ابتدا کارخواه، امضای مورد استفاده کلید عمومی CA را وارسی می‌کند (P_{CA})، دوم اینکه کارخواه تصدیق می‌کند که داده موجود در گواهی (شامل کلید عمومی P_M و شناساگر کاربر) همان داده مورد انتظار است و رویه کامل می‌شود.

پیوست ت

(اطلاعاتی)

امنیت سازوکار و انتخاب‌ها برای طول پارامتر

ت-۱ کلیات

در این پیوست امنیت ۸ سازوکار احراز هویت دستی که در این استاندارد شرح داده شد، مورد بحث قرار می-گیرد. همچنین راهنماهایی برای انتخاب توابع مقدار- وارسی، مقدار- خلاصه، MACها، رشته بیت‌های تصادفی و کلیدها فراهم شده است.

ت-۲ استفاده از سازوکارهای ۱ و ۲

همه داده‌هایی که باید از طریق پیوند ارتباطاتی مابین دو افزاره منتقل شوند، عمومی فرض شده‌اند، حتی اگر در بعضی مواقع ممکن است قسمتی از داده D سرّی باشد. هدف امنیتی سازوکار احراز هویت دستی حفاظت از یکپارچگی داده است و هدف محرمانگی آن نیست. حفاظت یکپارچگی ضروری، با استفاده از رویه بررسی بر پایه تابع مقدار- وارسی انجام می‌شود.

تابع مقدار- وارسی ، یک نگاشت f از یک فضای داده D و یک فضای کلید K به یک فضای مقدار- وارسی C است:

$$f: D \times K \rightarrow C, \quad c = f(d,k)$$

در سازوکار ۱ و ۲ مقدار- وارسی‌ها برای محافظت از یکپارچگی داده استفاده می‌شوند. بنابراین امنیت این سازوکارها بر پایه امنیت غیر شرطی از تابع مقدار- وارسی به جای امنیت محاسباتی است. امنیت غیر شرطی از توابع مقدار- وارسی بر پایه نتایج توسعه یافته از تئوری احراز هویت پیام هستند، برای مثال به بند ۴-۵ در [۲۸] مراجعه شود. دو نوع اصلی حملات عموماً ملاحظه می‌شود:

- حملات جعل هویت، و
- حملات جایگزینی

در یک حمله جعل هویت، حمله‌کننده سعی می‌کند دریافت‌کننده را متقاعد سازد که داده توسط فرستنده مشروع بدون مشاهده تبادل داده قبلی میان فرستنده و گیرنده صورت گرفته است. در یک حمله از نوع جایگزینی، حمله‌کننده ابتدا برخی داده D را مشاهده می‌کند و با داده دیگر $\hat{d} \neq d$ جایگزین می‌کند. احتمال اینکه حمله‌کننده در یک حمله جعل هویت و حمله جایگزینی موفق شود به‌وسیله P_1 و P_s مشخص و اینگونه بیان می‌شود:

$$P_1 \triangleq \max_{c \in C, d \in D} P_k(c = f(d,k))$$

$$P_s \triangleq \max_{d,\hat{d} \in D, d \neq \hat{d}} P_k\left(c = f(\hat{d},k) \middle| c = f(d,k)\right).$$

امنیت هر دو سازوکار بستگی به این احتمال دارد که حمله‌کننده موفق به جایگزینی دادهٔ مشاهده شده D با داده دیگر $\hat{d} \neq d$ شود. اگر $\hat{d}$ توسط مؤلفه به‌عنوان داده درست پذیرفته شود حمله‌کننده موفق عمل کرده است. از آنجایی که ما فرض کرده‌ایم دو افزاره فوق به‌صورت فیزیکی به همدیگر نزدیک هستند و هیچ داده‌ای پذیرفته نمی‌شود مگر اینکه هر دو افزاره فوق با سیگنالی نشان دهند که آماده هستند، بنابراین حمله جعل هویت به سناریوی احراز هویت دستی اعمال وارد نخواهد بود. علاوه‌براین موقعیت معمول برای حفاظت یکپارچه با استفاده از یک MAC این است که هر دو داده و MAC فرستاده می‌شوند و می‌تواند توسط حمله‌کننده مشاهده شود. این مورد برای سازوکار ۱ و ۲ نیست که در آن‌ها، مقدار- وارسی به جای MAC استفاده می‌شود. در اینجا داده فقط برروی کانال عمومی فرستاده می‌شود و حمله‌کننده تا بعد از اینکه داده D فرستاده شود، خروجی تابع مقدار- وارسی را نمی‌داند. (در واقع در سازوکار ۱، حمله‌کننده هرگز به خروجی این تابع دسترسی ندارد). این امر تحلیل امنیت و بیان حملات جایگزینی موفق را ساده می‌سازد. از این رو احتمال موفقیت در جایگزینی برای سازوکارهای ۱ و ۲ را می‌توان اینگونه بیان کرد:

$$P_s = \max_{d,\hat{d}\in D, d\neq\hat{d}} p(f(d,k) = f(\hat{d},k) | \text{مشاهده شود } d)$$

