کارگاه های آموزشی مرکز اطلاعات علمی جهاد دانشگاهی

کارگاه آنلاین آشنایی با پایگاه های اطلاعات علمی بین المللی و ترفند های جستجو

کارگاه آنلاین
پروپوزال نویسی و پایان نامه نویسی

کارگاه آنلاین
بررسی مقابله ای متون (مقدماتی)

یک سیستم احرازهویت در یک محیط توزیع شده

An Authentication System in Distributed Environment

لاله شیخ غلامحسین قندهاری دکتر علی موقر رحیم آبادی
ghandehari@ce.sharif.edu movaghar@sharif.edu
دانشکده مهندسی کامپیوتر
دانشگاه صنعتی شریف

چکیده

مهمترین مسئله در ایجاد یک سیستم توزیع شده امن، احراز هویت کاربران سیستم است. ساده ترین تکنیک در احراز هویت کاربران استفاده از کلمه عبور است. در این مقاله سیستم احراز هویتی بر مبنای کلمه عبور ارائه می شود. اساس این سیستم پروتکل Gong است که پروتکلی مقاوم در برابر حدس کلمه عبور و در عین حال بهینه در نوع خود است. معماری سیستم به صورت سلسله مراتبی و در دو سطح است. احراز هویت در هر ناحیه به صورت متمرکز و توسط دو کارگزار بر خط و یک پایگاه داده انجام می شود، علاوه بر این سیستم خاصیت SSO و forward secrecy را نیز پشتیبانی می کند.

کلمات کلیدی: احرازهویت، توزیع کلید، سیستم توزیع شده، محرمانگی، امنیت، رمزنگاری.

۱ مقدمه

احرازهویت کاربران یکی از مسائل اساسی در امنیت کامپیوترها و شبکه های کامپیوتری است. از میان تکنیک های احراز هویت کلمه عبور از آنجا که احتیاج به وسیله خاصی ندارد، بیشتر از سایر روش ها مانند کارت هوشمند مورد توجه است. با توجه به اینکه کاربران معمولاً کلمه های عبور ساده ای انتخاب می کنند، سیستم های احرازهویت در معرض خطر حدس کلمه عبور ویا حمله به فهرست[۱] هستند. دو تکنیک متفاوت برای مقابله با حدس کلمه عبور وجود دارد: کلمه های عبور یک بار مصرف[۲] و پروتکل های مقاوم در برابر حدس کلمه عبور. کلمه های عبور یک بار مصرف هزینه سربار برای کارفرما وکارگزار تولید می کنند [۸]. از میان تکنیک های کلمه عبور یک بار مصرف می توان به طرح Lamport اشاره کرد. در این

[۱] Dictionary attack
[۲] One time password

۱	$A \rightarrow B$	$[A,B,na1,na2,ca,na3,[na3]_{Ka}]_{KUs}$, ra
۲	$B \rightarrow S$	$[A,B,na1,na2,ca,na3,[na3]_{Ka}]_{KUs}$, $[B,A,nb1,nb2,cb,nb3,[nb3]_{Kb}]_{KUs}$
۳	$S \rightarrow B$	$[na1, k \oplus na2\ ]_{na3}$, $[nb1, k \oplus nb2\ ]_{nb3}$
۴	$B \rightarrow A$	$[na1, k \oplus na2\ ]_{na3}$, $[f1(ra) , rb\ ]_{k}$
۵	$A \rightarrow B$	$[f2(rb)]_{k}$

شکل ۱ : پروتکل Gong کمینه از نظر تعداد پیام

روش اگر مقدار محرمانه کاربر pw باشد و H تابع درهم ساز[۳]، دنباله کلمه های عبور pw ، H(pw) ، H(H(pw))، ... و$H^t(pw)$ خواهد بود. هزینه سرباری که این روش برای کارفرما دارد اعمال مکرر تابع H و هزینه سربار برای کارگزار نگهداری مقدار کلمه عبور قبلی هر کاربر است.

تا کنون پروتکل های احراز هویت بسیاری مقاوم در برابر حدس کلمه عبور ارائه شده است. این پروتکل ها اگرچه کلمه های عبور ضعیف را در برابر حدس محافظت می کنند، ولی از نظر تعداد دور و پیام کمینه نیستند. از این میان می توان به پروتکل Nonce که احراز هویت را با ۷ پیام انجام می دهد، اشاره کرد [۳].

