ششمین کنفرانس بین المللی انجمن رمز ایران
۱۵ و ۱۶ مهر ماه ۱۳۸۸، دانشگاه اصفهان
انجمن رمز ایران
دانشگاه اصفهان

بهبود یک پروتکل توافق کلید چندتایی مبتنی بر زوج‌سازی

محمد سبزی‌نژادفراش[۱]، محمود گردشی[۲]، مجید بیات[۳]

[۱و۳] تهران، پژوهشکده پردازش هوشمند علائم، گروه رمز و امنیت اطلاعات
{m.sabzinejad, m.bayat}@rcisp.com
[۲] تهران، دانشگاه امام حسین(ع)، مرکز تحقیقات فتح
mgardeshi@ihu.ac.ir

چکیده

زوج‌سازی دوخطی یک نگاشت است که اخیرا در طراحی پروتکل‌های رمزنگاری مورد توجه قرار گرفته است. ویژگی دوخطی این نگاشت، سبب جذاب شدن آن شده است. ما در این مقاله قصد داریم از این نگاشت در طراحی یک پروتکل توافق کلید چندتایی جدید استفاده کنیم. قبل از ارائه این طرح پیشنهادی، ابتدا یک پروتکل توافق کلید چندتایی را که از زوج‌سازی بهره گرفته است مورد بررسی قرار داده و یک حمله پیشنهادی به آن ارائه می‌دهیم. سپس یک پروتکل پیشنهادی ارائه داده و نشان خواهیم داد که از لحاظ امنیتی و محاسباتی بهینه‌تر از طرح بررسی شده است.

واژه‌های کلیدی

رمزنگاری، پروتکل‌های توافق کلید، طرح‌های امضا، خم‌بیضوی، زوج‌سازی‌های دوخطی

۱- مقدمه

زوج‌سازی‌های دوخطی[۱] قبل از اینکه در طراحی الگوریتم‌های های رمزنگاری مورد استفاده قرار بگیرند، ابتدا به عنوان ابزاری برای حمله به سیستم‌های مبتنی بر خم‌بیضوی مطرح بودند. تا اینکه در سال ۲۰۰۰ میلادی Juxe [12] از زوج‌سازی‌ها برای طراحی پروتکل توافق کلید سه سویه استفاده کرد. هر چند این طرح برخی ویژگی‌های امنیتی را نداشت اما بدلیل ابزار نوینی که در طراحی از آن استفاده شده بود بسیار مورد توجه قرار گرفت. از این به بعد بود که سیل تحقیقات رمزنگاری به سمت زوج‌سازی سرازیر شد. برای آگاهی بیشتر راجع به کاربرد زوج‌سازی در رمزنگاری به [8] و فصل دهم از [4] مراجعه کنید.

از آنجایی که قصد داریم در این مقاله، از زوج‌سازی‌ها در طراحی پروتکل‌های توافق کلید استفاده کنیم لازم است راجع به این پروتکل‌ها مقدماتی را بیان کنیم.

پروتکل‌های توافق کلید یکی از مفاهیم مهم در رمزنگاری هستند. این پروتکل‌ها به کاربران امکان می‌دهند تا در یک شبکه ناامن بتوانند کلید محرمانه‌ای را به اشتراک گذاشته و از آن برای برقراری ارتباط امن در شبکه استفاده کنند.

اولین پروتکل توافق کلید با استفاده از رمزنگاری کلید عمومی توسط دیفی[۲] و هلمن[۳] [7] ارائه شد. اما این پروتکل در برابر حمله حمله مردی در میانه[۴] آسیب پذیر بود زیرا طرفین پروتکل هویت همدیگر را وارسی[۵] نمی‌کردند. راه‌کار معمول برای حل این مشکل، مشکل، استفاده از یک زوج کلید عمومی و خصوصی معتبر توسط هر کاربر است. رایج‌ترین روش‌های موجود برای این کار، زیرساخت کلید عمومی[۶] (رمزنگاری کلید عمومی مبتنی بر گواهینامه) و رمزنگاری کلید عمومی مبتنی بر شناسه[۷] است.

در سال ۱۹۹۵ منزس[۸] و همکارانش [14] یک پروتکل توافق کلید کلید بنام MQV ارائه دادند که در آن برای امضای کلیدهای

[1] Bilinear Pairings

[2] Diffie
[3] Hellman
[4] Man in the Middle Attack
[5] Verification
[6] Public Key Infrastructure (PKI)
[7] Identity-Based Cryptography
[8] Menezes

دیفی- هلمن از توابع درهمساز[1] استفاده نمی‌شود. این پروتکل در استانداردهای بین المللی ANSI X9.42 [2] ، ANSI X9.63 [32] و IEEE 1363 [11] استاندارد شده است. با استفاده از ایده MQV (یعنی امضای بدون تابع درهمساز)، هارن[2] و لین[3] در سال ۱۹۹۸ [9] ایده پروتکل‌های توافق کلید چندتایی[4] را مطرح کردند. در این پروتکل‌ها کاربران در هر نشست، چند کلید مشترک می‌سازند. این ایده از این جهت مورد توجه است که در هر بار اجرای پروتکل چند کلید مشترک ساخته می‌شود در حالیکه برای ساخت این تعداد کلید با استفاده از پروتکل‌های توافق کلید معمولی، باید چند بار پروتکل اجرا شود که سبب بالا رفتن هزینه محاسباتی و ارتباطی می‌شود.