بنابراین کلید K که به‌صورت یکسان و تصادفی از فضای کلید K انتخاب می‌شود، در این صورت احتمال بالا را می‌توان اینگونه بیان کرد:

$$P_s = \max_{d,\hat{d}\in D, d\neq\hat{d}} \frac{|\{k \in K: f(d,k) = f(\hat{d},k)\}|}{|K|}$$

که $|K|$ نشان‌دهنده عدد اصلی مجموعه K است. این معادله این نتیجه را در پی دارد که به‌منظور فراهم کردن امنیت بالا، باید احتمال برخورد تابع مقدار- وارسی کم باشد. که می‌توان با استفاده از تابع مقدار- وارسی که از کدهای اصلاح خطا به‌دست می‌آید، آن را تضمین کرد. نمونه آن در پیوست ث شرح داده شده است.

بر پایه تحلیل بالا، کلید به طول ۱۶-۲۰ بیت و یک مقدار- وارسی به طول ۱۶-۲۰ بیت توصیه می‌شود. جدول پیوست ث احتمالات حملات موفق برای طول‌های ۱۶ و ۲۰ بیت را فراهم می‌کند.

ت-۳ استفاده از سازوکار ۳ و ۵

امنیت سازوکارهای ۳ و ۵ بر پایه اصولی است که سازوکارهای ۱ و ۲ مبتنی بر آن بود. در این سازوکار به جای تابع مقدار- وارسی باید از یک تابع خلاصه و یک تابع درهم‌ساز استفاده شود. توابع درهم‌ساز در استاندارد ISO/IEC 10118 استانداردسازی شده است [۱۱] و یک تابع درهم‌ساز از این استاندارد برای استفاده در سازوکارهای ۳ و ۵ توصیه شده است.

از آنجایی که کلید تصادفی بلند K برای برآورده کردن شرط دوم تابع درهم‌ساز استفاده شده است، یک کلید ۱۶۰ K بیتی برای این مورد توصیه می‌شود.

از آنجایی که مقادیر- خلاصه به‌صورت دستی منتقل می‌شوند یا بین دو افزاره A و B مقایسه می‌شوند، یک تابع خلاصه که خروجی آن $b = 16$ تا 20 بیت باشد، برای این مورد توصیه می‌شود. تابع خلاصه، مشابه الگوریتم MAC است، با این تفاوت که مقدار- خلاصه کوتاه‌تر از طول معمول یک MAC است. در نتیجه،

یک ساختار خلاصه ممکن شامل استفاده از اولین b بیت از هر یک از توابع MAC یا خروجی تابع-درهم‌ساز رمز نگاشتی استاندارد باشد، همان‌طور که در [۲۵] و پیوست چ اشاره شده است.
اثبات امنیت سازوکارهای ۳ و ۵ در [۲۴] آمده است.

ت-۴ استفاده از سازوکارهای ۴ و ۶

امنیت سازوکارهای ۴ و ۶ بر پایه اصولی متفاوت از آنهایی است که سازوکارهای ۱ و ۲ مبتنی بر آن است. به جای تابع مقدار- وارسی، یک تابع درهم‌ساز در این سازوکار استفاده می‌شود. توابع درهم‌ساز در استاندارد ISO/IEC 10118 استانداردسازی شده است [۱۱] و یک تابع درهم‌ساز از این استاندارد برای استفاده در سازوکارهای ۴ و ۶ توصیه شده است.
مطابق سازوکارهای ۳ و ۵، طول کلید K برای این مورد ۱۶۰ بیت توصیه می‌شود. این سازوکارها شامل انتقال دستی یا مقایسه رشته بیت کوتاه R بین افزاره A و B هستند. بنابراین طول بیت برای R در این مورد ۱۶-۲۰ بیت توصیه می‌شود.
اثبات امنیت سازوکارهای ۴ و ۶ در [۲۴] آمده است.

ت-۵ استفاده از سازوکارهای ۷ و ۸

امنیت سازوکارهای ۴ و ۶ بر پایه اصولی متفاوت از آنهایی است که سازوکارهای ۱ و ۲ مبتنی بر آن است. به‌جای تابع مقدار- وارسی، باید یک تابع MAC در این سازوکارها استفاده شود. توابع MAC طبق استاندارد ISO/IEC 9797 استانداردسازی شده است و یک تابع MAC از این استاندارد برای استفاده در سازوکارهای ۷ و ۸ توصیه شده است.
یک رشته بیت تصادفی R، ۱۶-۲۰ بیتی برای این مورد توصیه می‌شود، اما MAC باید طول بیت بیشتری داشته باشد. اندازه خروجی تابع MAC که برای سازوکارهای ۷و ۸ استفاده شود باید در حدود ۱۲۸-۱۶۰ بیت باشد. به‌طور مشابه کلیدهای تصادفی K_A و K_B (و K_{At} و K_{Bt}) که به‌عنوان کلیدهای تابع MAC استفاده می‌شوند، باید حدوداً در اندازه مشابه یعنی ۱۲۸-۱۶۰ بیت باشند. همچنین زمان انقضای رویه‌ها باید برای تشخیص وقفه‌های ممکن در این سازوکار پیاده‌سازی شود.