در سال ۱۹۹۵ پروتکلی توسط Gong ارائه شد که علاوه بر مقاوم بودن در برابر حدس کلمه عبور با توجه به [۱] در محیط خود (چالش و پاسخ بر اساس نانس و handshake) بهینه است[۲]. پروتکل Gong ، pull model است به این معنی که کارگزار برای دریافت کلید از کارگزار احراز هویت اقدام می کند؛ این پروتکل برای محیط های WAN مناسب است به این دلیل که در این شبکه ها تعداد کارفرماها بسیار بیشتر از تعداد کارگزارها است و کارگزار احراز هویت به کارگزارها نزدیکتر از کارفرماها است (در برابر push model که کاربر برای دریافت کلید با کارگزار احراز هویت ارتباط برقرار می کند). این پروتکل در شکل ۱ آمده است. در این شکل KU_s کلید عمومی و در مقابل KR_s کلید خصوصی S است. na1 ، na2، ca، na3 و ra نانس و $[x]_y$ به معنی رمزگذاری x با کلید y است. $\oplus$ عملیات یاء انحصاری را نشان می دهد و f یک تابع شناخته شده و از پیش تعیین شده است.

ایده اصلی پروتکل Gong انتخاب یک عدد تصادفی، na3 ، است که کارگزار برای رمزنگاری پاسخ پیام از آن استفاده می کند. این عدد تصادفی اگر چه توسط کارفرما انتخاب می شود، در مقایسه با کلمه عبور کاربر که انتخاب پروتکل های دیگر است، از نظر رمزنگاری قوی تر است. همچنین به دلیل آنکه از این عدد تصادفی تنها یک بار برای رمزگذاری پاسخ استفاده می شود، شکستن آن خطری برای کلمه عبور و کلید دوره های بعدی ندارد. به این ترتیب پروتکل خاصیت forward secrecy را پشتیبانی می کند. خاصیت forward secrecy به معنی جلوگیری از خطرات و لورفتن اطلاعات بیشتر در زمان لو رفتن یک اطلاع محرمانه است [۷]. در این پروتکل مهاجم با داشتن کلمه عبور کاذبذ قادر به دستیابی به کلید دوره نیست و همچنین فاش شدن کلید یک دوره باعث فاش شدن کلید دوره های بعدی نمی شود.

قابل ذکر است که تعداد اعداد تصادفی به کار رفته در پروتکل نیز کمینه است و حذف هر یک از آنها صحت پروتکل را دچار مشکل می کند. به عنوان مثال اگر na3 از پیام ۱ حذف شود در این صورت مهاجم با ارسال پیام به صورت $[A,B,na1,na2,ca,x]_{KUs}$ کارگزار را در تفسیر x به اشتباه خواهد انداخت و x به صورت $[na3]_{Ka}$ تفسیر می شود.

[۳] Hash function

مهاجم با دریافت پیام $y = [na1, k \oplus na2]_{na3}$ قدرت تست حدس خود در مورد کلمه عبور را خواهد داشت. به این ترتیب که مهاجم مقداری را برای Ka انتخاب کرده و از آن برای رمزگشایی x و به دست آوردن na3 استفاده می کند سپس از این مقدار برای رمزگشایی y و تست کلمه عبوری که حدس زده است، استفاده می کند. همچنین اگر از پیام شماره ۳ مقدار na1 حذف شود و به صورت $[k \oplus na2]_{na3}$ درآید. مهاجم می تواند خود را به جای B جا زده و با شرکت در پروتکل مقدار k را به طور قانونی به دست بیاورد. به این ترتیب با دانستن $k \oplus na2$ می تواند حدس خود از Ka را تست کند. پروتکل ارائه شده نه تنها کلمه عبور را از خطر مهاجمان خارجی بلکه از خطر مهاجمان داخلی نیز حفظ می کند. به این معنی که A قدرت حدس کلمه عبور B حتی با وجود اطلاعات باقیمانده از اجرای موفق پروتکل، را نخواهد داشت؛ همچنین B نیز قادر به حدس کلمه عبور A نیست.

