|  |  |  |
| --- | --- | --- |
|  | دانشکده مهندسی کامپیوتر و فناوری اطلاعات | F:\Arm\AKUT.svg.png |

آزمایشگاه اینترنت اشیاء

گروه پلتفرم

**عنوان سند:**

**تحلیل نیازمندی­های کارکردی پلتفرم اینترنت اشیا**

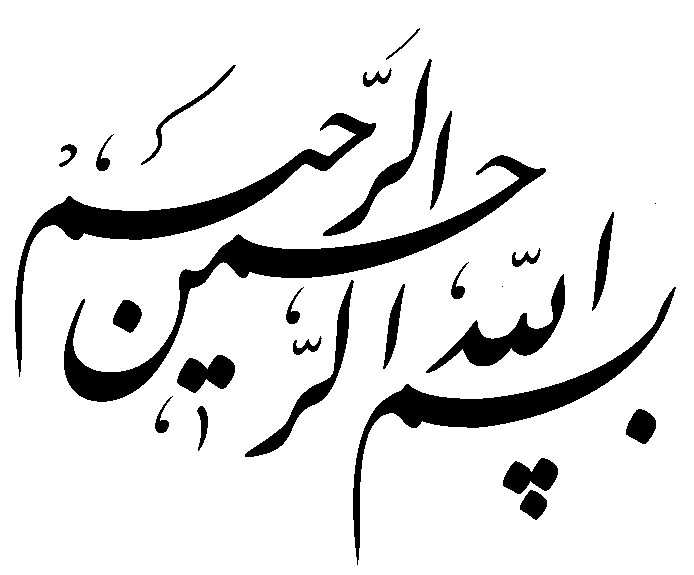
**(پروتکل­های ارتباطی با اشیا- LAN)**

**كد سند:**

**IoT-IMP-LAN-v1.0**

**تاريخ:**

**14/01/97**



اطلاعات سند

|  |  |
| --- | --- |
| نام پروژه: | طراحي و پياده‌سازي پلتفرم اينترنت اشياء |
| نام سند: | تحلیل نیازمندی­های کارکردی پلتفرم اینترنت اشیا (پروتکل­های ارتباطی با اشیا- LAN) |
| کد سند: | IoT-IMP-LAN-v1.0 |
| وضعيت: | نهایی |
| تاريخ انتشار نهايي: | --- |
| نوع طبقه‌بندي سند: | محرمانه |

تاریخچه بازبینی‌ها

|  |  |  |  |
| --- | --- | --- | --- |
| ردیف | نام بازبینی کننده | تاریخ بازبینی | تغييرات |
| 1 | تيم فني | 10/11/96 | بر اساس نظرات کارفرما در جلسه 7/11/96 |
| 2 | تیم فنی | 14/01/97 | ویرایش ادبی و فنی |
| 3 |  |  |  |
| 4 |  |  |  |

چکيده

این سند در راستای شناسایی نیازمندی­ها و طراحی پروتکل LAN برای اتصال اشیا، تنظیم و گردآوری شده است. با توجه به نیاز کارفرما علاوه بر پروتکل ارتباطی LoRaWAN، پیاده­سازی پروتکل LAN نیز جز مواردی است که در توسعه پلتفرم اینترنت اشیا در نظر گرفته شده است.

فهرست مطالب

[فصل 1: طراحی پروتکل LAN 6](#_Toc510015075)

[1-1- مقدمه 6](#_Toc510015076)

[1-2- پروتکل پیشنهادی 7](#_Toc510015077)

[1-3- پوشش نیازمندی­ها 9](#_Toc510015078)

فهرست جداول

[جدول (1-1) پوشش نیازمندی­های پروتکل LAN 9](#_Toc510015928)

1. طراحی پروتکل LAN
   1. مقدمه

پلتفرم اینترنت اشیاء قابلیت اتصال اشیاء از طریق پروتکل‌ LAN (خانواده IEEE 802 و مشخصاIEEE 802.3 ، IEEE 802.11 و ...) را فراهم می‌کند. برای این منظور، پروتکل لایه کاربرد پیشنهادی در این مستند دو دسته­ی کلی نیازمندی­ها به شرح زیر را برآورده می‌کند:

1. تبادل داده
   1. ارسال داده از اشیاء به پلتفرم (Up-Link)
   2. ارسال داده از پلتفرم به اشیاء (Down-Link)
2. امنیت
   1. احراز هویت پلتفرم توسط اشیاء (Platform Authentication)
   2. احراز هویت اشیاء توسط پلتفرم (Thing Authentication)
   3. محرمانگی (Confidentiality)
   4. تمامیت (Integrity)

لازم به ذکر است که این پروتکل علاوه بر پلتفرم بر روی اشیاء نیز باید پیاده‌سازی گردد، بنابراین استفاده از پروتکل‌های استاندارد موجود برای این منظور (در عمل) الزامی است. در ادامه این سند، در ابتدا پروتکل پیشنهادی شرح داده شده و سپس نحوه تامین نیازمندی‌ها مشخص خواهد شد.

* 1. پروتکل پیشنهادی

پروتکل پیشنهادی از سه مرحله اصلی تشکیل می‌گردد:

1. برقراری نشست (Session Establishment)
2. احراز هویت اشیاء (Thing Authentication)
3. تبادل داده (Data Communication)

هر یک از این مراحل در ادامه این بخش تشریح می‌گردد.

