MAXREFDES155# DeepCover Embedded Security in IoT: Public-key Secured Data Paths Quick Start Guide

UG6389; Rev 0; 2/17

Abstract

This Quick Start Guide provides information about preparing and running the MAXREFDES155# ARM® mbed™ example and web client. The MAXREFDES155# subsystem reference design secures an authenticated data link between IoT devices and the web using the DS2476 DeepCover® secure coprocessor.

Table of Contents

List of Figures

Required Equipment

The equipment, ARM® mbed™ shield (MAXREDES155#) and mbed base board (MAX32600MBED#), are available for separate purchase at Maxim Integrated's website to produce the hardware needed for the mbed system. Here is the list of the equipment required:

- MAXREFDES155# kit including:
  - MAXREFDES155# mbed shield.
  - Infrared (IR) laser-sensor module.
  - 30.48cm long x 1mm pitch SR Cable (A04SR04SR30K305B).
- MAX32600MBED# base board.
- Wifi Hotspot (internet access is needed for MAXREFDES155#)
- USB A to USB micro-B cable
- Internet-connected computer with USB-to-load firmware.

Import Firmware

1. Load the mbed repository page for the "MAXREFDES155" firmware program in your web browser.
    a. Go to the mbed web page located at https://developer.mbed.org.
    b. In the upper-right corner use the **Search mbed...** box type the text string "MAXREFDES155" and hit the enter key to initiate the search.
    c. Click on the top search result that begins with "MAXREFDES155."
2. Import the latest revision of the program into the online compiler by clicking on the **Import program** button in the mbed repository page (**Figure 1**). A free mbed account is required to access the online compiler.

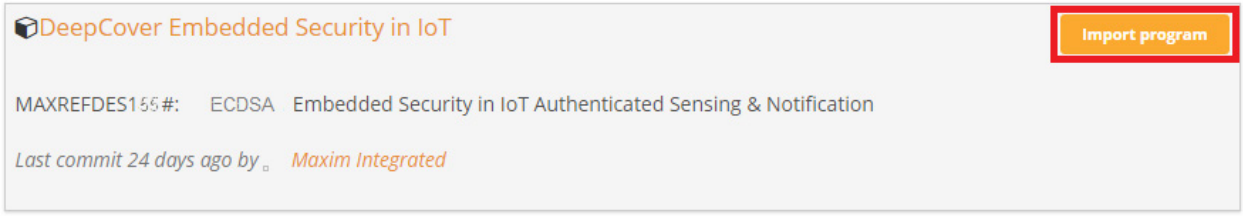

*Figure 1. **Import program** button.*

Load Firmware

1. Add the MAX32600MBED platform to your compiler by clicking the **Add to your mbed Compiler** button at the top right side of the MAX32600MBED platform page (**Figure 2**).

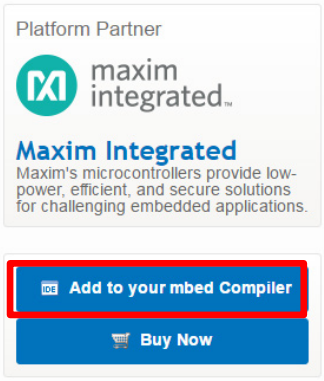

*Figure 2. **Add to your mbed Compiler** button.*

2. Return to the online compiler page, and select the MAX32600MBED as the compiler target by the following steps in **Figure 3**:

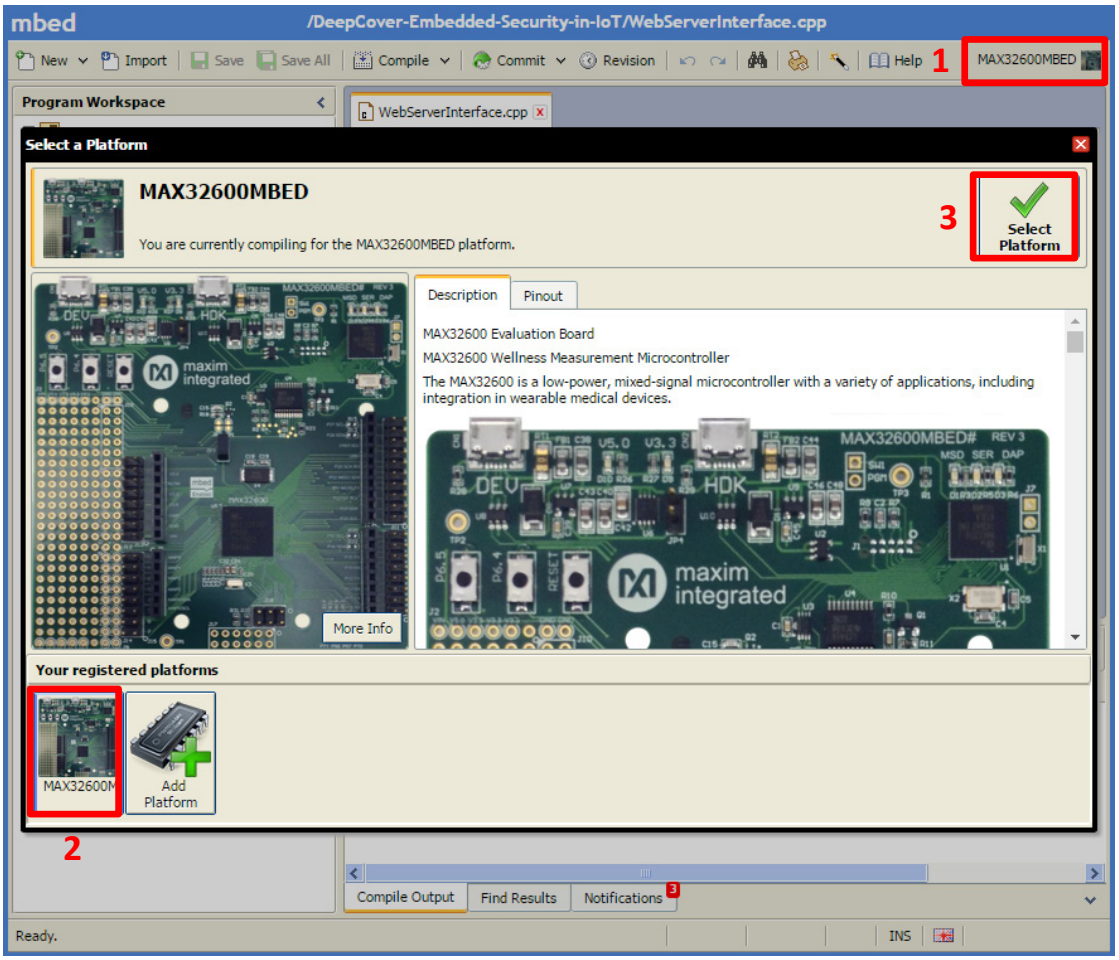

Figure 3. Selecting the MAX32600MBED as the compiler target.

3. You will be prompted to download "MAXREFDES155.bin," which is the compiled binary firmware. **Compile** the firmware (**Figure 4**).

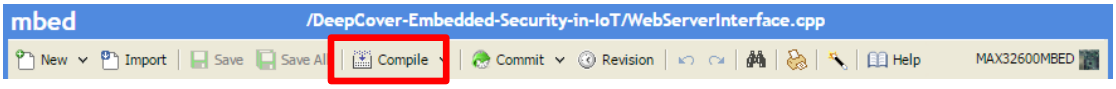

Figure 4. Compiling the firmware.

4. Connect the MAX32600MBED to the computer using the HDK USB port for programming (**Figure 5**).

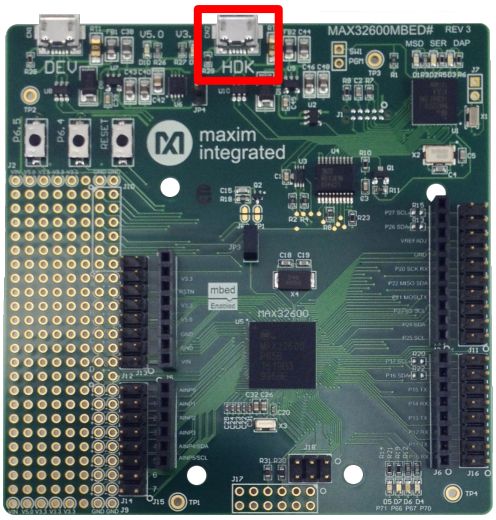

Figure 5. Connect the MAX32600MBED to a computer using the HDK USB port.