در این مقاله سیستم احراز هویتی بر مبنای پروتکل Gong ارائه می شود. در ادامه توسعه پروتکل Gong برای پشتیبانی از خاصیت SSO[۴] و ارتباط بین نواحی آورده می شود. طبق تعریف خاصیت SSO به مکانیزمی اطلاق می شود که بوسیله آن با انجام عملیات احرازهویت و مجوز دسترسی برای یکبار، می توان به کاربر اجازه دسترسی به همه کامپیوترها و سیستم هایی که کاربر مجوز دسترسی به آنها را دارد، داد؛ بدون آنکه نیازی به وارد کردن چندین باره کلمه عبور باشد[۵]. تحلیل امنیتی پروتکل توسعه یافته Gong در بخش بعدی مورد بررسی قرار می گیرد و در ادامه آن نیز نتایج حاصل از پیاده سازی نمونه ای سیستم ارائه می شود.

۲ معماری سیستم

سیستم موردنظر، احرازهویت در هر ناحیه را به صورت متمرکز و توسط یک کارگزار بر خط انجام می دهد. هر کارگزار سیستم از سه مؤلفه تشکیل شده است:

۱. کارگزار احرازهویت[۵] (AS) : وظیفه این کارگزار، احرازهویت کاربران به ازاء ورود اولیه آنها است و خاصیت SSO را پشتیبانی می کند.

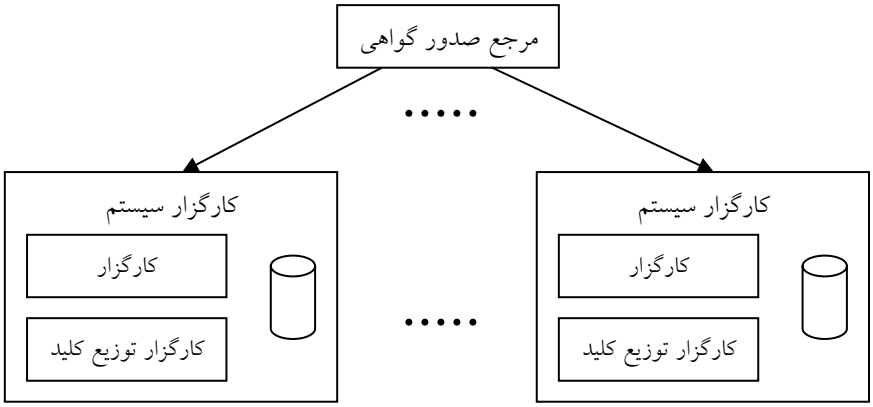
۲. کارگزار توزیع کلید[۶] (KDS) : توزیع کلید بین کاربر و کارگزار را به طور امن بر عهده دارد. کارگزار توزیع کلید به ازاء هر درخواست سرویس، کلید محرمانه‌ای بین کاربر و کارگزار مورد نظر مبادله می کند.

۳. پایگاه داده احرازهویت : این پایگاه داده وظیفه نگهداری از کلمه های عبور کاربران و کارگزارهای عضو در ناحیه را دارد. همچنین کلیدهایی که برای ورود اولیه کاربر توسط کارگزار احرازهویت صادر می شود، برای دسترسی کارگزار توزیع کلید در این پایگاه داده ذخیره می‌شود.

کارگزار توزیع کلید و کارگزار احرازهویت از نظر منطقی از یکدیگر مجزا هستند ولی به دلیل استفاده از پایگاه داده مشترک و به منظور فراهم آوردن امنیت بیشتر و همچنین حذف ساعت همگام در سیستم، بر روی یک میزبان اجرا می شوند. کاربر برای ورود به سیستم، کلمه عبور خود را به AS ارائه می دهد و AS پس از اطمینان از هویت کاربر کلید موقتی با مدت اعتبار محدود صادر می کند. این کلید که کلید احراز هویت نامیده می شود، توسط AS در پایگاه داده احراز هویت ذخیره

---

[۴] Single Sign On
[۵] Authentication server
[۶] Key distribution server



شکل۲ : معماری سیستم احرازهویت

می شود و همچنین در اختیار کاربر قرار می گیرد. کاربر برای استفاده از هر سرویس در شبکه، به جای کلمه عبور، کلید احراز هویت را به KDS ارائه می کند. به این ترتیب کاربر نیازی به وارد کردن چندین باره کلمه عبور به ازاء هر سرویس را ندارد.