Yen و Joye [16] نشان دادند که پروتکل هارن- لین در برابر حمله جعل امضا آسیب پذیر است و برای حل مشکل آن یک طرح پیشنهادی ارائه دادند. اما Wu و همکارانش [17] نشان دادند که این طرح نیز همان مشکل طرح هارن- لین را دارد و خود یک پروتکل بهبود یافته ارائه دادند ولی برخلاف ایده اولیه هارن- لین، از تابع درهمساز در آن استفاده کردند، با این وجود نیز مشکل حمله جعل هنوز پا برجا بود.

هارن و لین [10] با توجه به حملات ارائه شده بر روی طرح اولیه خود، به منظور برطرف کردن مشکل، یک تغییر در امضای آن بوجود آوردند و مدعی شدند که در اینصورت پروتکل در برابر حمله جعل مقاوم است اما طولی نکشید که Zhou و همکارانش [18] نشان دادند که این امضای جدید نیز در برابر حمله جعل آسیب پذیر است.

برای مطالعه دقیق‌تر پروتکل‌های توافق کلید چندتایی و نقاط ضعف آنها به فصل پنجم از [۱] مراجعه کنید که در آن مرور جامعی بر طرح‌های ارائه شده در این زمینه صورت گرفته است.

ما در این مقاله ابتدا یک پروتکل توافق کلید چندتایی را که از زوج‌سازی‌های دوخطی استفاده می‌کند مورد بررسی قرار می‌دهیم و یک حمله پیشنهادی روی آن ارائه می‌دهیم. سپس یک پروتکل بهبود یافته پیشنهاد می‌کنیم و نشان خواهیم داد که از لحاظ امنیتی و محاسباتی بهینه‌تر از پروتکل بررسی شده است.

۲- مفاهیم مقدماتی

۲-۱ زوج‌سازی‌های دوخطی

تعریف۱ (زوج‌سازی‌های دوخطی[5]): (صفحه ۱۸۳ از [4])
فرض کنید n یک عدد صحیح مثبت باشد و همچنین G_1 و G_2 گروه‌های جابجاپذیر جمعی با مرتبه n و عضو همانی صفر بوده و G_3 یک گروه ضربی دوری با مرتبه n و عضو همانی یک باشد. زوج سازی یک نگاشت بصورت $e: G_1 \times G_2 \to G_3$ می‌باشد که دارای ویژگی‌های زیر است:

- **دوخطی بودن[6]**: به ازای هر $P,P' \in G_1$ و $Q,Q' \in G_2$ روابط زیر برقرار می‌باشند:

$$e(P+P',Q) = e(P,Q)\cdot e(P',Q)$$
$$e(P,Q+Q') = e(P,Q)\cdot e(P,Q')$$

- **زوال ناپذیری[7]**: برای هر $P \in G_1$ $(P \neq 0)$ حداقل یک عضو $Q \in G_2$ وجود دارد بطوریکه $e(P,Q) \neq 1$ است. همچنین برای هر $Q \in G_2$ $(Q \neq 0)$ حداقل یک عضو $P \in G_1$ وجود دارد بطوریکه $e(P,Q) \neq 1$ است.
- **قابل محاسبه بودن**: یک الگوریتم برای محاسبه این نگاشت وجود داشته باشد. در عمل از زوج‌سازی‌های ویل[8] و تیت[9] استفاده می‌شود که از الگوریتم میلر[10] [15] برای محاسبه آنها بهره گرفته می‌شود.

امنیت سیستم‌های رمزنگاری مبتنی بر زوج‌سازی به سختی مسئله دیفی- هلمن دوخطی[11] که توسط بنه[12] و فرانکلین[13] [6] مطرح شد بستگی دارد.

تعریف۲ (مسئله دیفی- هلمن دوخطی BDHP): محاسبه $e(P,P)^{abc}$ به ازای مقادیر معلوم P، $P_1=[a]P$، $P_2=[b]P$ و $P_3=[c]P$ بطوریکه $e(P,P)\neq 1$.

مسئله دیفی- هلمن دوخطی (BDHP) از مسئله دیفی- هلمن خم‌بیضوی (ECDHP) و مسئله محاسباتی دیفی- هلمن (CDHP) سخت‌تر نیست.

فرض کنید E یک خم‌بیضوی روی میدان F_p و $P \in E(F_p)$ یک نقطه از مرتبه اول n باشد بطوریکه

$$n \mid \#E(F_p) \quad \text{و} \quad gcd(n,p)=1$$

همچنین فرض کنید k کوچکترین عدد صحیح مثبتی است که در رابطه $n \mid p^k-1$ صدق می‌کند. به مقدار k درجه نشاندن[14] گفته می‌شود.

[1] Hash Functions
[2] Harn
[3] Lin
[4] Multiple Key Agreement Protocols
[5] Bilinear Pairings
[6] Bilinearity
[7] Non-Degeneracy
[8] Wail
[9] Tate
[10] Miller
[11] Bilinear Diffie–Hellman Problem
[12] Boneh
[13] Franklin
[14] Embedding Degree

همانطور که قبلا گفته شد امنیت سیستم‌های رمزنگاری مبتنی بر زوج‌سازی به سختی مسئله دیفی- هلمن دوخطی بستگی دارد و از طرفی سختی این مسئله به سختی مسئله دیفی- هلمن در خم-بیضوی $E(F_p)$ و سختی مسئله دیفی- هلمن در میدان متناهی F_{p^k} بستگی دارد. لذا در کاربردهای رمزنگاری مبتنی بر زوج‌سازی باید مقدار n به اندازه کافی بزرگ انتخاب شود تا حل مسئله دیفی-هلمن در $E(F_p)$ سخت باشد و همچنین مرتبه F_{p^k} باید به اندازه کافی بزرگ باشد تا حل مسئله دیفی- هلمن در آن سخت باشد. برای برآورده شدن حداقل سطح امنیتی در سیستم‌های رمزنگاری مبتنی بر زوج‌سازی باید $n>2^{160}$ و $p^k>2^{1024}$ باشد.