5. The MAX32600MBED# acts as a USB-storage drive when connected to the computer. Drag the **MAXREFDES155.bin** binary to the **MBED** USB storage drive to program the MAX32600MBED. When done transferring the binary, safely disconnect the MAX32600MBED# HDK USB port as you would any USB storage drive.

Demo Setup

This section describes the steps to setup the hardware for the demo.

1. Connect the IR-laser-sensor module (J7) to the MAXREFDES155# mbed shield (J3) using the supplied SR cable.
2. Insert the downward facing pins on the MAXREFDES155# assembly into the MAX32600MBED# as shown (**Figure 6**).

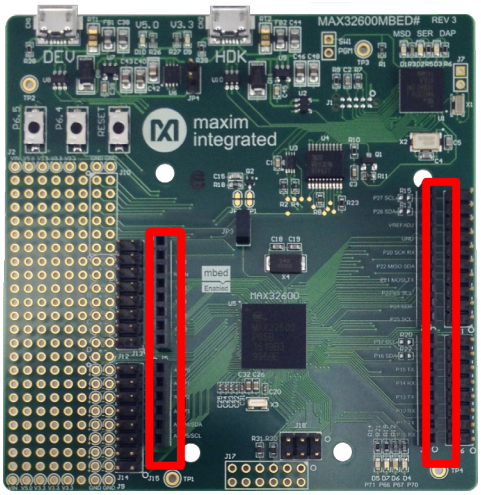

Figure 6. Inserting pins on the MAXREFDES155# shield into the MAX32600MBED# base board.

3. Power the MAX32600MBED# through the DEV USB port (**Figure 7**). The MAXREFDES155# mbed shield immediately displays the unique ID for the MAXREFDES155# mbed shield. This is used for session tracking on the web server. Write down this session ROM ID for later and do not click any button before executing the next step. This allows the web client to be ready to display any communication received from the mbed.

Figure 7. Power the MAX32600MBED# and MAXREFDES155# using the DEV USB port.

4. With a PC go to the following website: www.mxim-security.us/ds28c36
5. Type in the 64-bit ROM ID that you wrote down from **Demo Setup,** step 3 as per the example in **Figure 8** and click on **Continue** to "Sync with your mbed."