این تکنیک علاوه بر آنکه باعث تضمین خاصیت SSO و کاهش مبادله کلمه عبور و در نتیجه کاهش تهدیدهایی که متوجه آن است، می شود؛ به دلیل محدود بودن مدت اعتبار کلید احراز هویت خطرات ناشی از دزدیده شدن و لو رفتن این کلید در مقایسه با کلمه عبور که کلید بلند مدت کاربر است، بسیار کمتر و محدودتر خواهد بود. با قرار گرفتن کارگزار احرازهویت و کارگزار توزیع کلید در یک میزبان نیز برخلاف Kerberos به ساعت همگام در کل سیستم نیازی نیست؛ و اعتبار کلید با ساعت یک میزبان مطابقت داده می‌شود.

محیط توزیع شده مجموعه ای از گروه ها و دامنه ها است که هر دامنه اجرا، ناحیه[۷] نامیده می شود. از آن جا که کاربر می تواند درخواست دریافت سرویس حتی از ناحیه دیگر را نیز داشته باشد و به دلیل آنکه اطلاعات کاربران و کارگزارها در پایگاه داده ناحیه‌ای که در آن ثبت نام کرده‌اند، قرار دارد. برای ارتباط کاربر با کارگزاری در ناحیه دیگر هر یک از دو طرف باید در ناحیه مربوط به خود احرازهویت شوند و سپس نتیجه این احرازهویت به ناحیه طرف مقابل آنها منتقل شود.

ارتباط بین نواحی به صورت سلسله مراتبی و در دو سطح انجام می‌شود. شکل ۲ معماری دو سطحی سیستم را نشان می دهد. در بالاترین سطح یک مرکز صدور گواهی[۸] قرار دارد که وظیفه آن صدور گواهینامه برای کارگزارهای سیستم است، این مرکز می تواند به صورت برون خطی عمل کند. در سطح دوم کارگزارهای سیستم قرار دارند که وظیفه آنها احرازهویت در هر ناحیه است. هر یک از این کارگزارها چنانچه گفته شد از سه مؤلفه مجزا تشکیل شده اند. کارگزارهای احرازهویت توسط مرکز گواهی هویت و گواهینامه های صادر شده توسط آن ارتباط بین نواحی را انجام می دهند.

۳ پروتکل سیستم

پروتکل به کار رفته در سیستم به منظور پشتیبانی از خاصیت SSO در شکل ۳ آمده است. مرحله اول، شکل ۳-الف به ازاء هر بار ورود کاربر یک بار اجرا می‌شود. در این مرحله کاربر با ارائه کلمه عبور خود به کارگزار احراز هویت کلید احراز هویت،

---

[۷] Realm
[۸] Certificate authority

الف – رویه بدست آوردن کلید احرازهویت.

۱	$A \rightarrow AS$	$[A,KDS,na1,na2,ca,na3,[na3]_{Ka}]_{KUs}$
۲	$AS \rightarrow A$	$[na1, Kt \oplus na2\ ]_{na3}$

ب – رویه بدست آوردن کلید محرمانه سرویس.

۳	$A \rightarrow B$	$[A,B,na'1,na'2,ca,na'3,[na'3]_{Kt}]_{KUs}$, ra
۴	$B \rightarrow KDS$	$[A,B,na'1,na'2,ca',na'3,[na'3]_{Kt}]_{KUs}$, $[B,A,nb1,nb2,cb,nb3,[nb3]_{Kb}]_{KUs}$
۵	$KDS \rightarrow B$	$[na'1, k \oplus na'2\ ]_{na'3}$, $[nb1, k \oplus nb2\ ]_{nb3}$
۶	$B \rightarrow A$	$[na'1, k \oplus na'2\ ]_{na'3}$, $[f1(ra) , rb\ ]_{k}$
۷	$A \rightarrow B$	$[f2(rb)]_{k}$

شکل ۳ : تبادل پیام در پروتکل سیستم

Kt، را دریافت می کند. سپس در مرحله دوم شکل ۳–ب، که به ازاء هر بار درخواست دریافت سرویس انجام می شود، کاربر با ارائه کلید احراز هویت، کلید دوره، K، را دریافت می کند.

با فرض آنکه کارگزار احرازهویت و کارگزار توزیع کلید در یک میزبان اجرا می شوند، کلید عمومی و خصوصی هر دو کارگزار با یکدیگر برابر هستند. همچنین این امر باعث می شود تا برخلاف بلیط در Kerberos اطلاعات بیشتری در اختیار کاربر قرار نگیرد. (برای اطلاعات بیشتر به [۴] مراجعه شود.)

