r/esp32

Posts

Posted by u/frankcohen just now

1

Managing SSL certificates for HTTPS with IoT devices? ESP32+Nginx+Node.js+OTA

Many tutorials I found on using HTTPS connections encode the server's SSL public key certificate in the firmware/sketch. The Arduino code looks like this:

```
const char* root_ca= \
"-----BEGIN CERTIFICATE-----\n" \
"MIIG1TCCBL2gAwIBAgIQbFWr29AHksedBwzYEZ7WvzANBgkqhkiG9w0BAQwFADCB\n" \
. . .
"ON51WhJ6W0xNdNJwzYASZYH+tmCWI+N60Gv2NNMGHwMZ7e9bXgzUCZH5FaBFDGR5\n" \
"S9VWqHB73Q+OyIVvIbKYcSc2w/aSuFKGSA==\n" \
"-----END CERTIFICATE-----\n";

void loop() {
  if ((WiFi.status() == WL_CONNECTED)) { //Check the current connection status
    HTTPClient http;
    http.begin("https://mydomain.com/listfiles", root_ca); //Specify the URL and cert
    int httpCode = http.GET();                                        //Mak
```

They usually come with a warning not to put the SSL certificate into the code. SSL certificates have expiration dates. The firmware fails to make a connection with an expired SSL certificate and changing the certificate requires uploading new firmware.

My project is a wrist watch. It uses an ESP32 to get data using HTTPS calls to an nginx service for SSL protocol support to a node.js based service. It is an open-source project and the repository is at https://github.com/frankcohen/ReflectionsOS.

My preferred way to deliver new features and manage SSL certificates is to keep the certificates in the code and use Over The Air (OTA) protocol to update the firmware. I am looking at Chris Joyce's OTA library. The risk of this approach is the SSL certificate expires before the next OTA update. That would require the user to upload firmware over USB to the watch, or return the watch to me for updating.

What are your ideas on how to manage SSL certificates in an IoT device?

-Frank

By the way, I found the latest ESP32 HTTPClient does not support SSL TLS 1.3 protocol. Making an HTTPS request to nginx/node.js using an ESP32 gives this error:

> MBEDTLS_ERR_SSL_FATAL_ALERT_MESSAGE, -0x7780, -30592, "A fatal alert message was received from our peer."

ESP32 supports TLS 1.2 and not 1.3. Change your nginx configuration settings in /etc/nginx/conf.d/ssl.conf:

```
ssl_protocols TLSv1.2 TLSv1.1 TLSv1.3;
ssl_prefer_server_ciphers on;
ssl_ciphers EECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH:!aNULL:!MD5:!DSS;
ssl_ecdh_curve secp384r1; # Requires nginx >= 1.1.0
```

Post Insights

Check back later to see views, shares, and more. Share your post to spread the word!

Comment as frankcohen

What are your thoughts?

Markdown Mode Comment

Sort By: Best

No Comments Yet

Be the first to share what you think!

Back to Top