پیوست ث

(اطلاعاتی)

روشی برای تولید مقادیر- وارسی کوتاه

ث-۱ کلیات

در این پیوست یک تابع مقدار - وارسی برای استفاده در سازوکارهای ۱ و ۲ تعیین می‌شود. احتمالات حملات موفق بر این سازوکارها در زمان بهره بردن از طرح پیشنهادی مقدار - وارسی نیز منظور می‌شود. با استفاده از عبارت حمله موفق در پیوست ت، یکی از رویکردهای مناسب همان استفاده از توابع مقدار - وارسی مشتق از نظریه کدگذاری است. رابطه بین کدهای اصلاح خطا و چنین مقادیر - وارسی در [۱۵] مورد بحث قرار می‌گیرد.

قبل از درنظرگرفتن مثال‌های سخت، دو تعریف اصلی از نظریه کدگذاری مفروض است. برای ساده‌تر شدن امر، فقط کدهای تعریف شده برروی یک حوزه معین Fq می‌شوند. یک کد q-ary را برروی Fq با V مشخص کنید. فرض کنید کد واژه‌ها دارای طول n هستند. کد، یک نگاشت از پیام‌ها به کدواژه‌ها است. هر پیام دارای کدواژه منحصربه‌فرد خود است. کد V شامل تمام بردارها است می باشد $v \in V = \{v^{(d)} : d \in D\}, v^{(d)} = (v_1(d), v_2(d), ..., v_n(d))$ که در آن $v_i(d) \in F_q$

دو تعریف دیگر لازم هستند.

تعریف: اگر x و y دو چندتایی q-ary از طول n باشند، پس می‌توان گفت که فاصله- همینگ به این شرح است:

$$d_H(x,y) \triangleq |\{i \in \{1,...,n\} : x_i \neq y_i\}|$$

تعریف: حداقل فاصله برای کد V به این شرح است:

$$d_H(V) \triangleq \min_{x,y \in V,\ x \neq y} d_H(x,y)$$

حال ما چگونگی ایجاد یک تابع مقدار- وارسی را برای استفاده با سازوکارهای ۱ و ۲ براساس یک کد نشان می‌دهیم. ساخت این تابع بسیار ساده است، و نگاشت از پیام و فضای کلید به سادگی به این صورت به‌دست می‌آید:

$$f(d,k) = v_k(d)$$

که در آن $k \in K = \{1,...,n\}$. از این‌رو، یک تابع مقدار- وارسی با یک اندازه کلید برابر با n و اندازه فضای پیام برابر با اندازه فضای کدگذاری، به‌دست می‌آید.

احتمال یک حمله جایگزینی موفق برای این ساخت به این شرح تعیین می‌شود. با توجه به عبارت Ps در پیوست ت، بلافاصله بعد از آن داریم:

$$P_H = \max_{d,\acute{d} \in D,\ d \neq \acute{d}} \frac{|\{k \in K : f(d,k) = f(\acute{d},k)\}|}{|K|} = \max_{d,\acute{d} \in D,\ d \neq \acute{d}} \frac{|\{k \in K : v_k(d) = v_k(\acute{d})\}|}{|K|}$$

$$= \max_{d,\acute{d} \in D,\ d \neq \acute{d}} \frac{n - d_H(v^{(d)}, v^{(\acute{d})})}{n} = 1 - \frac{d_H(V)}{n}.$$

با توجه به این عبارت دقیق برای احتمال حمله جایگزینی موفق، در نظر گرفتن برخی ساخت‌های سخت در این مرحله مفید است. ترجیحاً کدهای طولانی با فاصله بسیار حداقلی مورد نیاز هستند. این خصوصیت دارای کدهای معروف رید- سولومون (RS)[1] است. یک کد RS را می‌توان برروی یک فیلد اختیاری محدود، $\boldsymbol{F_q}$ ساخت. محاسبه‌ی یک کدواژه، بسیار ساده بوده و با ارزیابی چندجمله‌ای فیلد محدود سروکار دارد. داده‌ها (پیام) مورد نظر برای کدگذاری یک qتایی با طول t را برروی $\boldsymbol{F_q}$ بیان کنید، $d = d_0, d_1, \ldots, d_{t-1}, d_i \in F_q$ پس، چندجمله‌ای تعمیم داده شده برای کدگذاری RS به این صورت ارائه می‌شود:

$$p^{(d)}(x) = d_0 + d_1 x + d_2 x^2 + \cdots + d_{t-1} x^{t-1}.$$

تابع مقدار- وارسی با ارزیابی چندجمله‌ای در یک نقطه اختیاری $k \in \boldsymbol{Fq}$ به‌طور مستقیم تعیین می‌شود:

$$f(d,k) = v_k(d) = p^{(d)}(k) = d_0 + d_1 k + d_2 x k^2 + \cdots + d_{t-1} k^{t-1}.$$