شکل ۴ توسعه پروتکل Gong برای پشتیبانی از ارتباط بین نواحی را نشان می دهد. در مرحله اول کاربر توسط کارگزار احراز هویت ناحیه خود احراز هویت شده و Kt را دریافت می کند. در صورت نیاز به ارتباط با کارگزاری در ناحیه دیگر پیام شماره ۱ به کارگزار مورد نظر (در ناحیه دیگر) ارسال می شود. کارگزار توزیع کلید در ناحیه دوم، KDSrem، تنها قادر به رمزگشایی بخش دوم پیام ۲ است و برای رمزگشایی بخش اول پیام با توجه به $Realm_a$، بخش اول پیام به همراه $[KDSrem ,KDS, ns1,ns2,cs,ns3,[ns3]_{KRsrem}]_{KUs}$ را به کارگزار توزیع کلید ناحیه A ارسال می کند. این کارگزار، KDS، به دلیل امضای رقمی پیام، اطمینان دارد که پیام از جانب کارگزار توزیع کلید ناحیه دیگر است؛ پس از رمزگشایی پیام، کلید دوره مورد نظر توسط KDS ناحیه A انتخاب می شود و پیام شماره ۴ به KDSrem ارسال می شود. بخش دوم پیام شماره ۴ به دلیل اطلاع KDSrem از کلید انتخاب شده توسط کارگزار KDS است. KDSrem با دریافت پیام شماره ۴ قادر است مقدار کلید، k، را از بخش دوم آن استخراج نموده و پیام ۵ را به B ارسال کند.

به این ترتیب هیچ گونه اطلاع محرمانه ای بین نواحی مبادله نمی شود و هر کارگزار توزیع کلید مقدار کلید را برای ناحیه خود رمز می کند. تعداد پیامهای مبادله شده برای احرازهویت بین نواحی کمینه است. به این دلیل که احرازهویت بین نواحی با تبادل کمتر از دو پیام و محرمانه نگه داشتن مقادیر محرمانه ممکن نیست.

۱	$A \rightarrow B$	$[A, Realm_a, B,na'1,na'2,ca',na'3,[na'3]_{Kt}]_{KUs}$, ra
۲	$B \rightarrow KDSrem$	$[A, Realm_a, B,na'1,na'2,ca',na'3,[na'3]_{Kt}]_{KUs}$, $[B, Realm_b, A,nb1,nb2,cb,nb3,[nb3]_{Kb}]_{KUsrem}$
۳	$KDSrem \rightarrow KDS$	$[A, Realm_a, B,na'1,na'2,ca',na'3,[na'3]_{Kt}]_{KUs}$, $[KDSrem ,KDS, ns1,ns2,cs,ns3,[ns3]_{KRsrem}]_{KUs}$
۴	$KDS \rightarrow KDSrem$	$[na'1, k \oplus na'2]_{na'3}$, $[ns1, k \oplus ns2]_{ns3}$
۵	$KDSrem \rightarrow B$	$[na'1, k \oplus na'2]_{na'3}$, $[nb1, k \oplus nb2]_{nb3}$
۶	$B \rightarrow A$	$[na'1, k \oplus na'2]_{na'3}$, $[f1(ra) , rb]_k$
۷	$A \rightarrow B$	$[f2(rb)]_k$

شکل ۴ : رویه بدست آوردن کلید محرمانه سرویس در ناحیه دیگر

۴ تحلیل امنیتی پروتکل

از آنجاکه KUs کلید عمومی کارگزار است تنها کارگزار قادر به رمزگشایی پیام شماره ۱ و به دست آوردن مقدار na3 و ایجاد پیام شماره ۲ در شکل ۳ است. همچنین به دلیل وجود na1، این پیام جدید و از جانب کارگزار است؛ در نتیجه k کلید محرمانه‌ای است که توسط کارگزار انتخاب شده است. اگر مهاجم قصد حدس کلمه عبور ،Ka، را داشته باشد باید بتواند پیام شماره ۱ را دوباره درست کند به دلیل آنکه در پیام های بعدی هیچ متنی برای تست حدس خود ندارد. برای ایجاد دوباره پیام ۱ علاوه بر Ka مهاجم باید ca را نیز حدس بزند و از آنجا که ca عدد تصادفی است و از فضای بزرگی انتخاب شده است حدس آن بوسیله جستجو در این فضا غیر ممکن است.