۲-۲ ویژگی‌های امنیتی پروتکل‌های توافق کلید

در تحلیل امنیتی پروتکل‌های توافق کلید به روش غیر فرمال، امنیت پروتکل در برابر حملات موجود مورد ارزیابی قرار می‌گیرد. مهم‌ترین این ویژگی‌های امنیتی [5] عبارتند از:

امنیت کلید شناخته شده[1]: این ویژگی بیان می‌دارد که اگر مهاجم به یک کلید نشست دسترسی پیدا کرد توانایی بدست آوردن کلیدهای نشست بعدی را نداشته باشد.

امنیت پیشرو[2]: این ویژگی امنیتی بیان می‌دارد که درصورت آشکار شدن کلیدهای خصوصی طولانی‌مدت کاربران امنیت کلیدهای نشست قبلی به خطر نیفتد. اگر با آشکار شدن کلید خصوصی طولانی مدت یکی از کاربران، محاسبه کلیدهای نشست قبلی ممکن نباشد اما با داشتن کلیدهای خصوصی طولانی‌مدت هر دو طرف چنین امکانی وجود داشته باشد گوییم پروتکل دارای امنیت پیشرو جزئی[3] است. در صورتی که با داشتن کلیدهای خصوصی طولانی‌مدت هر دو طرف نیز محاسبه کلیدهای نشست قبلی امکان پذیر نباشد گفته می‌شود پروتکل دارای امنیت پیشرو کامل[4] است.

امنیت در مقابل جعل هویت با کلید آشکار شده[5]: درصورت آشکار شدن کلید خصوصی طولانی مدت طرف A ، مهاجمی که این کلید را در اختیار دارد نتواند خود را بجای طرف B به A معرفی کند.

امنیت کلید ناشناخته[6]: فرض کنید A و B در حال اجرای پروتکل توافق کلید هستند. مهاجم فعال C نباید بتواند به نحوی در اجرای پروتکل دخالت کند تا بعد از اتمام پروتکل، A بر این باور باشد که با B توافق کلید انجام داده اما B معتقد است با C یک کلید محرمانه مشترک ساخته است.

حمله «مردی در میانه»[7]: فرض کنید A و B در حال اجرای پروتکل توافق کلید هستند. در این حمله مهاجم فعال C به نحوی در اجرای پروتکل دخالت می‌کند تا سبب شود طرفین پروتکل روی کلیدهای متفاوتی توافق کنند.

علاوه بر ویژگی‌هایی که ذکرشد دو ویژگی اساسی دیگر نیز برای پروتکل‌های توافق کلید وجود دارد که عبارتند از:

تأیید کلید ضمنی[8]: یک پروتکل توافق کلید دارای ویژگی تأیید کلید ضمنی است اگر طرفین پروتکل مطمئن باشند که فقط طرف مقابل آنها توانایی محاسبه کلید نشست را دارد.

تأیید کلید تضمینی[9]: یک پروتکل توافق کلید دارای ویژگی تأیید کلید تضمینی است اگر طرفین پروتکل مطمئن باشند که طرف مقابل آنها کلید نشست را محاسبه کرده‌است.

نمادگذاریهایی که در این مقاله برای توصیف پروتکل‌ها استفاده می‌شود در جدول(۲-۱) نشان داده شده است.

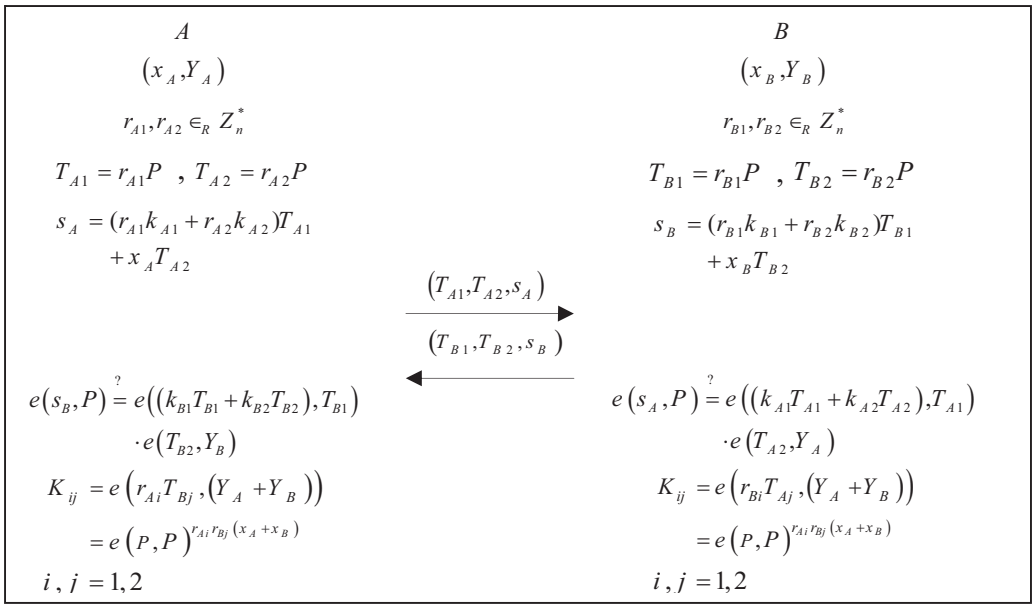
جدول۲-۱: نمادها

x_A, x_B	کلیدهای خصوصی طولانی مدت A و B
Y_A, Y_B	کلیدهای عمومی طولانی مدت A و B
$E(F_p)$	E یک خم‌بیضوی تعریف شده روی F_p
P	نقطه مبنا از مرتبه اول n روی خم‌بیضوی
$e(\ )$	زوج‌سازی دوخطی
r_A, r_B	کلیدهای خصوصی کوتاه مدت (مقادیر تصادفی) A و B
T_A, T_B	کلیدهای عمومی کوتاه مدت (مقادیر تصادفی) A و B
k_A, k_B	مولفه x نقاط T_A, T_B
s_A, s_B	امضاهای کاربران A و B
K	کلید نشست