کد تعمیم داده شده RS دارای خصوصیت‌های زیر است ([۲۵]):

$$n = q = |K|$$
$$|D| = q^t = n^t$$
$$d_H(V) = n - t + 1.$$

این کد بر این نکته دلالت دارد که $PS = (t\text{-}1)/n$ برای یک مقدار- وارسی به‌دست آمده از کد تعمیم داده شده RS. این احتمال با اندازه فضای پیام، D، افزایش می‌یابد. از این‌رو، یک رویکرد مناسب در وهله اول اعمال یک تابع درهم‌ساز یک راهه، از جمله یکی از توابع درهم‌ساز معین در استاندارد ISO/IEC 10118-3 ، برای داده‌ها بوده و سپس استفاده از خروجی تابع درهم‌ساز یک راهه به‌عنوان ورودی برای کد رید- سولومون است. نتیجه می‌گیریم که ما یک احتمال ضعیف را بدون افزایش معنی‌دار طول کلید یا طول خروجی تابع مقدار- وارسی حفظ کنیم. با استفاده از این رویکرد، یک فضای پیام به میزان حدود ۱۲۸ بیت (SHA-1 ناقص) امنیت کافی را فراهم می‌کند. در جدول ۱، دو مثال ساخت و احتمالات مربوطه به حملات ارائه شده‌اند.

1- Reed-Solomon

جدول ث-۱ مقادیر- وارسی کد RS: احتمال حمله جانشینی موفق، P_s

$\log_2\|D\|$	$\log_2(n)$	P_S
128	16	2^{-13}–2^{-16}
256	16	2^{-12}–2^{-16}
128	20	2^{-17}–2^{-20}
256	20	2^{-16}–2^{-20}

همان‌طور که از جدول قابل مشاهده است، یک کد دارای کلید ۴ رقمی در مبنای شانزده و مقدار- وارسی، یک احتمال جعل امضا به میزان 2^{-12} یا کمتر را ارائه می‌دهد. اگر این تعداد تا ۵ رقم در مبنای شانزده افزایش یابد، احتمال تا میزان 2^{-17} یا کمتر کاهش می‌یابد.

پیوست ج

(اطلاعاتی)

تحلیل مقایسه‌ای امنیت و کارایی سازوکارهای ۱ تا ۸

از زمانی‌که سازوکارهای ۱، ۲، ۷ و ۸ در سال ۲۰۰۵ استاندارد شده‌اند (سازوکارهای ۷ و ۸ تحت عناوین ۳ و ۴ نامگذاری شده‌اند)، تعدادی شمای بهتر ارائه شده‌اند. این شماها، شماهایی هستند که ورودی کاربر کمتری استفاده می‌کنند و از طریق دستیابی به باند محکم‌تر، احتمال موفقیت را برای یک حمله‌کننده کاهش می‌دهند. علاوه براین، بعضی سازوکارهایی که هم اکنون موجودند دارای شواهد امنیتی هستند، در حالیکه سازوکارهای ۱، ۲، ۷ و ۸ چنین شاهد امنیتی قوی‌ای ندارند (اگرچه ما درباره مشکلات این سازوکارها آگاه هستیم). به ویژه کارهای لور و نیبرگ[۱۹]، وادنای و پاسنی[۲۵] و[۳۰]، کاگالج، کاپگن و هاباکس[۱]، وانگ و استاجانو[۳۲] و[۳۳]، هاپمن[۶]و[۷]، لیندل[۲۰] و نوین و رسکن[۲۱]،[۲۳] و[۲۴]. اگرچه تعدادی شماهای مختلف ارائه شده‌اند، همه آنها در یکی از دو گروه قرار گرفته‌اند با توجه به روشی که داده احراز هویت شده D به‌وسیله توابع رمزنگاشتی مورد استفاده که در [۲۴] اشاره شده، پردازش می‌شود. هر دو رویکرد که در سازوکارهای جدید نمایش داده شده‌اند در قسمت دوم موجودند. به‌طور مثال گروه (۱) سازوکارهای ۳ و ۵ از تابع خلاصه (خروجی کوتاه) استفاده می‌کنند تا D را به کلید تصادفی بلند K پیوند دهند و گروه (۲) سازوکارهای ۴ و ۶ برای پیوند دادن D به کلید تصادفی بلند K و یک کلید تصادفی کوتاه جریان داده‌ای R از تابع درهم‌ساز (خروجی بلند یا رمزنگاشتی) استفاده می‌کنند.

به‌منظور اینکه کاربران این استاندارد به بهترین تکنیک دسترسی داشته باشند، ۴ سازوکار جدید (سازوکارهای ۳ تا ۶) در بند ۷ آمده است. این سازوکارها، جایگزین‌های بهتر و کارآتری به نسبت سازوکارهای ۱، ۲، ۷ و ۸ پیشنهاد می‌دهند در حالی که نیاز به هیچ تغییری در واسطه‌های ورودی و خروجی افزاره ارتباطی از قبیل پروتکل‌ها ندارند.

