Journal of Soft Computing and Information Technology (JSCIT)
Babol Noshirvani University of Technology, Babol, Iran
Journal Homepage: jscit.nit.ac.ir
Volume 8, Number 1, Spring 2018, pp. 14-23
Received: 12/30/2016, Revised: 01/14/2019, Accepted: 01/30/2019

SAML Standard Optimization for Use on CoAP-Based Web Servers on Internet of Things

Nima Mollaei[1], Hossein Shirazi[2*] and Alireza Pourebrahimi[3]
1- Tehran Jonoub Branch, Islamic Azad University, Tehran, Iran.
2*- Malek Ashtar Industrial University, Tehran, Iran.
3- Tehran Jonoub Branch, Islamic Azad University, Tehran, Iran.
[1]Nima.Mollaei@yahoo.com.com, [2*]Shirazi@mut.ac.ir, and [3]Poorebrahimi@gmail.com

Corresponding author address: Hossein Shirazi, Malek Ashtar Industrial University, Tehran, Iran, Post Code : 47148 – 71167.

Abstract- The use of web services has been increased by IoT technology development and increasing interoperability between objects. IoT web services access control is a challenging problem in IoT environment. Many standards such as SAML has been proposed for authorization and access control in common web services, but it is not possible to use these standards in IoT web services, because IoT resources has constraints in network, memory and process. This article proposed a modified version of SAML standard for using in IoT web services. In proposed changes, CoAP protocol has been chosen as application layer protocol, in order to reduce process time and memory consumption the JSON format has been used, and CBOR has also been used to reduce network traffic. COSE is also used to enhance the security of messages exchange between servers. In comparison of SAML standard, optimized SAML is more appropriate for IoT Web services because of low memory usage and processing time, and consequently, the reduction of the time for providing access in IoT environments.

Keywords- IoT, Access Control, SAML Standard, Modified SAML, CBOR, COSE.

مجله علمی پژوهشی رایانش نرم و فن‌آوری اطلاعات

دانشگاه صنعتی نوشیروانی بابل

صفحه مجله: jscit.nit.ac.ir

جلد ۸، شماره ۱، بهار ۱۳۹۸، صفحه ۲۳-۱۴

دریافت: ۱۳۹۵/۱۰/۱۰، بازنگری: ۱۳۹۷/۱۰/۲۴، پذیرش: ۱۳۹۷/۱۱/۱۰

بهینه‌سازی استاندارد SAML برای استفاده در وب‌سرویس‌های مبتنی بر CoAP در اینترنت اشیاء

نیما ملایی[۱]، حسین شیرازی*[۲]، علیرضا پورابراهیمی[۳]

۱- دانشگاه آزاد اسلامی، واحد تهران جنوب، تهران، ایران

*۲- دانشگاه صنعتی مالک اشتر، تهران، ایران،

۳- دانشگاه آزاد اسلامی، واحد تهران جنوب، تهران، ایران.

[1] Nima.Mollaei@yahoo.com, [2*]shirazi@mut.ac.ir, and [3]poorebrahimi@gmail.com

* نشانی نویسنده مسئول: حسین شیرازی، تهران، لویزان، دانشگاه صنعتی مالک اشتر

چکیده- امروزه با گسترش فن‌آوری اینترنت اشیاء و افزایش تعاملات بین اشیاء، استفاده از وب‌سرویس‌ها نیز رو به افزایش نهاده است. یکی از چالش‌های موجود در پیاده‌سازی وب‌سرویس‌های اینترنت اشیاء کنترل دسترسی در این محیط است. در وب‌سرویس‌های عمومی، استفاده از استانداردهایی مانند SAML برای تعیین مجوز و کنترل دسترسی متداول بوده اما در وب‌سرویس‌های اینترنت اشیاء به دلیل محدودیت در شبکه، حافظه و پردازش، امکان استفاده از این نوع استانداردها وجود ندارد. مقاله حاضر تلاش نموده است تا با انجام اصلاحاتی بر روی استاندارد SAML، آن را برای به‌کارگیری در وب‌سرویس‌های اینترنت اشیاء بهینه نماید. در تغییرات پیشنهادی، پروتکل CoAP به‌عنوان پروتکل لایه کاربرد انتخاب و به‌منظور کاهش حجم پردازش و حافظه، فرمت JSON ، به‌عنوان فرمت نگهداری داده‌ها استفاده گردیده است. برای کاهش بار شبکه نیز از CBOR بهره گرفته‌شده است. همچنین به‌منظور بالا بردن امنیت پیام‌های ارسالی بین سرویس‌دهنده‌ها از COSE استفاده‌شده است. مقایسه استاندارد SAML با نمونه اصلاح‌شده آن نشان داد که SAML تغییریافته به دلیل کاهش میزان حافظه موردنیاز و زمان پردازش و درنتیجه کاهش مدت‌زمان ارائه دسترسی برای محیط‌های محدود اینترنت اشیاء مناسب است.

واژه‌های کلیدی: اینترنت اشیاء، کنترل دسترسی، استاندارد SAML، CBOR، COSE

۱- مقدمه

به‌منظور پیاده‌سازی وب‌سرویس‌ها، معمولاً از دو معماری SOAP و REST استفاده می‌شوند. باوجودآنکه در طراحی‌های مبتنی بر SOAP، اطلاعات امنیتی در متن پیام گنجانده و امنیت مستقل از لایه‌های پایین‌تر (مانند لایه انتقال) است، اما به دلیل سنگینی استفاده از آن در اشیای دارای ظرفیت‌های محدود، عملاً به‌کارگیری آن در اینترنت اشیاء با مشکلاتی مواجه است [1]. ازاین‌رو روش‌های مبتنی بر REST موردتوجه بسیاری از محققان قرارگرفته است [2]. این روش از ویژگی بدون وضعیت[۱] برخوردار بوده لذا هیچ‌گونه نشستی بین سرویس‌دهنده و سرویس‌گیرنده شکل نگرفته و احراز هویت پایدار نیست. برای رفع این نقیصه از

روش‌هایی مانند HTTP Basic Authentication ، HTTP Digest Authentication استفاده‌شده است. در هر یک از آن‌ها مشکلات امنیتی مانند injection attack و middleware hijacking وجود داشته [3] یا بدون در نظر گرفتن محدودیت‌های حوزه اینترنت اشیاء ارائه گردیده است. ازاین‌رو می‌بایست از روشی استفاده شود تا مسئله محدودیت منابع در اینترنت اشیاء را نیز در نظر بگیرد. به‌منظور حل این مسئله، وب‌سرویس‌های مبتنی بر پروتکل CoAP[۲] ارائه شده است[4]. در این سرویس‌دهنده‌ها پروتکل CoAP به‌عنوان جایگزین HTTP در لایه کاربرد استفاده شده و از DTLS یا IPSec برای برقراری امنیت در لایه‌های پایین‌تر استفاده می‌گردد [5] .