۳- مروری بر پروتکل توافق کلید چندتایی LWW

این پروتکل [13] یکی از معدود پروتکل‌های توافق کلید چندتایی مبتنی بر گواهینامه است که از زوج‌سازی‌های دوخطی بهره گرفته است. هدف ارائه دهندگان این پروتکل رسیدن به سطح امنیتی بالاتر بوده اما حجم محاسباتی زیادی برای اجرای آن لازم است که از لحاظ عملی مطلوب نیست. این پروتکل برخلاف ادعای طراحان آن در برابر حمله جعل هویت آسیب پذیر است که ما در این بخش به آن می‌پردازیم.

[1] Known-Key Security
[2] Forward Secrecy
[3] Partial Forward Secrecy
[4] Perfect Forward Secrecy
[5] Key-Compromise Impersonation
[6] Unknown Key Security
[7] Man-In-The-Middle Attack
[8] Implicit Key Authentication
[9] Explicit Key Authentication



پروتکل۳-۱: پروتکل توافق کلید چندتایی LWW

۳-۱ پروتکل توافق کلید LWW

این پروتکل بر اساس رمزنگاری مبتنی بر گواهینامه است لذا زوج کلیدهای کاربران بصورت $\left(\left(x_A \in Z_n^*\right), \left(Y_A = x_A P\right)\right)$ ، $\left(\left(x_B \in Z_n^*\right), \left(Y_B = x_B P\right)\right)$ است. چگونگی اجرای این پروتکل همانطور که در پروتکل(۳-۱) نشان داده شده بصورت زیر است:

- A اعداد تصادفی $r_{A1}, r_{A2} \in_R Z_n^*$ را انتخاب می‌کند. سپس مقادیر $T_{A1} = r_{A1}P$ و $T_{A2} = r_{A2}P$ را محاسبه و با استفاده از رابطه $s_A = (r_{A1}k_{A1} + r_{A2}k_{A2})T_{A1} + x_A T_{A2}$ امضا می‌کند. A در پایان این مرحله مقادیر (T_{A1}, T_{A2}, s_A) را برای B ارسال می‌کند.
- B نیز اعداد تصادفی $r_{B1}, r_{B2} \in_R Z_n^*$ را انتخاب می‌کند. سپس مقادیر $T_{B1} = r_{B1}P$ و $T_{B2} = r_{B2}P$ را محاسبه و با استفاده از رابطه $s_B = (r_{B1}k_{B1} + r_{B2}k_{B2})T_{B1} + x_B T_{B2}$ امضا می‌کند. B در پایان این مرحله مقادیر (T_{B1}, T_{B2}, s_B) را برای A ارسال می‌کند.
- A به محض دریافت مقادیر (T_{B1}, T_{B2}, s_B) ابتدا صحت امضا را بصورت زیر بررسی می‌کند

$$e(s_B, P) \stackrel{?}{=} e\left(\left(k_{B1}T_{B1} + k_{B2}T_{B2}\right), T_{B1}\right) \cdot e\left(T_{B2}, Y_B\right)$$

در صورتی که تساوی بالا برقرار نباشد A اجرای پروتکل را رها می‌کند و درغیر این صورت کلیدهای نشست را بصورت زیر محاسبه می‌کند

$$K_{ij} = e\left(r_{Ai}T_{Bj}, \left(Y_A + Y_B\right)\right) = e\left(P,P\right)^{r_{Ai}r_{Bj}\left(x_A + x_B\right)} \quad ; \quad i, j = 1,2$$

- B نیز به محض دریافت مقادیر (T_{A1}, T_{A2}, s_A) ابتدا صحت امضا را بصورت زیر بررسی می‌کند

$$e(s_A, P) \stackrel{?}{=} e\left(\left(k_{A1}T_{A1} + k_{A2}T_{A2}\right), T_{A1}\right) \cdot e\left(T_{A2}, Y_A\right)$$

در صورتی که تساوی بالا برقرار نباشد B اجرای پروتکل را رها می‌کند و درغیر این صورت کلیدهای نشست را بصورت زیر محاسبه می‌کند

$$K_{ij} = e\left(r_{Bi}T_{Aj}, \left(Y_B + Y_A\right)\right) = e\left(P,P\right)^{r_{Ai}r_{Bj}\left(x_A + x_B\right)} \quad ; \quad i, j = 1,2$$

۳-۲ حمله پیشنهادی جعل هویت بر روی پروتکل LWW

در این حمله مهاجم قصد دارد خود را بجای A به B معرفی کند. بنابراین بصورت زیر رفتار می‌کند:

- مهاجم فعال C مقادیر تصادفی $r'_{A1}, r'_{A2} \in_R Z_n^*$ را انتخاب و سپس $t'_{A1} = r'_{A1}P$، $t'_{A2} = r'_{A2}P$ و امضای $s'_A = (r'_{A1}k'_{A1} + r'_{A2}k'_{A2})T'_{A1} + r'_{A2}Y_A$ را محاسبه می‌کند. در پایان این مرحله مهاجم مقادیر $\left(T'_{A1}, T'_{A2}, s'_A\right)$ را برای B ارسال می‌کند.
- B پس از دریافت مقادیر $\left(T'_{A1}, T'_{A2}, s'_A\right)$ بصورت زیر صحت امضا را بررسی می‌کند:

$$
\begin{aligned}
e(s'_A,P) &= e\left(\left(k'_{A1}T'_{A1}+k'_{A2}T'_{A2}\right),T'_{A1}\right)\cdot e\left(T'_{A2},Y_A\right)\\
&= e\left(\left(k'_{A1}r'_{A1}+k'_{A2}r'_{A2}\right)P,r'_{A1}P\right)\cdot e\left(T'_{A2},x_AP\right)\\
&= e\left(\left(k'_{A1}r'_{A1}+k'_{A2}r'_{A2}\right)r'_{A1}P,P\right)\cdot e\left(r'_{A2}x_AP,P\right)\\
&= e\left(\left(k'_{A1}r'_{A1}+k'_{A2}r'_{A2}\right)T'_{A1},P\right)\cdot e\left(r'_{A2}Y_A,P\right)\\
&= e\left(\left(k'_{A1}r'_{A1}+k'_{A2}r'_{A2}\right)T'_{A1}+r'_{A2}Y_A,P\right)\\
&= e(s'_A,P)
\end{aligned}
$$

همانطور که در رابطه بالا مشاهده می‌شود صحت امضا توسط B تایید می‌شود، یعنی B می‌پذیرد که پیامی از طرف A دریافت کرده است در حالیکه اینچنین نیست، بنابراین پروتکل(۳-۱) در برابر حمله جعل هویت ناامن است.

۳-۳ بررسی امنیت پیشرو در طرح LWW

امنیت پیشرو جزئی: این ویژگی امنیتی بیان می‌دارد که درصورت آشکار شدن کلید خصوصی طولانی‌مدت یکی از طرفین پروتکل، امنیت کلیدهای نشست قبلی به خطر نیفتد. در پروتکل(۳-۱) اگر مهاجم یکی از کلیدهای خصوصی طولانی مدت x_A یا x_B را در اختیار داشته باشد کلیدهای نشست

$$K_{ij} = e\left(r_{Bi}T_{Aj},\left(Y_B+Y_A\right)\right) = e\left(T_{Aj},T_{Bi}\right)^{(x_A+x_B)}$$

را نمی‌تواند محاسبه کند، زیرا برای محاسبه آنها به هر دو کلید خصوصی نیاز دارد، بنابراین پروتکل LWW امنیت پیشرو جزئی دارد.

امنیت پیشرو کامل: این ویژگی امنیتی بیان می‌دارد که درصورت آشکار شدن کلیدهای خصوصی طولانی‌مدت طرفین پروتکل، امنیت کلیدهای نشست قبلی به خطر نیفتد. در پروتکل(۳-۱) اگر مهاجم کلیدهای خصوصی طولانی مدت x_A و x_B را در اختیار داشته باشد کلیدهای نشست

$$K_{ij} = e\left(r_{Bi}T_{Aj},\left(Y_B+Y_A\right)\right) = e\left(T_{Aj},T_{Bi}\right)^{(x_A+x_B)}$$

را براحتی می‌تواند محاسبه کند، بنابراین پروتکل LWW امنیت پیشرو کامل ندارد.

۴- معرفی پروتکل پیشنهادی

ضعف پروتکل LWW از وجود عبارت x_AT_{A2} در رابطه امضا ناشی می‌شود، جزئیات این رابطه بصورت زیر است

$$x_AT_{A2} = x_Ar_{A2}P = r_{A2}\left(x_AP\right) = r_{A2}Y_A$$

همانطور که در رابطه بالا مشاهده می‌شود، ضرب شدن کلید خصوصی x_A در P عملا باعث بی تاثیر شدن آن در فرایند تعیین هویت می‌شود، زیرا رابطه امضا بصورت زیر خواهد بود

$$s_A = (r_{A1}k_{A1}+r_{A2}k_{A2})T_{A1}+r_{A2}Y_A$$

که بدون داشتن کلید خصوصی طولانی‌مدت کاربر A ، توسط هر فرد دیگری نیز قابل محاسبه است. به همین دلیل ما در طرح پیشنهادی خود که در ادامه به آن خواهیم پرداخت از وقوع چنین ترکیبی خودداری می‌کنیم.

۴-۱ توصیف پروتکل پیشنهادی

فرض کنید E یک خم‌بیضوی تعریف شده روی F_p باشد، همچنین فرض کنید نقطه مبنای $P\in E(F_p)$ از مرتبه اول n باشد. از آنجایی که پروتکل پیشنهادی مبتنی بر گواهینامه است لذا زوج کلیدهای کاربران بصورت $\left(\left(x_A\in Z_n^*\right),\left(Y_A=x_AP\right)\right)$ ، $\left(\left(x_B\in Z_n^*\right),\left(Y_B=x_BP\right)\right)$ می‌باشد.

چگونگی اجرای این پروتکل همانطور که در پروتکل(۴-۱) نشان داده شده است بصورت زیر است:

- A اعداد تصادفی $r_{A1},r_{A2}\in_R Z_n^*$ را انتخاب می‌کند، سپس مقادیر $T_{A1}=r_{A1}P$ و $T_{A2}=r_{A2}P$ را محاسبه و بصورت زیر امضا می‌کند

$$T_A = r_{A1}Y_B+T_{A2}$$
$$s_A = x_A-x_{T_A}\left(r_{A1}k_{A1}+r_{A2}k_{A2}\right) \mod n$$

A در پایان این مرحله مقادیر (T_{A1},T_{A2},s_A) را برای B ارسال می‌کند.