دلیل اصلی آنکه پروتکل ارائه شده کمینه و مقاوم در برابر حدس کلمه عبور است؛ این است که جدید بودن پیام برای کارگزار احرازهویت اهمیتی ندارد. در این پروتکل اگر مهاجم پیام دوره با دوره های قبل پروتکل را دوباره به کارگزار ارسال کند، پیام یا پیامهایی به صورت $[na1, k \oplus na2]_{na3}$ و $[na1, k' \oplus na2]_{na3}$ دریافت خواهد کرد، که این هیچ کمکی برای حدس کلمه عبور، Ka و یا کلید دوره های بعدی، K'' نمی کند و تنها باعث می شود تا بارکاری کارگزار با پیامهای تکراری افزایش یابد. به این ترتیب حمله replay به این پروتکل خطری را متوجه سیستم نمی کند. همچنین حمله به کلمه عبور نیز تنها با دزدیدن کلمه عبور امکان پذیر است و حدس آن ممکن نیست.

با توجه به [۱] استفاده از زمان مهر و ساعت همگام در مقایسه با نانس بهینه تر است و دلیل این امر نیز آن است که کارگزار احرازهویت با استفاده از زمان مهر می تواند از جدید بودن پیام مطمئن شود. ولی از آنجا که امنیت پروتکل ارائه شده به جدید بودن درخواست های احراز هویت از کارگزار بستگی ندارد. در نتیجه استفاده از زمان مهر کارآیی پروتکل را افزایش نخواهد داد. پروتکل احرازهویت مقاوم در برابر کلمه عبور حتی با وجود ساعت همگام کاراتر از پروتکل مبتنی بر نانس نخواهد بود. علاوه بر این درستی پروتکل Gong و همچنین توسعه آن که در بخش ۳ آمده است، بوسیله منطق BAN[۹] قابل اثبات است. جزئیات این اثبات در [۹] آمده است.

[۹] منطق BAN در سال ۱۹۸۹ توسط Burrow، Abadi و Needham برای درستی یابی پروتکل های احرازهویت ارائه شد. این منطق بر اساس عقاید و اطلاعات اعضا بنا شده است. منطق BAN از زمان ایجاد تا کنون توسعه های فراوانی یافته است. برای اطلاعات تکمیلی به [۶] مراجعه شود.

۵ پایگاه داده سیستم

پایگاه داده ای که در کارگزار سیستم وظیفه نگهداری از کلمه های عبور را دارد، پایگاه داده احرازهویت نام دارد. اطلاعاتی که به ازاء هر کاربر عضو در این پایگاه داده ذخیره می‌شود، شامل موارد زیر است:

- مشخصه عضو : هر عضو با دو مؤلفه نام عضو و نام ناحیه آن شناخته می‌شود.
- کلید محرمانه یا کلمه عبور عضو: به منظور فراهم کردن امنیت بیشتر برای کلید محرمانه، این کلید پس از اعمال یک تابع درهم ساز ویا پس از رمزگذاری توسط کلید عمومی کارگزار در پایگاه داده ذخیره می شود.
- تاریخ انقضای این مشخصه.
- طول عمر کلیدهایی که برای این عضو صادر می‌شود. این فیلد، زمان انقضای کلید احراز هویت صادر شده برای عضو را مشخص می کند.

همچنین در پایگاه داده احرازهویت به ازاء هر کارگزار نیز یک رکورد ذخیره می‌شود. اطلاعات مربوط به کارگزار شامل نام کارگزار و کلید خصوصی آن است. در پاسخ به درخواست کاربر، کارگزار احراز هویت کلید احراز هویت را در پیام شماره ۲ (شکل۳) به کاربر ارسال می کند، همچنین این کلید در پایگاه داده احراز هویت نیز ذخیره می شود. به این ترتیب به ازاء هر بار ورود کاربر به سیستم نیز باید یک رکورد شامل : ۱- مشخصه کاربر ۲- کلید محرمانه احرازهویت ۳- آدرس میزبان که کاربر از آن استفاده می کند ۴- زمان انقضای کلید، در پایگاه داده ذخیره شود.