امنیت فراهم‌شده توسط پروتکل CoAP شامل احراز هویت، رمزنگاری و یکپارچگی بوده و تعیین و کنترل دسترسی در این پروتکل وجود ندارد. علاوه بر این، در سرویس‌دهنده‌های مبتنی بر CoAP در مواردی که شبکه‌های ارتباطی از پروتکل‌های متفاوت استفاده می‌نمایند و گره‌های مبدأ و مقصد در شبکه‌های مجزا قرار می‌گیرند، برقراری ارتباط امن به دلیل ذخیره‌سازی بسته‌ها در گره‌های میانی مشکل می‌شود [6].

استاندارد SAML[۳] [7] ، از طریق امکان اشتراک اطلاعات احراز هویت به ایمن‌سازی سرویس‌دهنده‌ها کمک می‌نماید. در این استاندارد کاربر مجوزهای لازم را از یک مرکز دریافت نموده و از آن برای دسترسی به منابع سرویس‌دهنده استفاده می‌نماید. ازآنجاکه این استاندارد مبتنی بر تعدادی استاندارد دیگر مانند XML ، SOAP و HTTP است، استفاده از آن برای اشیایی که دارای محدودیت در پردازنده، حافظه و شبکه می‌باشند مقرون‌به‌صرفه نیست. ازجمله تلاش‌های صورت گرفته در خصوص انطباق استاندارد SAML با محیط‌های محدود می‌توان به [8] اشاره نمود. نویسندگان این مقاله تلاش نموده‌اند تا با قرار دادن JSON [۴] به‌جای XML در ساختار این استاندارد از آن در وب‌سرویس‌های RESTful برای محیط‌های دارای محدودیت پردازشی استفاده نماید. اما آنچه در این مقاله بدان توجه نشده چالش‌های استفاده از پروتکل HTTP در محیط محدود است.

در مقاله حاضر نیز تلاش شده تا با انجام تغییراتی در استاندارد SAML، آن را برای کنترل دسترسی در لایه کاربرد (پروتکل CoAP به‌عنوان یک پروتکل سرتاسری) در محیط اینترنت اشیاء بهینه نماید. در تغییرات پیشنهادی، علاوه بر استفاده از پروتکل CoAP در لایه کاربرد، به‌منظور کاهش بار پردازنده و حافظه از فرمت CBOR و روش رمزنگاری COSE برای امنیت محتوای بسته‌های مبادله شده استفاده گردیده است.

ساختار مقاله حاضر بدین‌صورت خواهد بود: در بخش بعد به بررسی موضوعات مرتبط با تعریف اینترنت اشیاء و پروتکل CoAP و همچنین کنترل دسترسی در وب‌سرویس‌های اینترنت اشیاء خواهیم پرداخت. در ادامه روش پیشنهادی برای اصلاح استاندارد SAML برای کنترل دسترسی در وب‌سرویس‌های اینترنت اشیاء ارائه‌شده و در پایان نیز استاندارد SAML با اصلاحات پیشنهادی مقایسه گردیده و نتایج آن ازنظر میزان حافظه موردنیاز، زمان پردازش و همچنین زمان ارائه مجوز دسترسی ارائه گردیده است.

۲- اینترنت اشیاء چیست؟

مؤسسات استاندارد و تحقیقات بسیاری وجود دارند که اقدام به ارائه تعاریف اینترنت اشیاء نموده‌اند، ازجمله آن‌ها می‌توان به انجمن مهندسان برق و الکترونیک (IEEE)، موسسه استانداردهای ارتباطی اروپا (ETSI)، آژانس‌های اختصاصی ملل متحد در حوزه فن‌آوری‌های اطلاعات و ارتباطات (ITU)، انجمن نیروی کار مهندسی اینترنت (IETF)، W3C، NIST و ... اشاره نمود.

شکل ۱: معماری ۳ لایه اینترنت اشیاء [9]

در گزارش اختصاصی که توسط انجمن مهندسی برق و الکترونیک (IEEE) در خصوص اینترنت اشیاء ارائه‌شده، این فن‌آوری را به‌صورت زیر تعریف نموده است[9] :

"شبکه‌ای از آیتم‌ها (که هر یک دارای حس‌گر تعبیه‌شده هست) که به اینترنت متصل است".

بر اساس معماری IEEE P2413، اینترنت اشیاء از ۳ لایه تشکیل شده است [9]: الف) لایه کاربرد، ب) لایه شبکه که برای انتقال داده استفاده شده است و ج) لایه حسگر که وظیفه جمع‌آوری داده از محیط را به عهده دارد (شکل ۱).

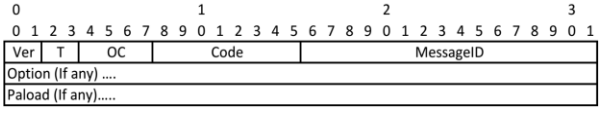
۲-۱- معرفی پروتکل CoAP

این پروتکل که یکی از پروتکل‌های لایه کاربرد در اینترنت اشیاء است، توسط گروه CoRE [۵] برای نرم‌افزارهای کاربردی اینترنت اشیاء ارائه گردیده است [5]. CoAP یک پروتکل انتقال وب مشابه HTTP و بر مبنای REST[۶] است.

برخلاف HTTP که از TCP برای ارتباط استفاده می‌نماید، پروتکل CoAP از UDP که برای کاربردهای اینترنت اشیاء، مناسب‌تر است استفاده می‌نماید[9] .