- B نیز اعداد تصادفی $r_{B1},r_{B2}\in_R Z_n^*$ را انتخاب می‌کند، سپس مقادیر $T_{B1}=r_{B1}P$ و $T_{B2}=r_{B2}P$ را محاسبه و بصورت زیر امضا می‌کند

$$T_B = r_{B1}Y_A+T_{B2}$$
$$s_B = x_B-x_{T_B}\left(r_{B1}k_{B1}+r_{B2}k_{B2}\right) \mod n$$

و در پایان این مرحله مقادیر (T_{B1},T_{B2},s_B) را برای A ارسال می‌کند.

- A به محض دریافت مقادیر (T_{B1},T_{B2},s_B) ابتدا صحت امضا را بصورت زیر بررسی می‌کند

$$T_B = x_AT_{B1}+T_{B2}$$
$$Y_B \stackrel{?}{=} s_BP+\left(x_{T_B} \bmod n\right)\left(k_{B1}T_{B1}+k_{B2}T_{B2}\right) \qquad (۴-۱)$$

در صورتی که تساوی (۴-۱) برقرار نباشد A اجرای پروتکل را رها می‌کند و درغیر اینصورت کلیدهای نشست را بصورت زیر محاسبه می‌کند

A		B
(x_A, Y_A)		(x_B, Y_B)
$r_{A1}, r_{A2} \in_R Z_n^*$		$r_{B1}, r_{B2} \in_R Z_n^*$
$T_{A1} = r_{A1}P \ , \ T_{A2} = r_{A2}P$		$T_{B1} = r_{B1}Y_A \ , \ T_{B2} = r_{B2}Y_A$
$T_A = r_{A1}Y_B + T_{A2}$		$T_B = r_{B1}Y_A + T_{B2}$
$s_A = x_A - x_{T_A}(r_{A1}k_{A1} + r_{A2}k_{A2}) \bmod n$		$s_B = x_B - x_{T_B}(r_{B1}k_{B1} + r_{B2}k_{B2}) \bmod n$
(x_{t_A} مولفه x نقطه t_A است)	$\xrightarrow{(T_{A1}, T_{A2}, s_A)}$	(x_{t_B} مولفه x نقطه t_B است)
	$\xleftarrow{(T_{B1}, T_{B2}, s_B)}$	
$T_B = x_A T_{B1} + T_{B2}$		$T_A = x_B T_{A1} + T_{A2}$
$Y_B \overset{?}{=} s_B P + x_{T_B}(k_{B1}T_{B1} + k_{B2}T_{B2})$		$Y_A \overset{?}{=} s_A P + x_{T_A}(k_{A1}T_{A1} + k_{A2}T_{A2})$
$K_1 = e(T_{B1}, T_{B2})^{r_{A1}} = e(P,P)^{r_{A1}r_{B1}r_{B2}}$		$K_1 = e(T_{A1}, T_{B2})^{r_{B1}} = e(P,P)^{r_{A1}r_{B1}r_{B2}}$
$K_2 = e(T_{B1}, T_{B2})^{r_{A2}} = e(P,P)^{r_{A2}r_{B1}r_{B2}}$		$K_2 = e(T_{B1}, T_{A2})^{r_{B2}} = e(P,P)^{r_{A2}r_{B1}r_{B2}}$
$K_3 = e(T_{B1}, T_{A2})^{r_{A1}} = e(P,P)^{r_{A1}r_{A2}r_{B1}}$		$K_3 = e(T_{A1}, T_{A2})^{r_{B1}} = e(P,P)^{r_{A1}r_{A2}r_{B1}}$
$K_4 = e(T_{A1}, T_{B2})^{r_{A2}} = e(P,P)^{r_{A1}r_{A2}r_{B2}}$		$K_4 = e(T_{A1}, T_{A2})^{r_{B2}} = e(P,P)^{r_{A1}r_{A2}r_{B2}}$

پروتکل(۴-۱): پروتکل پیشنهادی

$$K_1 = e(T_{B1}, T_{B2})^{r_{A1}} = e(P,P)^{r_{A1}r_{B1}r_{B2}}$$
$$K_2 = e(T_{B1}, T_{B2})^{r_{A2}} = e(P,P)^{r_{A2}r_{B1}r_{B2}}$$
$$K_3 = e(T_{B1}, T_{A2})^{r_{A1}} = e(P,P)^{r_{A1}r_{A2}r_{B1}}$$
$$K_4 = e(T_{A1}, T_{B2})^{r_{A2}} = e(P,P)^{r_{A1}r_{A2}r_{B2}}$$

- B نیز به محض دریافت مقادیر (T_{A1}, T_{A2}, s_A) ابتدا صحت امضا را بصورت زیر بررسی می‌کند

$$T_A = x_B T_{A1} + T_{A2}$$
$$Y_A \overset{?}{=} s_A P + (x_{T_A} \bmod n)(k_{A1}T_{A1} + k_{A2}T_{A2}) \qquad (۴-۲)$$

در صورتی که تساوی(۴-۲) برقرار نباشد B اجرای پروتکل را رها می‌کند و درغیر اینصورت کلیدهای نشست را بصورت زیر محاسبه می‌کند

$$K_1 = e(T_{A1}, T_{B2})^{r_{B1}} = e(P,P)^{r_{A1}r_{B1}r_{B2}}$$
$$K_2 = e(T_{B1}, T_{A2})^{r_{B2}} = e(P,P)^{r_{A2}r_{B1}r_{B2}}$$
$$K_3 = e(T_{A1}, T_{A2})^{r_{B1}} = e(P,P)^{r_{A1}r_{A2}r_{B1}}$$
$$K_4 = e(T_{A1}, T_{A2})^{r_{B2}} = e(P,P)^{r_{A1}r_{A2}r_{B2}}$$