همه سازوکارهای ۳ تا ۶ که به تازگی منتشر شده‌اند، شواهد امنیتی خود را دارند. به [۲۱]، [۲۲]، [۲۳]، [۲۴]، [۳۲] و [۳۳] مراجعه کنید.

خصوصیات اصلی ۴ سازوکار جدید به قرار زیر است:

- سازوکارهای ۳ تا ۶ مقدار انتقال یا مقایسه دستی داده مورد نیاز در سازوکارهای ۱ تا ۲ را نصف می‌کنند. در سازوکارهای ۱ و ۲ کاربر باید به‌صورت دستی یک کلید کوتاه و یک مقدار کنترلی از یک افزاره ارتباطی به دیگری انتقال دهد، یا به‌صورت چشمی مقادیر نمایش داده شده افزاره ارتباطی را مقایسه کند. با نصف کردن مقادیر تراکنش‌های انسانی، در سازوکارهای ۳ تا ۶ ، کاربرهای انسانی تنها نیاز دارند که یک مقدار- خلاصه کوتاه یا جریان- بیت تصادفی کوتاه را انتقال دهند یا مقایسه کنند، هر دو به‌عنوان کلید کوتاه یا مقدار- وارسی در سازوکارهای ۱ تا ۲ دارند.

یادآوری ۱– انتقال دستی کلید تصادفی کوتاه، یک مقدار- وارسی یا یک مقدار- خلاصه مهمتر از عمل‌های ساده‌تر مثل فشردن دکمه‌هایی برای آغاز پروتکل، خواندن نتایج یک بیتی (پذیرش/ رد)، یا سایر مقایسه‌های sing bit

هستند که در همه سازوکارها اتفاق می‌افتند. نوع اخیر تراکنش‌های انسانی در این تحلیل و در جدول ج-۱ نادیده گرفته شده است.

- سازوکارهای ۳ تا ۶ سطح امنیتی بالاتری نسبت به سازوکارهای ۱ تا ۲ دارند. علیرغم اینکه آنها تنها نیاز به نصف کردن تلاش انسانی مورد نیاز در سازوکارها دارند. در [۲۱]و [۲۴] نشان داده شده است که سازوکارهای ۱ تا ۲، به‌خاطر طول بیت کوتاه کلید K که در محاسبه مقدار- وارسی سودمند است تا آن اندازه‌ای که ایده‌آل است امنیت ارائه نمی‌دهند. در مقابل، کلید K مورد استفاده در سازوکارهای ۳ تا ۶ می‌تواند به پیوند ارتباطاتی مشترک (پهنای باند بالا) ارسال شود. از این رو به‌طور مؤثری می‌تواند بلندتر باشد مثلاً، همان‌طوری که در ضمیمه D مشخص شده ۱۶۰ بیت. این خاصیت از پهنای باند کوتاه‌تر تئوریکی طول کلید توابع درهم‌ساز عمومی ناشی می‌شود، که به طور گسترده در [۲]، [۵]، [۲۷]، [۳۱] مطالعه شده‌اند.
- سازوکارهای ۳ تا ۶ دارای نشان‌های امنیتی آورده شده در [۲۱] و [۲۴] هستند. اگرچه سازوکارهای ۳ تا ۶ با هم متفاوتند، سطح امنیتی یکسان و مشابهی با توجه مقدار برابر تلاش انسانی دارند.
- اگرچه سازوکارهای ۷ تا ۸، مانند سازوکارهای ۳ تا ۶ سطح امنیتی یکسانی فراهم می‌آورند، به داده دستی سرّی کلید تصادفی کوتاه R که منتقل می‌شود تکیه دارند. برخلاف سازوکارهای ۱ تا ۶، کلید تصادفی در سازوکارهای ۷ تا ۸ باید سرّی نگهداری شود، تا اینکه تنها به افزاره وسایل ارتباطی و کاربر انسانی شناخته شود. بنابراین در طول انتقال دستی داده، باید مراقبت شود تا از مشاهده کلید تصادفی به‌طور مثال به‌وسیله دوربین مخفی جلوگیری شود. اگر کلید سرّی به‌خطر افتد، یک مزاحم ممکن است یک حمله مرد میانی را انجام دهد.

جدول ج-۱ تفاوت انتقال داده دستی و امنیت را بین سازوکارهای ۱ تا ۸ خلاصه کرده است. در اینجا، حمله موفقیت‌آمیز نشان‌گر آن است که پروتکل جایی اجرا شود که وسایل ارتباطی رشته بیتی یکسان که دستی منتقل شده است محاسبه می‌کنند، هر چند رقیب (مزاحم) داده D را به‌صورت موفقیت‌آمیز دست‌کاری کرده است وقتی در یک پیوند ارتباطی (ناامن) مشترک مبادله می‌شود تا اینکه وسایل ارتباطاتی نسخه متفاوتی از داده D دریافت می‌کنند.

