توضیحات پروتکل های ارتباطی امن پیام رسان

علی سالمی – محمد پناهی

پروتکل ما از دو بخش کلی تقسیم شده است. بخش اول مربوط به پروتکل تبادل کلید و بخش دوم مربوط به تصدیق اصالت است. در این پروتکل C معادل Client، S معادل Server، NonceA و NonceB معادل ۲ عدد تصادفی بزرگ، KS معادل کلید نشست، KUX معادل کلید عمومی X و KPX معادل کلید خصوصی X است.

در **بخش اول** پروتکل، کلید session با استفاده از رمز نگاری نا متقارن تبادل می شود.

$C \rightarrow S: \{NonceA\}_{KUS}, KUC$

$S \rightarrow C: \{NonceA + ۱\}_{KUC}, \{NonceB\}_{KUC}, sign(NonceB, KUC)_{KPS}$

$C \rightarrow S: \{NonceB + ۱\}_{KUS}$

$S \rightarrow C: \{KS\}_{KUC}$

برای امن شدن این پروتکل در برابر حملات شناخته شده ای مانند replay، forgery و MITM از تکنیک nonce و رمز نگاری نا متقارن استفاده شده است. در مرحله دوم nonce ها به همراه امضای hash کلید عمومی کاربر و nonce فرستاده می شود. بدین ترتیب اثبات می شود که پیام توسط سرور ساخته شده و کلید عمومی کاربر بدون تغییر به سرور رسیده است. در نتیجه مهاجم نمی تواند خود را به جای سرور قرار دهد یا حمله MITM را اجرا کرده و کلید نشست را پیدا کند. در مرحله سوم کاربر به سرور اثبات می کند که کلید عمومی داده شده در مرحله قبل مربوط به خودش است.

در نهایت بعد از تصدیق اصالت شدن سرور و اطمینان از اصالت کلید کاربر و نبودن حمله MITM کلید نشست توسط سرور تولید شده و به کاربر داده می شود.

در **بخش دوم** پروتکل کاربر تصدیق اصالت می شود. ارتباطات این بخش با استفاده از کلید نشست که در بخش قبل به دست آمده انجام می شود. برای ورود کاربر از پروتکل زیر استفاده می شود.

$C \rightarrow S: \{username, MD5(password)[۰: ۴۰]\}_{KS}$

$S \rightarrow C: \{JWT\}_{KS}$

کاربر با ارسال نام کاربری و کلمه عبور خود وارد سیستم می شود. برای جلوگیری از انتخاب کلمه عبور ساده از hash چیزی که کاربر وارد می کند به عنوان کلمه عبور استفاده می شود. این کار مانع فرآیند های امنیتی روی سرور نیست و در سرور نیز فرآیند های hash کردن و salt اضافه کردن و ... انجام خواهد شد.

در صورت موفقیت آمیز بودن، سرور یک توکن JWT تولید کرده و برای کاربر ارسال می کند. JWT یک عبارت سه بخشی است که به صورت Base۶۴ نوشته شده است. در قسمت اول header های لازم نوشته شده است. در قسمت دوم payload قرار می گیرد و در قسمت سوم امضای قسمت اول و دوم نوشته می شود. بدین ترتیب توکن غیر قابل تغییر می شود. payload شامل اطلاعاتی مانند id کاربر، تاریخ انقضاء و ... است.

کاربر در هر درخواست این توکن را ارسال می کند و بدین ترتیب اصالت کاربر تصدیق می شود. حتی کاربر می تواند این توکن را ذخیره کرده و در ارتباطات بعدی بدون نیاز به وارد کردن نام کاربری و کلمه عبور تصدیق اصالت شود.