جزئیات رابطه امضای مورد استفاده در طرح پیشنهادی را بصورت زیر می‌توان بررسی کرد:

$$\begin{aligned}
T_A &= x_B T_{A1} + T_{A2} \\
Y_A &= s_A P + x_{T_A}(k_{A1}T_{A1} + k_{A2}T_{A2}) \\
&= \left(x_A - x_{T_A}(k_{A1}r_{A1} + k_{A2}r_{A2})\right)P + x_{t_A}(k_{A1}T_{A1} + k_{A2}T_{A2}) \\
&= x_A P - x_{T_A}k_{A1}r_{A1}P - x_{x_{T_A}}k_{A2}r_{A2}P + x_{x_{T_A}}(k_{A1}T_{A1} + k_{A2}T_{A2}) \\
&= Y_A - x_{x_{T_A}}k_{A1}T_{A1} - x_{x_{T_A}}k_{A2}T_{A2} + x_{x_{T_A}}k_{A1}T_{A1} + x_{x_{T_A}}k_{A2}T_{A2} \\
&= Y_A
\end{aligned}$$

۴-۲ تحلیل امنیتی پروتکل پیشنهادی

امنیت کلید شناخته شده: در حمله کلید شناخته شده مهاجم با در اختیار داشتن کلیدهای نشست قبل قصد دارد امنیت نشست‌های بعدی را به خطر اندازد. در پروتکل(۴-۱) مهاجم با در اختیار داشتن کلیدهای نشست، هیچ اطلاعات مفیدی برای تهدید امنیت نشست‌های بعدی بدست نخواهد آورد. زیرا برای ساخت کلید نشست از پارامترهای موقت r_{A1} و r_{A2} استفاده شده است که در هر نشست تغییر می‌کنند. بنابراین پروتکل(۴-۱) دارای امنیت کلید شناخته شده است.

امنیت کلید ناشناخته: در حمله کلید ناشناخته مهاجم فعال C قصد دارد بگونه‌ای در روند اجرای پروتکل بین A و B دخالت کند که در پایان، B تصور داشته باشد با C نشست انجام داده درحالیکه A بر این باور است که با B توافق کلید انجام داده است. مهاجم C درصورت اعمال چنین حمله‌ای بر روی پروتکل(۴-۱)، پیام ارسالی (T_{A1}, T_{A2}, s_A) از طرف A را سد کرده و قصد دارد مقادیر (T_{A1}, T_{A2}) را با استفاده از کلید خصوصی خود امضا می-

کند. اما برای چنین کاری، مقادیر تصادفی r_{A1} و r_{A2} را باید در اختیار داشته باشد تا بتواند رابطه امضا را بصورت

$$T_A = \underline{r_{A1}} Y_B + T_{A2}$$
$$s_C = x_C - x_{T_A}\left(\underline{r_{A1}} k_{A1} + \underline{r_{A2}} k_{A2}\right)$$

محاسبه کند. همانطور که در رابطه بالا مشاهده می‌شود مقادیر تصادفی r_{A1} و r_{A2} در محاسبه امضا نقش دارند. از آنجایی که این مقادیر کلیدهای خصوصی کوتاه‌مدت کاربر A هستند بنابراین مهاجم C به آنها دسترسی ندارد. با توجه به این توضیحات نتیجه می‌گیریم که پروتکل پیشنهادی در برابر حمله کلید ناشناخته مقاوم است.

حمله جعل هویت با کلید آشکار شده: در این حمله مهاجم فعال C با در اختیار داشتن کلید خصوصی طولانی‌مدت A تلاش می‌کند خود را بجای B به A معرفی کند. در پروتکل(۴-۱) اگر مهاجم کلید خصوصی طولانی‌مدت x_A را در اختیار داشته باشد برای اینکه بتواند خود را بجای B به A معرفی کند باید قادر باشد امضای

$$T_B = r_{B1} Y_A + T_{B2}$$
$$s_B = x_B - x_{T_B}\left(r_{B1} k_{B1} + r_{B2} k_{B2}\right)$$

را به ازای (T_{B1}, T_{B2}) دلخواه محاسبه کند، درحالیکه بدلیل نداشتن کلید خصوصی x_B قادر به محاسبه این مقدار نخواهد بود. پس پروتکل پیشنهادی در برابر حمله جعل هویت با کلید آشکار شده امن است.

امنیت پیشرو کامل: این ویژگی امنیتی بیان می‌دارد که درصورت آشکار شدن کلیدهای خصوصی طولانی‌مدت طرفین پروتکل، امنیت کلیدهای نشست قبلی به خطر نیفتد. در پروتکل(۴-۱) اگر مهاجم کلیدهای خصوصی طولانی مدت x_A و x_B را در اختیار داشته باشد نمی‌تواند کلید نشست را محاسبه کند. زیرا برای محاسبه کلیدهای نشست

$$K_1 = e(P,P)^{r_{A1} r_{B1} r_{B2}} \quad , \quad K_2 = e(P,P)^{r_{A2} r_{B1} r_{B2}}$$
$$K_3 = e(P,P)^{r_{A1} r_{A2} r_{B1}} \quad , \quad K_4 = e(P,P)^{r_{A1} r_{A2} r_{B2}}$$

مهاجم مقادیر تصادفی r_{Ai} یا r_{Bi} را به ازای $i \in \{1,2\}$ باید در اختیار داشته باشد در حالیکه آنها را ندارد. بنابراین پروتکل پیشنهادی امنیت پیشرو کامل دارد.