یادآوری ۲- در سازوکارهای ۱، ۲، ۳، ۵، ۷ و ۸ کلیدهای مورد استفاده با یک تابع مقدار- وارسی، یک تابع خلاصه یا یک الگوریتم MAC در هر دوره پروتکل همیشه تصادفی و تازه هستند. حملات جانشینی، که به استفاده دوباره از یک کلید واحد برای ورودی‌های چند متنی تکیه دارند همان‌طور که در ضمیمه ت-۲ ذکر شد، مناسب نیستند. اطلاعات بیشتر درباره نشان‌های امنیتی و تعاریف یک حمله موفقیت‌آمیز در [۲۱] و [۲۴] موجود است.

برای مقایسه مقدار انتقال داده دستی، طول‌های بیت یک مقدار- خلاصه (سازوکارهای ۳ و ۵)، یک رشته - بیتی تصادفی کوتاه R (سازوکارهای ۴، ۶، ۷ و ۸)، یک مقدار- وارسی و یک کلید سرّی کوتاه با یک تابع مقدار- وارسی (سازوکارهای ۱ و ۲) همگی b بیت هستند.

چون دو نوع تراکنش انسانی در این سازوکارها وجود دارند: ۱ـ انتقال دستی بیت‌های اطلاعات (از قبیل مقدار- وارسی و رشته‌های - بیتی تصادفی کوتاه) و ۲ـ مقایسه دستی دو رشته - بیتی، جدول ج۱ از نشان‌گذاری (c و m) استفاده خواهد کرد تا تعداد بیت‌های انتقال دستی داده (m) و تعداد بیت‌های مقایسه‌ای (c) را نشان دهد. (به جدول ج۱ مراجعه شود.)

جدول ج-۱- مقایسه بین سازوکارهای ۱ تا ۸

سازوکار	نوع افزاره‌های A و B	تعاملات انسان (در بیت‌های عمومی/سرّی)	توابع رمزنگاشتی	احتمال یک حمله موفق
۱(قدیمی)	A: واسط ورودی ساده B: واسط خروجی ساده	(2b, 0) (داده عمومی)	تابع مقدار- وارسی	بزرگتر از 2^{-b}
۲(قدیمی)	A و B: واسط ورودی ساده	(0, 2b) (داده عمومی)	تابع مقدار- وارسی	بزرگتر از 2^{-b}
۳(جدید)	A: واسط ورودی ساده B: واسط خروجی ساده	(b, 0) (داده عمومی)	تابع- درهم‌ساز و تابع خلاصه	$2^{-b} + \epsilon$
۴(جدید)	A: واسط ورودی ساده B: واسط خروجی ساده	(b, 0) (داده عمومی)	تابع- درهم‌ساز	$2^{-b} + \epsilon$
۵(جدید)	A و B: واسط ورودی ساده	(0, b) (داده عمومی)	تابع- درهم‌ساز و تابع خلاصه	$2^{-b} + \epsilon$
۶(جدید)	A و B: واسط ورودی ساده	(0, b) (داده عمومی)	تابع- درهم‌ساز	$2^{-b} + \epsilon$
۷(قدیمی)	A و B: واسط خروجی ساده	(2b, 0) (داده سری)	الگوریتم MAC	$2^{-b} + \epsilon$
۸(قدیمی)	A: واسط ورودی ساده B: واسط خروجی ساده	(b, 0) (داده سری)	الگوریتم MAC	$2^{-b} + \epsilon$

یادآوری۳- سازوکارهای ۱، ۲، ۷ و ۸ در سال ۲۰۰۵ استاندارد شده‌اند و لذا به‌عنوان «قدیمی» در جدول ج.۱ نشان‌گذاری شده‌اند در مقابل سازوکارهای ۳ تا ۶ به عنوان «جدید» نشان‌گذاری شده‌اند.

یادآوری۴– انتقال دستی داده یا بیت‌های عمومی نشانگر آن است که داده انتقالی برای هر کس در سازوکارهای ۱ تا ۶ قابل شناسایی است. این بر خلاف سازوکارهای ۷ تا ۸ است که کلید کوتاه منتقل شده دستی برای هر کس به‌جز وسایل ارتباطی و کاربر انسانی سرّی است.

یادآوری۵– ما مکانیزها را براساس توابع رمزنگاشتی مورد استفاده طبقه‌بندی می‌کنیم زیرا وقتی ما سازوکارهای ۳ تا ۶ را در قالب پیچیدگی محاسباتی در ضمیمه ج-۲ مقایسه می‌کنیم اطلاعات مفید می‌شوند.

یادآوری۶- در جدول ج-۱ ، ϵ یک مقدار قابل اغماض در مقابل 2^{-b} است، در حالی که « بزرکتر از 2^{-b} » نشانگر آن است که احتمال یک حمله موفقیت‌آمیز بزرگتر از 2^{-b} با یک مقدار غیر قابل اغماض است.