هدف CoAP توانا نمودن دستگاه‌های کوچک با قابلیت‌های پردازشی، حافظه و ارتباط ضعیف باهدف انجام تعاملات مبتنی بر RESTful است.

CoAP نیز همانند HTTP از متدهای PUT،GET، POST، و DELETE برای عملیات ایجاد، بازیابی، به‌روزرسانی و حذف استفاده می‌نماید. برای نمونه متد GET می‌تواند برای پرسیدن دما از کلاینت استفاده شود. کلاینت در صورت وجود، دما را برمی-گرداند در غیر این صورت با استفاده از کد وضعیتی که نشان‌دهنده عدم وجود داده درخواست شده است، به سرور پاسخ می‌دهد. CoAP از یک فرمت ساده و کوچک برای کد کردن پیام استفاده می‌کند. اولین قسمت (که ثابت نیز است) ۴ بایت سرآیند پیام است. و سپس مقدار توکن به طول صفر تا ۸ بایت می‌تواند اضافه شود. مقدار توکن به درخواست و پاسخ وابسته است. آپشن و سربار فیلدهای انتخابی بعدی است. یک پیام CoAP معمولاً بین ۱۰ تا ۲۰ بایت است. فرمت بسته CoAP در شکل ۲ نمایش داده‌شده است [5].



شکل ۲: ساختار بسته CoAP [5]

۲-۲- وب‌سرویس‌های اینترنت اشیاء

وب‌سرویس‌ها سیستم‌های نرم‌افزاری هستند که به‌وسیله آن‌ها تعامل ماشین-به-ماشین در سطح شبکه پشتیبانی می‌گردد[10] . درواقع وب‌سرویس، منبعی محاسباتی است که به‌منظور تسهیل همکاری بین سیستم‌های شبکه یا اینترنت، با استفاده از پروتکل‌های شبکه و استانداردهای کدگذاری متداول، در دسترس است.

به دلیل محدودیت منابع دستگاه‌ها، پیاده‌سازی وب‌سرویس‌ها برای تعامل نحوی اشیاء یکی از چالش‌های موجود در اینترنت اشیاء بوده است. این محدودیت‌ها، ایجاب می‌کند تا از پیام‌های کوتاه و ارتباطات سبک مبتنی بر رخداد استفاده شود. تحقیقات موجود نشان داده است که با استفاده از IPv6 و پروتکل‌هایی مانند 6LowPAN یا CoAP می‌توان وب‌سرویس‌هایی را برای استفاده در محیط‌های محدود پیاده‌سازی نمود[11] و [12].

۳- کنترل دسترسی در وب‌سرویس‌های اینترنت اشیاء

بر اساس تعریف موجود در RFC2828، کنترل دسترسی به‌صورت زیر تعریف می‌شود[13] :

"حفاظت از منابع سیستم در مقابل دسترسی‌های غیرمجاز. فرایندی که به‌وسیله آن استفاده از منابع سیستم بر اساس سیاست‌های امنیتی قاعده‌مند شده و اجازه دسترسی تنها به موجودیت‌ها (کاربران، برنامه‌ها، فرایندها یا سایر سیستم‌ها) بر اساس آن سیاست‌ها داده می‌شود."

یکی از توسعه‌های صورت گرفته در خصوص کنترل دسترسی استفاده از آن در سیستم‌های توزیع‌شده و پورتال‌های سازمانی است. نمونه آن را می‌توان در Google یا Facebook مشاهده نمود [14]. زمانی که شما در یکی از برنامه‌های کاربردی مبتنی بر این سایت‌ها وارد می‌شوید، برای استفاده از سایر برنامه‌ها از شما نام کاربری و رمز عبور پرسیده نمی‌شود. به این ایده ثبت‌نام منفرد[۷] (SSO) گفته می‌شود. مزیت اصلی استفاده از این روش کاربرپسند بودن آن است.

چندین راه‌حل برای پیاده‌سازی SSO ارائه شده است که می‌توان به SAML [7] ، OAuth [15] و OpenID [16] اشاره نمود.

تمرکز این مقاله بر روی استفاده از استاندارد SAML برای به‌کارگیری در سرویس‌دهنده‌های اینترنت اشیاء است.

۳-۱- کنترل دسترسی با استفاده از استاندارد SAML

استاندارد SAML در سال ۲۰۰۱ معرفی و در سال ۲۰۰۵ دومین نسخه از آن ارائه گردید. این استاندارد احراز هویت و تعیین مجوز را باهم انجام می‌دهد. در این استاندارد یک سرویس‌دهنده (SP) وجود دارد که درواقع وب‌سرویسی است که کاربر تلاش می‌کند تا به منابع آن دسترسی یابد. همچنین یک فراهم‌کننده مشخصه (IdP) نیز وجود دارد که مشخصات کاربران و مجوزهای آنان را نگهداری می‌نماید.

یکی دیگر از موجودیت‌های مورداستفاده در SAML، ادعا[۸] است که به‌صورت یک فایل XML نگهداری می‌شود، این فایل شامل عباراتی درباره احراز هویت، مجوز یا ویژگی‌های کاربر است.

۳-۲- معرفی فرمت‌های نگهداری و تبادل داده در وب‌سرویس‌ها

یکی از فرمت‌های معمول برای تبادل پیام در سطح لایه کاربرد، فرمت XML است. این فرمت که برای انسان و ماشین قابل‌درک است، دارای قابلیت فشرده‌سازی، امضا و رمزنگاری نیز هست. لذا از آن برای تبادل پیام‌ها در بین وب‌سرویس‌ها نیز استفاده می‌شود. (مانند استاندارد SAML که از این فرمت برای نگهداری و تبادل پیام استفاده می‌نماید) ازآنجاکه سرویس‌دهنده‌های اینترنت اشیاء دارای محدودیت منابع پردازشی و حافظه می‌باشند، یافتن فرمت‌های جایگزین می‌تواند به استفاده بهینه از منابع آن‌ها کمک شایانی نماید.