* **برقراری نشست**

در مرحله اول، با درخواست اشیاء برای اتصال به پلتفرم یک تونل SSL/TLS مابین شی و پلتفرم برقرار می‌گردد. در ایجاد این تونل موارد زیر در نظر گرفته شده است:

1. الگوریتم‌های رمزنگاری متقارن (برای رمز داده) و الگوریتم‌های رمزنگاری نامتقارن (برای تبادل کلید) الگوریتم‌هایی است که در RFC 5246 و RFCهاي مرتبط مشخص شده است.
2. از گزینه اختیاری “Client Certificate Request” استفاده نشده است بنابراین اشیاء نیاز به داشتن Certificate ندارند که برای توسعه دهندگان اشیاء مزیت محسوب می‌شود.
3. در زمان استفاده از این پلتفرم در محیط واقعی، Certificate استفاده شده برای پلتفرم باید توسط یک CA امضاء شده باشد که اشیاء یک کپی از Certificate آن CA را داشته باشند یا به نحوی امکان صحت‌سنجی آن Certificate توسط اشیاء (مثلا زيرساخت PKI) امکان‌پذیر باشد كه جزييات آن خارج از محدوده اين پروتكل پيشنهادي است.

* **احراز هویت اشیاء**

بعد از ایجاد تونل TLS، مرحله دوم این پروتکل اجرا می‌شود که در آن، هویت اشیاء توسط پلتفرم مورد تایید قرار می‌گیرد. برای احراز هویت اشیاء، در این پروتکل از jwt استفاده می‌شود که یک روش مبتنی بر token می‌باشد كه هم در پلتفرم و هم در شي پيكربندي مي‌شود. شی می‌بایست از این token در تمام ارتباط‌های بعدی خود استفاده کند. همانگونه که ذکر شد، این فرایند بعد از ایجاد تونل TLS و با استفاده از پروتکل HTTP (به عبارت دیگر بر روی پروتکل https) انجام می‌گیرد. در ساخت این token از مقدار hash آدرس IP شی استفاده می‌شود. با توجه به این که هر شی در پلتفرم می‌بایست توسط یک شناسه یکتا مشخص شود، پیشنهاد می‌شود برای اشیایی که توسط این پروتکل به پلتفرم متصل میشوند، اشیا توسط آدرس IP و آدرس سخت افزاری MAC به صورت منحصر به فرد مشخص ‌گردد.

در صورت احراز هویت موفق شی، طی یک سری پیغام handshake، شي به يك کانال socket.io مشخص شده توسط پلتفرم، subscribe مي‌كند. جهت تامين امنيت اين بخش، ارتباطات socket.io نيز بر روي تونل TLS ارسال مي‌گردد (در حقيقت از مكانيزم wss استفاده مي‌شود)

* **تبادل داده: ارسال داده‌ها از اشیاء به پلتفرم**

برای ارسال داده‌ها در این پروتکل از لایه‌ی انتقال TCP و پروتکل HTTP‌ استفاده می‌شود. Gateway متصل به پلتفرم هیچ پردازشی روی داده‌ها صورت نمی‌دهد و آن‌ها را مستقیما به لایه‌ی بالاتر انتقال می‌دهد. در این بین تنها token شی از نظرت صحت بررسی می‌گردد. شی با استفاده از تقاضا POST داده را به endpoint پیشنهادی /data بر روی پورت 80 Gateway ارسال می‌کند. در ازای این تقاضا یک پاسخ مبنی بر دریافت gateway به شی ارسال خواهد شد.

* **تبادل داده: ارسال داده از پلتفرم به اشیاء**

سرور داده‌ها را از طریق ارتباط socket.io که پیشتر در فاز احراز هویت شکل گرفته است برای شی مورد نظر ارسال می‌کند. با توجه به اينكه اين ارتباط در داخل يك تونل TLS انجام مي‌گيرد، داده‌ها قابل شنود توسط سایرین نيستند علاوه بر آن ساختار این داده‌ها از منظر Gateway پنهان می‌باشد و توسط لایه‌ی بالاتر عمل encoding صورت پذیرفته است. همانگونه که ذکر شد، شی در مرحله احراز هویت با استفاده از http و پورت 80 Gateway‌ ارتباط socket.io را با سرور شکل می‌دهد و در ادامه داده‌ها در channelای با نام پیشنهادی data برای شی منتشر خواهند.

* **راه کارهای پیشنهادی**

در این پروتکل می‌توان جهت کاهش حجم داده‌ی انتقالی در صورت توافق با کارفرما از روش‌های compression استفاده کرد یا به جای استفاده از پروتکل HTTP‌ از پروتکل سبکتر CoAP بهره برد.

* 1. پوشش نیازمندی­ها

پروتکل پیشنهادی در مرحله قبلی، نیازمندی‌های ذکر شده در بخش ۱ را به شرح زیر پوشش می‌دهد.

پوشش نیازمندی­های پروتکل LAN

|  |  |  |
| --- | --- | --- |
| ردیف | نیازمندی | نحوه پوشش |
| 1 | تبادل داده Up-Link | ارسال بسته POST به End-Point مشخص شده |
| 2 | تبادل داده Down-Link | تبادل داده بر روی socket.io |
| 3 | Platform Authentication | Verify کردن Certificate پلتفرم توسط شی با استفاده از CA |
| 4 | Thing Authentication | استفاده شناسه و رمز عبور در درخواست اول، استفاده از token در درخواستهای بعدی |
| 5 | Confidentiality | الگوریتم رمزنگاری استفاده شده از TLS كه هم بسته‌هاي HTTP و هم ارتباطات socket.io بر روي آن بستر انجام ‎‌مي‌شود. |
| 6 | Integrity | الگوریتم HMAC استفاده شده از TLS |