در میان سازوکارهای جدید اضافه شده، همچنین یک مزیت نهفته در کارایی محاسباتی سازوکارهای ۳ و ۵ نسبت به سازوکارهای ۴ و ۶ وجود دارد. این موضوع به‌وسیله مشاهدات انجام شده در [۱۶]، [۲۱]، [۲۲] و [۲۳] قابل توضیح است. داده D که گره‌ها برای احراز هویت، تنها نیاز دارد تا به‌وسیله یک تابع خلاصه خروجی کوتاه در سازوکارهای ۳ و ۵ پردازش شود، (برای مثال b=۱۶-۲۰) در مقابل در سازوکارهای ۴ و ۶ به‌وسیله یک تابع درهم‌ساز خروجی بلند (به‌طور مثال ۱۶۰ بیت یا بیشتر). چون در عمل داده D به‌صورت معمول بلند است، به‌طور مثال ممکن است شامل تصاویر یا DVDهایی باشد که به طور چشمگیری از یک کلید تصادفی K بلندتر هستند، به‌صورت بالقوه محاسبه یک خلاصه کوتاه D از مقدار- وارسی بلند سریعتر است. اگر این‌رو سازوکارها بر وسایل ارتباطی کوچک و یک برنامه رمزنگاشتی سبک پیاده سازی شوند جایکه بهینه‌سازی کارایی محاسباتی است یک معیار مهم به‌حساب می‌آید.

پیوست چ

(اطلاعاتی)

روش‌هایی برای تولید مقدار- خلاصه کوتاه

در این ضمیمه دو ساختار از توابع خلاصه که مناسب استفاده در سازوکارهای ۳ و ۵ هستند تشریح می‌شود. اول از الگوریتم MAC مشتق می‌شود، مانند آنکه در استاندارد ISO/IEC 9797، استاندارد شده است؛ دومی از تابع درهم‌ساز مشتق می‌شود، مانند آنکه در استاندارد ISO/IEC 10118، استاندارد شده است.

یادآوری۱- تعداد متنوعی از دیگر سازوکارها که همچنین می‌تواند برای محاسبه توابع خلاصه استفاده شود وجود دارد، از قبیل یکی از آن‌ها که براساس ضرب ماتریس Toeplitz یا ضرب عدد صحیح که در [۱۷]، [۲۳] و [۲۱] پیشنهاد شده است. اما این ساخت‌ها برای توابع خلاصه تماماً تحلیل و آزموده نشده‌اند، بنابراین آن‌ها در این پیوست تشریح نشده‌اند.

در تعاریف زیر، تابع $trunc_b(x)$ خروجی‌ای می دهد که b بیت سمت چپ رشته- بیت x را تشکیل می‌دهد.

تعریف۱: برای محاسبه مقدار- خلاصه پیام m با استفاده از کلید k، محاسبه کنید:

$d(m,k) = trunc_b(MAK_k(m))$

تعریف۲: برای محاسبه مقدار- خلاصه پیام m با استفاده از کلید K، محاسبه کنید:

$d(m,k) = trunc_b(h(m \| k))$

یادآوری۲- روش داده شده در تعریف ۱ در [۴] ارائه شده است و روش داده شده در تعریف ۲ در [۲۲] توضیح داده شده است.

کتاب‌نامه

[۱] استاندارد ملی شماره ۳-۱۰۸۲۲: سال۱۳۸۷، فناوری اطلاعات- فنون امنیتی- مدیریت کلید- قسمت ۳: - ساز و کارهای مبتنی بر فنون نامتقارن

[۲] استاندارد ملی شماره ۳-۱۰۸۲۴: سال ۱۳۸۷، فناوری اطلاعات- فنون امنیتی- الگوریتم‌های رمزنگاری- قسمت۳: رمزهای بلوکی

[3]M. Cagalj, S. Capkun, and J. Hubaux, 'Key agreement in peer-to-peer wireless networks', in: Proceedings of the IEEE, Special Issue on Security and Cryptography 94(2) (2006), 467–478

[4]J.L. Carter and M.N. Wegman, 'Universal Classes of Hash Functions', Journal of Computer and System Sciences 18(2) (1979), 143–154

[5]C. Gehrmann and K. Nyberg, 'Enhancements to Bluetooth baseband security', in: Proceedings of Nordsec 2001, Copenhagen, Denmark, November 2001

[6]C. Gehrmann, C.J. Mitchell and K. Nyberg, 'Manual authentication for wireless devices', Cryptobytes 7(1) (2004), 29–37

[7]P. Gemmell and M. Naor, 'Codes for Interactive Authentication', in: Advances in Cryptology – Crypto 1993, LNCS, Vol. 773, D.R. Stinson, ed., Springer, 1993, pp. 355–367

[8]J.-H. Hoepman, 'Ephemeral Pairing on Anonymous Networks', in Proceedings of the Second International Conference on Security in Pervasive Computing (SPC 2005), LNCS, Vol. 3450, D. Hutter and M. Ullmann, eds., Springer, 2005, pp. 101–116

[9]J.-H. Hoepman, 'Ephemeral Pairing Problem', in: Proceeding of the 8th International Conference on Financial Cryptography, LNCS, Vol. 3110, A. Juels, ed., Springer, 2004, pp. 212–226