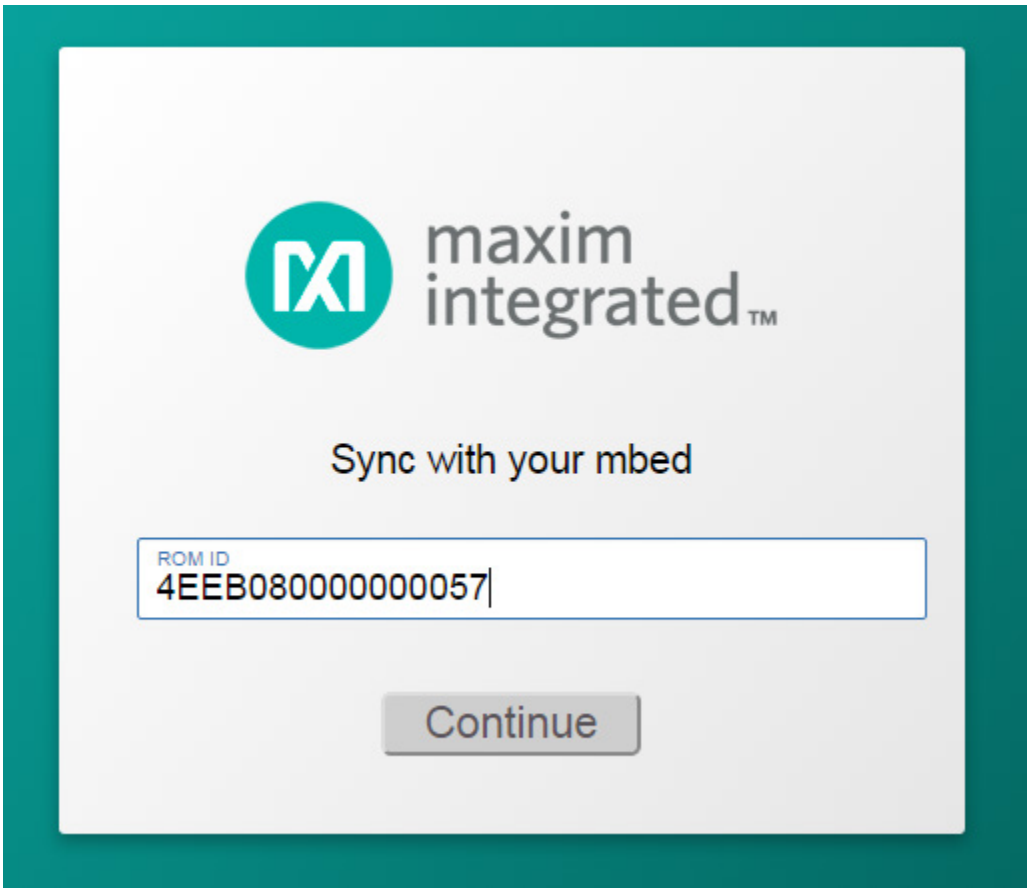

Figure 8. Sync with your mbed.