در زیر برخی از فرمت‌های مشابه که قابلیت جایگزینی با XML را دارند معرفی گردیده است.

۳-۳- فرمت JSON

فرمت JSON یک فرمت تبادل داده سبک است. این فرمت برای انسان قابل‌درک بوده و همچنین برای ماشین نیز قابل‌تجزیه و تولید است. فرمت JSON، دارای ساختار متنی بوده و مستقل از زبان است، اما زبان‌های برنامه‌نویسی بسیاری (مانند c, java, Perl , JavaScript, Paytoon و ...) از آن پشتیبانی می‌نمایند.

JSON بر مبنای قواعد مجموعه از زوج‌های "نام- مقدار" و همچنین یک لیست مرتب از مقادیر تعریف‌شده است.

۳-۴- مقایسه JSON و XML

معمولاً به دلیل سادگی JSON از آن به‌عنوان جایگزین XML استفاده می‌نمایند. هر دو زبان XML و JSON خود تعریف هستند. یعنی فهمیدن و خواندن آن‌ها برای انسان قابل‌درک است. هر دو از ساختار سلسله مراتبی برای ذخیره اطلاعات استفاده می‌کنند. هر دو فرمت توسط زبان‌های برنامه‌نویسی مختلف قابل‌خواندن هستند.

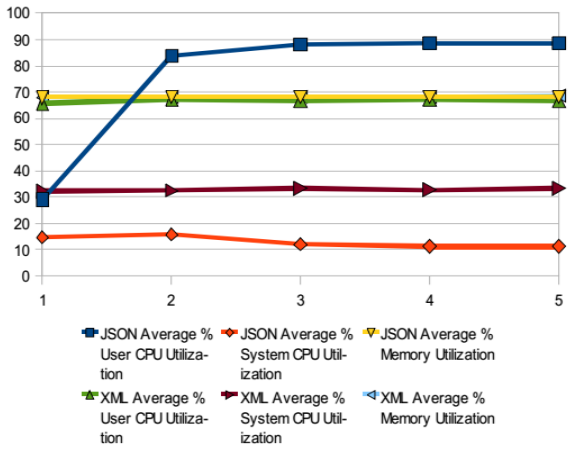
در کنار شباهت‌هایی که در بالا ذکر شد، این دو فرمت دارای اختلافاتی نیز هستند. در زیر به مهم‌ترین آن‌ها اشاره می‌نماییم.

اختلافات ظاهری JSON و XML

یکی از بارزترین اختلافات ظاهری بین JSON و XML، ساختار آن‌ها است. XML دارای ساختاری درختواره بوده درحالی‌که JSON از زوج مرتب "کلید-مقدار" در ساختار خود استفاده می‌نماید در JSON، برخلاف XML نمی‌توان توضیحات درج نمود.

تفاوت‌های کمی بین JSON و XML

عموماً JSON در مقایسه با XML برای انتقال داده از طریق شبکه سریع‌تر است. همچنین به‌صورت میانگین زمان تجزیه JSON حدود ۲۴ تا ۳۴ درصد سریع‌تر از XML، برآورد گردید[17] . بر اساس مطالعه موردی صورت گرفته در [18] درزمینه مقایسه فرمت‌های JSON و XML، میانگین بهره‌وری حافظه و پردازشگر در فرمت JSON بهتر از XML است(شکل ۳).



شکل ۳ : مقایسه بهره‌وری حافظه و پردازش در فرمت‌های JSON و XML [18]

۳-۵- معرفی فرمت‌های CBOR و COSE

فرمتCBOR [19] ، نمایش کدگذاری شده فرمت JSON به‌صورت خلاصه و کوتاه است. با این تفاوت که JSON دارای قالبی متنی و قابل‌خواندن بوده اما CBOR برای انسان قابل‌فهم نیست. ازآنجاکه یکی از ویژگی‌های این فرمت کوتاه بودن طول پیام است استفاده از آن را برای شبکه‌های محدود منطقی می‌نماید.

فرمت JSON دارای استانداردی پرکاربرد و بسیار ساده است. این استاندارد مبتنی بر متن است. ازآنجاکه کاربردهای اینترنت اشیاء، در تعاملات بین ماشین‌ها بوده و کمتر به دخالت انسان نیاز است، بهتر است که از استانداردهای باینری و غیر متنی استفاده شود. استانداردهای دیگری مانند BSON نیز برای کدگذاری متون JSON ارائه‌شده‌اند اما بسیاری از آن‌ها به دلیل سنگینی و پیچیدگی بالا قابل‌استفاده در محیط‌های محدود و اینترنت اشیاء نمی‌باشند.

ازآنجاکه فرمت‌های JSON و CBOR (بعد از خروج از حالت کدشده) به‌صورت متن ساده قابل‌مشاهده بوده، بنابراین امکان دست‌کاری داده‌های موجود متصور است. بنابراین استفاده از فرمتی که بتواند ویژگی‌های فرمت‌های یادشده را حفظ نموده و درعین‌حال محرمانگی و یکپارچگی را نیز به همراه داشته باشد، ضروری است.

فرمت COSE[۹] [20] با دارا بودن ویژگی‌های بالا، با استفاده از فرمت CBOR محتوا را بازنمایی نموده و از عملیات رمزنگاری بر روی محتوای JSON پشتیبانی می‌نماید. داده شامل یک شی CBOR با آرایه‌ای از بعدهایی از داده و محتوی رمزنگاری‌شده است.

- اولین بعد در این ارائه بنام فیلد حفاظت‌شده[۱۰] شناخته می‌شود. این فیلد شامل اطلاعاتی است که می‌بایست به‌وسیله فرایندهای رمزنگاری محافظت شود. این فیلد همیشه وجود دارد.
- بعد دوم، فیلد حفاظت نشده بوده و حاوی اطلاعاتی است که نیازی به محافظت ندارند.
- سومین بعد، محتوای پیام است. این محتوا نتیجه نوع داده و رمزنگاری استفاده‌شده، است.