[10]ISO 7498-2:1989, Information processing systems — Open Systems Interconnection — Basic Reference Model — Part 2: Security Architecture

[11]ISO/IEC 9797 (all parts), Information technology — Security techniques — Message Authentication Codes (MACs)

[12]ISO/IEC 8825-1, Information technology — ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)

[13]ISO/IEC 10118 (all parts), Information technology — Security techniques — Hash-functions

[14]ISO/IEC 8825-1, Information technology — ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)

[15]ISO/IEC 18033-4:2005, Information technology — Security techniques — Encryption algorithms — Part 4: Stream ciphers

[16]G. Kabatianskii, B. Smeets and T. Johansson, ‘On the cardinality of systematic authentication codes via error correcting codes’, IEEE Transactions on Information Theory 42(2) (1996), 566–578

[17]K. Khoo, F.-L. Wong and C.-W. Lim, 'On a construction of short digests for authenticating ad-hoc networks', in: Proceedings of ICCSA 2009, LNCS vol. 5593, pp. 863-876

[18]H. Krawczyk, 'New Hash Functions for Message Authentication', in: Advances in Cryptology – Eurocrypt 1995, LNCS, Vol. 921, L.C. Guillou and J.-J. Quisquater, eds., Springer, 1995, pp. 301–310

[19]J.-O. Larsson, ‘Higher layer key exchange techniques for Bluetooth security’, in: Opengroup Conference, Amsterdam, October 2001 Single-user licence only, copying and networking prohibited.

[20]S. Laur and K. Nyberg, ‘Efficient Mutual Data Authentication Using Manually Authenticated Strings’, LNCS, Vol. 4301, D. Pointcheval, ed., Springer, 2006, pp. 90–107

[21]A.Y. Lindell. 'Comparison-Based Key Exchange and the Security of the Numeric Comparison Mode in Bluetooth v2.1', in: Proceedings of the Cryptographers' Track at the RSA Conference 2009 on Topics in Cryptology, LNCS, Vol. 5473, M. Fischlin, ed., Springer, 2009, pp. 66–83

[22]L. H. Nguyen and A. W. Roscoe, ‘Authentication protocols based on low-bandwidth unspoofable channels: a comparative survey’, Journal of Computer Security (to appear). See: http://eprint.iacr.org/2010/206.pdf

[23]L. H. Nguyen and A. W. Roscoe, ‘Authenticating ad hoc networks by comparison of short digests’, Information & Computation 206(2–4) (2008), 250–271

[24]L. H. Nguyen and A. W. Roscoe, ‘Efficient group authentication protocol based on human interaction’ in: Proceedings of Workshop on Foundation of Computer Security and Automated Reasoning Protocol Security Analysis, 2006, pp. 9–31. See: http://eprint.iacr.org/2009/150

[25]L. H. Nguyen and A. W. Roscoe, ‘Separating two roles of hashing in one-way message authentication’, in Proceedings of Workshop on Foundation of Computer Security, Automated Reasoning Protocol Security Analysis, and Issues in the Theory of Security, 2008, pp. 195–209. See: http://eprint.iacr.org/2009/003

[26]S. Pasini and S. Vaudenay, ‘SAS-based Authenticated Key Agreement’, in Proceedings of International Conference on Practice and Theory in Public Key Cryptography (PKC 2006), LNCS, Vol. 3958, M. Yung, Y. Dodis, A. Kiayias and T. Malkin, eds., Springer, 2006, pp. 395–409

[27]I.S. Reed and G. Solomon, ‘Polynomial codes over certain finite fields’, SIAM Journal 8 (1960), 300–304

[28]D.R. Stinson, ‘Universal Hashing and Authentication Codes’, in Advances in Cryptology – Crypto 1991, LNCS, Vol. 576, J. Feigenbaum, ed., Springer, 1992, pp. 74–85

[29]D.R. Stinson, ‘Cryptography – Theory and Practice’, CRC Press, 2002, 2nd edition

[30]SHAMAN Project Deliverable D13 (Annex 2), Final technical report – Workpackage 2 – Security for distributed terminals, 2003. Available at http://www.ist-shaman.org/

[31]S. Vaudenay, ‘Secure Communications over Insecure Channels Based on Short Authenticated Strings’, in: Advances in Cryptology – Crypto 2005, LNCS, Vol. 3621, V. Shoup, ed., Springer, 2005, pp. 309-326

[32]M.N. Wegman and J.L. Carter, ‘New Hash Functions and Their Use in Authentication and Set Equality’, Journal of Computer and System Sciences 22(3) (1981), 265–279

[33]F.-L. Wong and F. Stajano, ‘Multi-channel Protocols’, in Proceedings of the 13th International Workshop on Security Protocols, LNCS, Vol. 4631, B. Christianson, B. Crispo, J.A. Malcolm and M. Roe, eds., Springer, 2005, pp. 128–132

[34]F.-L.Wong and F. Stajano, ‘Multi-channel Security Protocols’; IEEE Pervasive computing 6(4) (2007), 31–39