کارگزار توزیع کلید با دریافت پیام شماره ۴ با توجه به بازیابی کلید احرازهویت از پایگاه داده احرازهویت قادر است مقدار na3 را برای ارسال پیام شماره ۵ به B استخراج کند. رکوردهایی که توسط کارگزار احراز هویت ثبت می‌شوند برای کاهش حجم پایگاه داده پس از طی شدن زمان حیات آن از پایگاه داده حذف می‌شوند.

۶ نتیجه گیری

نمونه آزمایشگاهی از سیستم ارائه شده، بر روی سیستم عامل Linux پیاده سازی شده است. این نمونه از بانک اطلاعاتی mysql و بسته نرم افزاری mysql++ 1.7.9 استفاده می کند. همچنین برای انجام عملیات رمزنگاری نیز از نسخه 9.7.beta2 بسته نرم‌افزاری OpenSSL استفاده شده است.

جدول های ۱ و۲ نتایج اجرای برنامه را بر روی پردازنده Intel PIII 800 MHZ نشان می دهد. جدول ۱ میانگین زمان اجرای برنامه برای AS و کارفرما، برای دریافت کلید احراز هویت و ورود اولیه به سیستم را به تفکیک نشان می دهد. به عبارت دیگر این زمان برای کامل کردن پیام های ۱و ۲ در شکل ۳ است. جدول ۲ نیز میانگین زمان اجرای برنامه در هر عضو در زمان درخواست دریافت سرویس از جانب کارفرما را نشان می دهد.

جدول ۱ : میانگین زمان اجرا در هر یک از اعضا برای ورود به سیستم

عضو	زمان (msec)
کارگزار احرازهویت AS	45.3
کارفرما	14.9

جدول ۲ : میانگین زمان اجرا در هر یک از اعضا برای دریافت سرویس

عضو	زمان (msec)
کارگزار توزیع کلید KDS	93.4
کارگزار	49.5
کارفرما	17.7
کارفرما + handshake	17.7+4.1

مقایسه سیستم ارائه شده با سیستم های احراز هویت مشابه مانند Kerberos نشان می دهد که علاوه بر آنکه این سیستم امنیت بیشتری را فراهم می کند و بر خلاف Kerberos در برابر حدس کلمه عبور و حمله های replay مقاوم است، از نظر کارآیی نیز قابل مقایسه با آن است. علاوه بر این سیستم ارائه شده بر خلاف Kerberos و SPX به ساعت همگام نیاز ندارد، همچنین سایر خصوصیات مانند SSO و forward secrecy را نیز پشتیبانی می کند.

۷ منابع

[1] L. Gong , " Efficient Network Authentication protocols: Lower Bounds and Optimal Implementations ", Technical Report SRI-CSL-94-15, Computer Science Laboratory, SRI International, Menlo Park, California, 1994.

[2] L. Gong , "Optimal Authentication protocols Resistant to Password Guessing Attack ", 8th IEEE Computer Foundation Workshop,1995.

[3] L. Gong , T.M.A. Lomas, R.M. Needham, J.H. Saltzer. " Protecting Poorly Chosen Secrets From Guessing Attacks", IEEE Journal on Selected Area in Comminications, 1993.

[4] http://web.mit.edu/kerberos/www/ , Date of access 8/20/2003.

[5] Open Group. " Introduction to Single Sign-On " , available at http://www.opengroup.org/security/sso/.

[6] P. Syverson, I. Cervesato. " The Logic of Authentication Protocols", In Foundations of Security Analysis and Design, 2001.

[7] http://srp.stanford.edu ,Date of access 08/25/2003.

[8] N. Haller, C. Metz, P. Nesser, M. Straw, "A one-time password system", RFC2289, 1998.

[۹] لاله شیح غلامحسین قندهاری، " یک سیستم احرازهویت در یک محیط توزیع شده" ، پایان نامه کارشناسی ارشد، دانشگاه صنعتی شریف، ۱۳۸۲.

کارگاه های آموزشی مرکز اطلاعات علمی جهاد دانشگاهی

کارگاه آنلاین آشنایی با پایگاه های اطلاعات علمی بین المللی و ترفند های جستجو

کارگاه آنلاین
پروپوزال نویسی و پایان نامه نویسی

کارگاه آنلاین
بررسی مقابله ای متون (مقدماتی)