۴- تغییرات پیشنهادی بر روی SAML برای کنترل دسترسی در وب‌سرویس‌های اینترنت اشیاء

در استاندارد SAML پیشنهاد می‌شود تا تصمیمات اصلی مجوزهای دسترسی در یک نود با محدودیت کمتر که بنام سرور احراز هویت شناخته‌شده، سپرده شود. به‌عبارت‌دیگر، انجام تعیین مجوز می‌بایست در محیط‌های مطمئنی که سرویس‌دهنده منبع با آن ارتباط دارد، انجام شود و بهتر است که تا حد ممکن به منبع نزدیک باشد.

به‌منظور انتقال تصمیمات تعیین دسترسی از سرور تعیین مجوز به دستگاهی که کنترل دسترسی در مورد آن انجام می‌شود، از موجودیتی بانام ادعای مجوز[۱۱] استفاده می‌نماید. نمونه‌های متعددی از ادعاهای مجوز برای استاندارد مشابه (استاندارد SAML) ارائه شده است.

به‌منظور نگهداری و تبادل ادعاهای SAML تولیدشده در سرورهای مشخصه و سرویس‌دهنده، از استاندارد XML استفاده می‌شود. در این مقاله پیشنهاد می‌گردد تا به‌جای فرمت XML از JSON به‌عنوان فرمت تبادل ادعاها استفاده گردد. مزیت استفاده از این فرمت، علاوه بر کم شدن حجم داده ارسالی، پردازش را در سرویس‌دهنده اینترنت اشیا کاهش می‌دهد.

به‌منظور کاهش بار ترافیکی شبکه، می‌توان بسته‌های تهیه‌شده با فرمت JSON را نیز کوتاه‌تر نمود. بدین معنی که با استفاده از CBOR می‌توان طول بسته‌های ایجادشده با فرمت JSON را کوتاه‌تر کرده و آن را از طریق شبکه ارسال نمود.

سایر مراحل همانند روش ارائه شده در نسخه استاندارد SMAL ویراش دوم [7] است. تفاوت‌های اساسی روش پیشنهادی با روش موجود در استاندارد SAML به شرح زیر است:

۱- جایگزینی پروتکل HTTP با پروتکل CoAP

۲- استفاده از فرمت JSON در ذخیره و پردازش داده‌ها

۳- استفاده از فرمت‌های CBOR و COSE برای تبادل پیام

۴-۱- پیاده‌سازی SAML اصلاح‌شده

برای پیاده‌سازی روش پیشنهادی، از زبان جاوا و از کتابخانه Californium [21] به‌عنوان کتابخانه پروتکل CoAP استفاده شده است. برای پیاده‌سازی این روش، ابتدا دو سرویس‌دهنده وب با

استفاده از پروتکل CoAP راه‌اندازی گردید. سپس یک کلاینت نیز با استفاده از این پروتکل در شبکه قرار گرفت.

قبل از هر چیز، ارائه‌دهنده سرویس مشخصه (IdP) یک جفت کلید عمومی-خصوص برای توزیع بین وب‌سرویس‌هایی که مشترک آن هستند، می‌سازد. چگونگی ساخت انتقال آن در حوزه این تحقیق نبوده لذا به‌صورت پیش‌فرض در ارائه‌دهنده مشخصه (IdP) و مشترکانش قرار گرفته است.

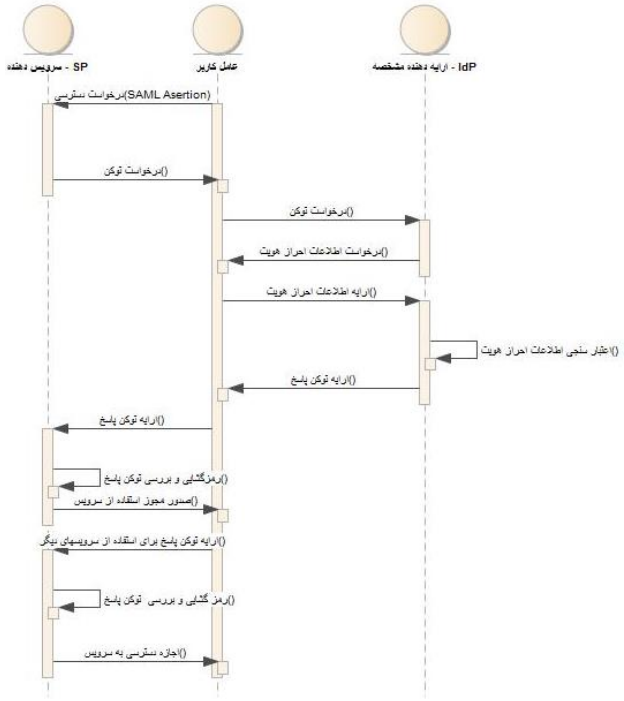
زمانی که کلاینت می‌خواهد با سرویس‌دهنده ارتباط برقرار نماید، با استفاده از کلید عمومی سرویس‌دهنده، درخواست خود را رمز می‌نماید. پیام ارسال‌شده که در قالب CBOR ارسال می‌شود، حاوی اطلاعاتی مانند: نام کلاینت، ادعای مجوز (پاسخ SAML که قبلاً توسط IdP برای کلاینت تخصیص داده شده است)، مشخصه کاربر و ... است (شکل ۴). در موارد زیر اجازه دسترسی به منابع موردنیاز کلاینت در سرویس‌دهنده فراهم نمی‌گردد:

۱- سرویس‌دهنده پارامترهای موردنیاز ادعای مجوز را پیدا نکند.
۲- نتواند ادعای مجوز دریافت شده را رمزگشایی نماید. (رمزگشایی در دو مقطع انجام می‌گیرد، یک مرحله رمزگشایی کل پیام و مرحله دوم، رمزگشایی بخش‌هایی از پیام که با استفاده از COSE رمز شده است)
۳- ادعای مجوز ، مربوط به کلاینت موردنظر نباشد.