۳-۴ مقایسه پروتکل پیشنهادی و پروتکل LWW

همانطور که در بخش (۴-۲) توضیح داده شد، پروتکل پیشنهادی ما همه ویژگی‌های امنیتی مطلوب را دارد بنابراین از لحاظ امنیتی بهینه‌تر از طرح LWW است. زیرا در برابر حمله جعل هویت مقاوم است و امنیت پیشرو کامل دارد در حالیکه پروتکل LWW در برابر حمله جعل هویت آسیب پذیر است و امنیت پیشرو کامل ندارد. در جدول(۴-۱) از لحاظ محاسبات مقایسه‌ای بین دو پروتکل انجام گرفته است.

همانطور که در جدول(۴-۱) مشاهده می‌شود تعداد ضرب‌های اسکالر در طرح پیشنهادی بیشتر از طرح LWW است. اما طرح پیشنهادی ما سه زوج‌سازی کمتر از طرح LWW دارد، از آنجایی که محاسبه یک زوج‌سازی تقریبا با ده ضرب اسکالر معادل است، بنابراین در مجموع حجم محاسبات در پروتکل پیشنهادی نسبت به طرح LWW کاهش چشم گیری دارد.

جدول۴-۱

	طرح LWW	طرح پیشنهادی
A	۶	۴
M	۱۰	۱۱
P	۷	۴

A : جمع نقاط خم‌بیضوی، B : ضرب اسکالر در نقطه خم‌بیضوی، P : زوج‌سازی،

۵- نتیجه گیری

در این مقاله به بررسی یک پروتکل توافق کلید چندتایی مبتنی بر زوج‌سازی پرداخته و یک حمله پیشنهادی به آن اعمال شد. سپس برای حل مشکل امنیتی آن یک پروتکل پیشنهادی ارائه و نشان داده شد که از لحاظ امنیتی و محاسباتی بهینه‌تر از طرح بررسی شده است.

نکته قابل توجه در پروتکل پیشنهادی این است که از زوج‌سازی فقط برای ساخت کلید استفاده شده است و طرح امضای مورد استفاده، با استفاده از خم‌بیضوی طراحی شده‌است. این امر سبب کاهش تعداد زوج‌سازی‌ها و افزایش امنیت پروتکل شده که در متن بطور مفصل توضیح داده شده است.

مراجع

[۱] سبزی‌نژادفراش، محمد، بررسی پروتکل‌های توافق کلید مبتنی بر خم‌بیضوی و زوج‌سازی‌های دوخطی و مقایسه‌ی آنها، پایان نامه کارشناسی ارشد، دانشگاه امام حسین(ع)، تهران، بهمن ۱۳۸۷.

[2] ANSI X9.42, Agreement of Symmetric Algorithm Keys Using Diffie–Hellman, Working Draft, May 1998.

[3] ANSI X 9.63, Elliptic Curve Key Agreement and Key Transport Protocols, Working Draft, July 1998.

[4] I.-F. Blake, G. Seroussi and N.-P. Smart. Advances Elliptic Curves in Cryptography. Cambridge University Press, 2005.

[5] S. Blake-Wilson, D. Johnson, and A. Menezes. Key agreement protocols and their security analysis. In Proc. of Sixth IMA International Conference on Cryptography and Coding, pages 30 – 45. Cirencester, UK, 1997.

[6] D. Boneh and M. Franklin, Identity-based encryption from the Weil pairing. In Advances in Cryptology - CRYPTO '01, LNCS 2139, pages 213-229, Springer-Verlag, 2001.

[7] W. Diffiee, M. Hellman. New Directions in Cryptography. In IEEE Transaction on Information Theory, IT-22 (6), pp. 644-654, 1976.
[8] M.C. Gorantla, R. Gangishetti, and A. Saxena, A Survey on ID-Based Cryptographic Primitives, In: Cryptology ePrint Archive: Report 2005/094, eprint.iacr.org/2005/094.pdf
[9] L. Harn, H.-Y. Lin, An authenticated key agreement protocol without using one-way function. In: Proceedings of eighth information security conference, Taiwan, May 1998; p. 155–60.
[10] L. Harn, H.-Y. Lin, Authenticated key agreement without using one-way hash function. Electron Lett 2001;37(10):629–30.
[11] IEEE P1363, Standards Specifications for Public-Key Cryptosystems, Working Draft, July 1998.
[12] Joux, A one-round protocol for tripartite Diffie-Hellman. In: Algorithmic Number Theory Symposium-ANTS-IV, pp. 385–394 Springer, Heidelberg, LNCS 1838 (2000)
[13] N.-Y. Lee, C.-N. Wu, C.-C. Wang, Authenticated multiple key exchange protocols based on elliptic curves and bilinear pairings, In: Computers and Electrical Engineering 34 (2008), pp. 12–20.
[14] A.J. Menezes, M. Qu, and S.A. Vanstone. Some key agreement protocols providing implicit authentication. In: Proceeding of the second workshop on selected areas in cryptography (SAC'95), 1995; pp. 22–32.
[15] V. Miller, The Weil pairing, and its efficient calculation, Journal of Cryptology, 17 (2004), 235-261.
[16] S.-M. Yen, M. Joye, Improved authenticated multiple-key agreement protocol, Electronics Letters 1998;34 (18):1738–1739
[17] T.-S. Wu, W.-H. He, C.-L. Hsu, Security of authenticated multiple-key, Electronics Letters 35 (5) (1999) 391–392.
[18] H.-S. Zhou, L. Fan and J.-H. Li, Remarks on unknown key-share attack on authenticated multiple-key agreement protocol, In Electronics Letters 2003;39 (17):1248–1249.