Assertion
ClientID
Issuer
Subject
NotBefore
NorAfter
AccessScope
Signature

شکل ۴ : ساختار ادعای مجوز در اصلاحات پیشنهادی

درصورتی‌که سرویس‌دهنده تشخیص دهد که اجازه دسترسی به منبع درخواست شده برای کلاینت وجود ندارد، سرویس‌دهنده یک درخواست رمز شده (با استفاده از کلید عمومی ارائه‌دهنده مشخصه) حاوی مشخصه کاربر (یا برنامه کاربردی) درخواست‌کننده که به منبع نیاز دارد و منبع مورد درخواست را به‌منظور تعیین مجوز به ارائه‌دهنده مشخصه (IdP) ارسال می‌نماید. ارائه‌دهنده مشخصه، پس از دریافت درخواست، آن را با استفاده از کلید عمومی خود رمزگشایی نموده و سپس کاربر و منبع درخواست شده را ازنظر مجوزهای دسترسی بررسی می‌نماید. بعد از موفقیت در رمزگشایی و چک‌های ضروری، ارائه‌دهنده مشخصه، درخواست اطلاعات تکمیلی (در قالب توکن‌های دسترسی) از کاربر می‌نماید. پس از موفقیت در این مرحله، ارائه‌دهنده مشخصه، پاسخ مناسب را در قالب ادعای مجوز (بر اساس ساختار نمایش داده‌شده در شکل ۴) به سرویس‌دهنده برمی‌گرداند. این پاسخ با استفاده از کلید ارائه‌دهنده مشخصه، رمز شده است. مصرف‌کننده این پاسخ می‌تواند آن را برای دسترسی به سرویس‌دهنده‌های دیگری که از این ارائه‌دهنده مشخصه، استفاده می‌نمایند، به کارگیرد. نمودار توالی این روش در شکل ۵ ارائه شده است.

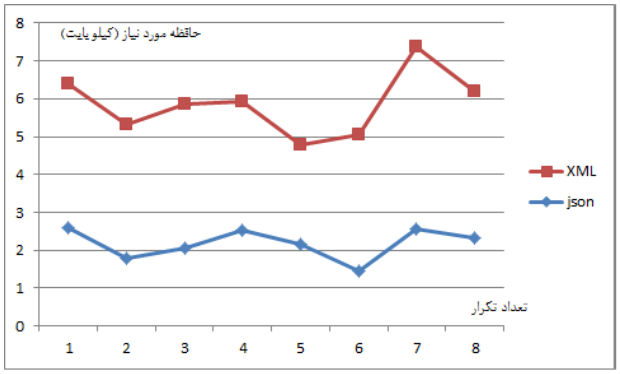


شکل ۵: نمودار توالی برای کنترل دسترسی در SAMLاصلاح‌شده

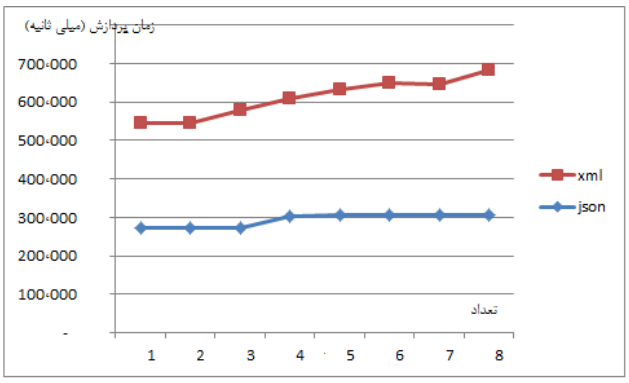
۵- بررسی نتایج و ارزیابی اصلاحات پیشنهادی

همان‌گونه که قبلاً اشاره گردید، تلاش این مقاله بر انجام تغییراتی بر روی استاندارد SAML باهدف به‌کارگیری آن در محیط‌های محدود اینترنت اشیاء است. لذا ارزیابی صورت گرفته بر روی به‌کارگیری این استاندارد در محیط‌هایی که ازنظر میزان حافظه و قدرت پردازش و شبکه دارای محدودیت می‌باشند، خواهد بود.

در این بخش در ابتدا به مقایسه میزان حافظه موردنیاز و مدت‌زمان پردازش بسته‌های ارسالی پرداخته و سپس به بررسی شبکه موردنیاز برای SAML اصلاح‌شده خواهیم پرداخت.



شکل ۶ : مقایسه میزان حافظه موردنیاز برحسب کیلوبایت برای پردازش بسته‌های ارسالی برحسب کیلوبایت



شکل ۷ : مقایسه مدت‌زمان پردازش بسته‌های ارسالی بین کاربر و ارائه‌دهنده مشخصه برحسب میلی‌ثانیه

۵-۱- بررسی میزان حافظه و مدت‌زمان پردازش

به‌منظور بررسی میزان حافظه مورداستفاده و همچنین مدت‌زمان پردازش بسته‌های مبادله شده از شبیه‌ساز " Californium (Cf) CoAP framework" [21] بر روی یک ماشین مجازی با 4GB حافظه و ۲ پردازشگر 2.5GHz و سیستم‌عامل Ubuntu 12 ، در دو حالت زیر استفاده شده است.

- در حالت اول بسته‌هایی با فرمت XML که با استفاده از پروتکل CoAP بین کاربر و ارائه‌دهنده مشخصه مبادله می‌گردد.
- درروش دوم از همان بستر استفاده نموده و از بسته‌هایی با فرمت JSON (معادل بسته‌های XML) استفاده شده است.

برای انجام مقایسه سناریوی زیر به تعداد ۵۰ بار تکرار و میانگین نتایج حاصله محاسبه گردید:

۱- تعداد ۸ فایل حاوی ادعاهای SAML با دو فرمت JSON و XML ایجاد می‌گردد.

۲- با استفاده از شبیه‌ساز Californium از سمت سرویس‌گیرنده به سرویس‌دهنده‌ها ارسال گردید.

۳- زمان پردازش و میزان حافظه مورداستفاده محاسبه گردید.

نتایج به‌دست‌آمده در شکل ۶ (میزان حافظه مورداستفاده برحسب بایت) و شکل ۷ (مدت‌زمان پردازش برحسب میلی‌ثانیه) نشان داده‌شده است. قابل‌ذکر است که علت رفتار زیگزاگی به دلیل انجام آزمون به تعداد زیاد و با طول‌های افزاینده و میانگین‌گیری نتایج حاصله است.

همان‌گونه که در شکل‌های ۶ و ۷ مشاهده می‌گردد، با انتخاب فرمت JSON در بسته‌های ارسالی، مدت‌زمان پردازش و همچنین میزان حافظه موردنیاز در مقایسه با ادعاهای با فرمت XML، کاهش قابل‌توجهی حاصل‌شده و این موضوع می‌تواند در شبکه‌های محدود اینترنت اشیاء بسیار حائز اهمیت باشد.

۵-۲- مقایسه زمان ارائه دسترسی

به‌منظور مقایسه زمان ارائه دسترسی برای استاندارد SAML و اصلاح‌شده آن، از طریق دو سناریوی زیر اقدام به آزمودن آن می‌نماییم:

۱- آزمودن کنترل دسترسی بر اساس روش پیشنهادی برای کلاینت‌هایی که دارای "مجوز دسترسی معتبر" می‌باشند.

۲- آزمودن کنترل دسترسی برای کلاینت‌هایی که هم‌اکنون فاقد "مجوز دسترسی معتبر" می‌باشند.

در زیر به تشریح هر یک از سناریوها و ارزیابی نتایج حاصله می‌پردازیم.

بدون در نظر گرفتن وضعیت کلاینت‌ها، متغیرهای زیر در ارزیابی مشارکت خواهند داشت:

T_{Send}: زمان موردنیاز برای انتقال پیام از سرویس‌گیرنده (SR_i) به سرویس‌دهنده (SP) و همچنین بین سرویس‌دهنده و ارائه‌دهنده مشخصه (IdP). در اینجا زمان‌های موردنیاز برابر فرض شده و از تأخیر شبکه صرف‌نظر می‌گردد.

$T_{Sp,Assert}$: زمان موردنیاز برای اعتبار سنجی مجوز دسترسی (توکن پاسخ) درخواست‌کننده سرویس در سرویس‌دهنده.

T_{Chlnge}: زمان موردنیاز برای دریافت اطلاعات هویتی از کاربر و اعتبار سنجی آن

N: حداکثر تعداد درخواست سرویسی که می‌تواند توسط یک درخواست‌کننده سرویسی (SR_i) ایجاد شود.

M: حداکثر تعداد درخواست‌کنندگان سرویس که می‌توانند برای یک منبع درخواست ایجاد نمایند.

T_{all}: مجموع کل زمان موردنیاز برای اعطای دسترسی به همه درخواست‌کنندگان سرویسی که می‌تواند تعریف شود.

TSR_i: مجموع زمان موردنیاز برای خدمت‌دهی به یک درخواست سرویس از یک سرویس‌دهنده SR_i

مجموع زمان موردنیاز برای ارائه سرویس به یک درخواست‌کننده به‌صورت زیر است:

$$TSR_i = ((6 \times T_{Send}) + T_{Sp,Assert} + T_{Chlnge} \quad (۱)$$

توضیح: ضریب۶ برای T_{Send} نشان‌دهنده تعداد دفعات ارسال پیام بین سرویس‌دهنده، ارائه‌دهنده مشخصه و سرویس‌گیرنده است.

مجموع زمان موردنیاز برای ارائه سرویس به کلیه درخواست‌های یک درخواست‌کننده:

$$T_{all} = ((6 \times T_{Send}) + T_{Sp,Assert} + T_{Chlnge} + (\mathrm{N}-1)(T_{Send} + T_{Sp,Assert}) \quad (۲)$$

$$T_{all} = (\mathrm{N}-5) + \mathrm{N} \times (T_{Send} + T_{Sp,Assert})$$

همان‌گونه که مشاهده می‌گردد، مجموع کل زمان موردنیاز برای اعطای دسترسی، به زمان انتقال پیام از سرویس‌گیرنده به سرویس‌دهنده و همچنین به مدت‌زمان پردازش بسته‌های ارسالی به سرویس‌دهنده وابسته است. که با توجه به نتایج ارائه‌شده در بخش قبل، می‌توان نتیجه گرفت که مدت‌زمان اعطای دسترسی به سرویس‌گیرنده در اینترنت اشیاء با اصلاح صورت گرفته در استاندارد SAML ، کاهش خواهد یافت.

درصورتی‌که چند درخواست‌کننده سرویس (M) از منابع یک سرویس تقاضای استفاده نمایند، مجموع زمان موردنیاز برای ارائه سرویس به کلیه درخواست‌ها به‌صورت زیر است:

$$\sum_{i=1}^{M} \left[(\mathrm{N}-5) + \mathrm{N} \times T_{Sp,Asset} + T_{Chlnge}\right] \quad (۳)$$

همان‌گونه که در فرمول (۳) مشاهده می‌گردد، مجموع زمان موردنیاز برای ارائه مجوز به یک درخواست‌کننده به تعداد سرویس‌دهنده‌ها رابطه مستقیم داشته و به‌ازای هر سرویس‌دهنده نیز مقدار ثابت سربار شبکه ایجاد می‌گردد.

بعلاوه بر اساس فرمول (۳)، یکی از پارامترهای مؤثر بر روی زمان کلی ارائه مجوز، $T_{Sp,Asset}$ است که با توجه به جایگزینی فرمت JSON به‌جای XML زمان موردنیاز برای پردازش پیام‌های دریافتی در سرویس‌دهنده و سرویس‌گیرنده به میزان قابل‌توجهی کاهش خواهد یافت.

از سوی دیگر استفاده از فرمت کد شده CBOR یا فرمت COSE برای تبادل پیام بین سرویس‌دهنده و ارائه‌دهنده مشخصه، به دلیل حجم پایین آن‌ها نسبت به حجم پیام‌های با فرمت XML، حجم بسیار پایینی از ترافیک شبکه را به خود اختصاص خواهد داد.

نکته مهم و قابل‌توجه در استفاده از SAML اصلاح‌شده، محدودیت‌های به وجود آمده بر روی پروتکل SAML به دلیل جایگزین پروتکل HTTP و فرمت XML و همچنین روش کدگذاری و رمزنگاری است. این موضوع باعث کاهش طول بسته ادعای مبادله شده و همچنین به‌کارگیری الگوریتم‌های رمزنگاری محدود و سبک‌تر بجای الگوریتم‌های پیچیده مورداستفاده در XML، می‌گردد. همچنین همان‌گونه که در مقدمه مقاله نیز به آن اشاره گردید، استفاده از پروتکل CoAP در لایه کاربرد باعث الزام استفاده از IPSec یا DTLS در لایه‌های پایین‌تر می‌گردد. این مسائل علیرغم انطباق با محدودیت‌های موجود در اینترنت اشیاء، می‌بایست در زمان به‌کارگیری آن موردتوجه استفاده‌کنندگان قرار گیرد.

۶- نتیجه‌گیری

در این مقاله تلاش گردیده تا با انجام اصلاحاتی بر روی استاندارد SAML، از آن در ارائه دسترسی در وب‌سرویس‌های محدود اینترنت اشیاء که با پروتکل CoAP پیاده‌سازی شده‌اند، استفاده نماید. برای رسیدن به این هدف علاوه بر تغییر در پروتکل لایه کاربرد از HTTP به CoAP، و جایگزینی JSON بجایXML از فرمت COSE نیز استفاده گردید. به‌منظور بررسی عملکرد نمونه تغییریافته‌ی SAML ، آن را با استاندارد SAML پایه ازنظر میزان حافظه موردنیاز و مدت‌زمان پردازش و درنهایت ازنظر مدت‌زمان اعطای دسترسی مورد مقایسه قرار دادیم. نتیجه آزمون صورت گرفته بهبود عملکرد در پارامترهای زمان پردازش و میزان حافظه مورد نیاز و همچنین مدت‌زمان ارائه دسترسی را نشان داد.

مراجع

[1] T Aihkisalo, "Latencies of Service Invocation and Processing of the REST and SOAP Web Service Interfaces," in *IEEE Eighth World Congress on Services*, 2012, pp. 100-107.

[2] Hiro Gabriel, Cerqueira Ferreira, and Edna Dias Canedo, "IoT Architecture to Enable Intercommunication Through REST API a nd U PnP U sing IP, ZigBee a nd Arduino," in *1 st International Workshop on Internet of Things Communications and Technologies (IoT'13)*, 2013, pp. 53 – 60.

[3] Jo. Juyeon, Kim. Yoohwan, and Lee Sungchul, "Mindmetrics: Identifying users without their login IDs," in *IEEE International Conference on Systems, Man and Cybernetics (SMC)*, 2014, pp. 2121 - 2126.

[4] Matthias Kovatsch, Martin Lanter, and Zach Shelby, "Californium: Scalable Cloud Services for the Internet of Things with CoAP," in *4th International Conference on the Internet of Things (IoT 2014)*, 2014.

[5] Z. Shelby, K. Hartke, C. Bormann, and B. Frank, "Constrained Application Protocol (CoAP). draft-ietf-core-coap-12," 2012.

[6] Mahdi Aiash, "Security Analysis of the Constrained Application Protocol in the Internet of Things," , 2013.

[7] OASIS. (2008, March) OASIS. [Online]. www.oasis-open.org/committees/download.php/27819/sstc-saml-tech-overview-2.0-cd-02.pdf

[8] M. Ali, T. S. Sobh, and S. EL-Gamal, "Identity Management: Lightweight SAML for Less Processing Power," *I.J. Information Technology and Computer Science*, pp. 42-49, 2015.

[9] IEEE Internet Initiative, "Towards a definition of the Internet of Things (IoT)," 2015.

[10] D. Booth et al. (2004) Web Services Architecture. [Online]. http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/

[11] G Moritz, F. Golatowski, and D Timmermann, "A Lightweight SOAP over CoAP Transport Binding for Resource Constraint Networks," in *Eighth IEEE International Conference on Mobile Ad-Hoc and Sensor Systems*, 2011, pp. 861 - 867.

[12] B. Negasha, A. Rahmani, T Westerlunda, P. Liljeberg, and H. Tenhunena, "LISA: Lightweight Internet of Things Service Bus Architecture ," in *The 6th International Conference on Ambient Systems, Networks and Technologies* , 2015.

[13] R. Shirey. (2010) Network Working Group. [Online]. https://www.ietf.org/rfc/rfc2828.txt

[14] Yinzhi Cao et al., "Protecting Web-Based Single Sign-on Protocols against Relying Party Impersonation Attacks through a Dedicated Bi-

directional Authenticated Secure Channel," in *Research in Attacks, Intrusions and Defenses*.: Springer International Publishing , 2014, pp. 276-298.

[15] D. Hardt. (2012) Internet Engineering Task Force (IETF). [Online]. https://datatracker.ietf.org/doc/rfc6749/

[16] Kohlar.F and Schwenk.J Sovis.P, "Security analysis of OpenID," in *Proceedings of the Securing Electronic Business Processes-Highlights of the Information Security Solutions Europe 2010 Conference*, 2010.

[17] P. Wang, X. Wu, and H. Yang, "Analysis of the efficiency of data transmission format based on Ajax applications," in *nternational Conference of Information Technology, Computer Engineering and Management Sciences*, 2011, pp. 265-268.

[18] N. Nurseitov, M. Paulson, and P. Reynol, "Comparison of JSON and XML Data Interchange Formats: A Case Study," in *Computers and Their Applications in Industry and Engineering*, 2009.

[19] C. Bormann and P. Hoffman. (2013) RFC 7049 Internet Engineering Task Force. [Online]. https:lltools.ietf.org/html/rfc7049

[20] J. Schaad. (2016, Jan.) CBOR Object Signing and Encryption (COSE). [Online]. https://tools.ietf.org/id/draft-ietf-cose-msg-24.txt

[21] The Eclipse Foundation. (2014) Californium (Cf) CoAP framework. [Online]. https://www.eclipse.org/californium/

زیرنویس‌ها:

[1] StateLess

[2] Constrained Application Protocol

[3] Security Assertion Markup Language

[4] JavaScript Object Notation

[5] IETF Constrained RESTful Environments (CoRE) working group

[6] Representational State Transfer

[7] single sign-on (SSO)

[8] Assertion

[9] CBOR Object Signing and Encryptio

[10] the protected field

[11] Authotization Assertion