6. Verify the PC is connected to the web server as shown in **Figure 9**.

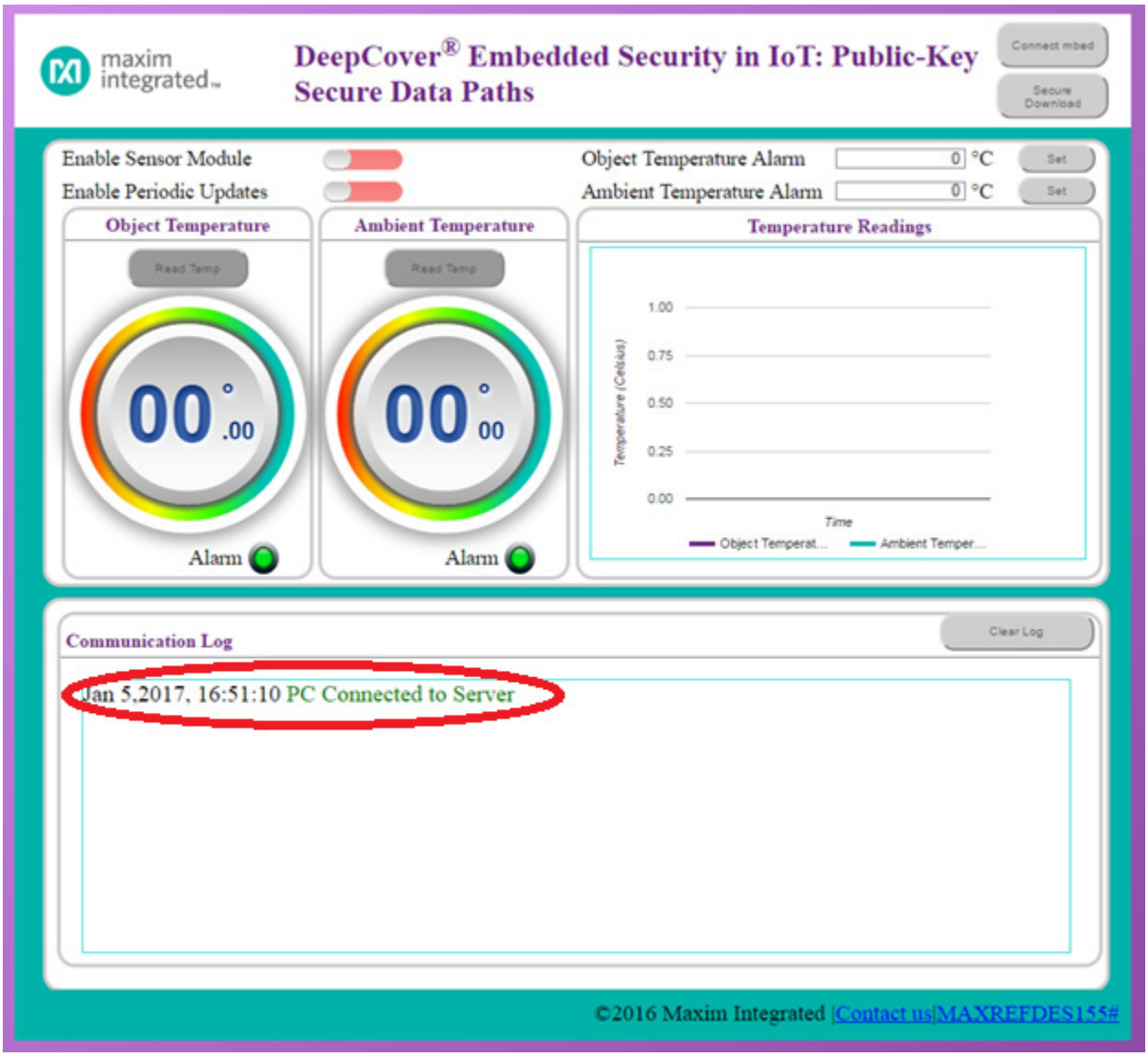

Figure 9. Verify the PC Connected to Server.

7. Setup an internet hotspot setup with a network name of "*MAXREFDES155*" and a network password of "*maxim1234*." Make sure to place the hotspot in reasonable proximity to the mbed.
8. Next, provision the MAXREFDEF155# mbed shield by pressing its **Left Click** button when prompted. Follow the directions of each step on the LCD screen of the mbed shield.

9. Verify the mbed shield LCD screen shows a **Valid signature: 1** as shown in **Figure 10** indicating the mbed is sending valid signatures to the web server.

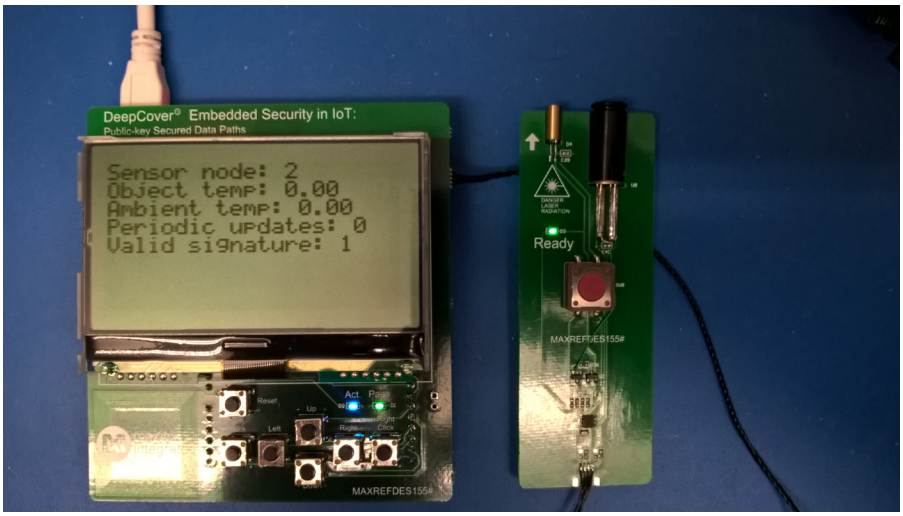

Figure 10. Sending valid signatures to the web server.

10. Verify the web client shows an **Authentic mbed** as in **Figure 11**. This is critical as it indicates from the web server that an *authentic mbed* connection has been recognized. Note: The example web client for this design utilizes a shared instance provided by Maxim for your convenience. Maxim makes no guarantees regarding reliability, availability, or data security when utilizing the shared web-client instance. The authenticated connection recognized in this demo is between the mbed and the web server.

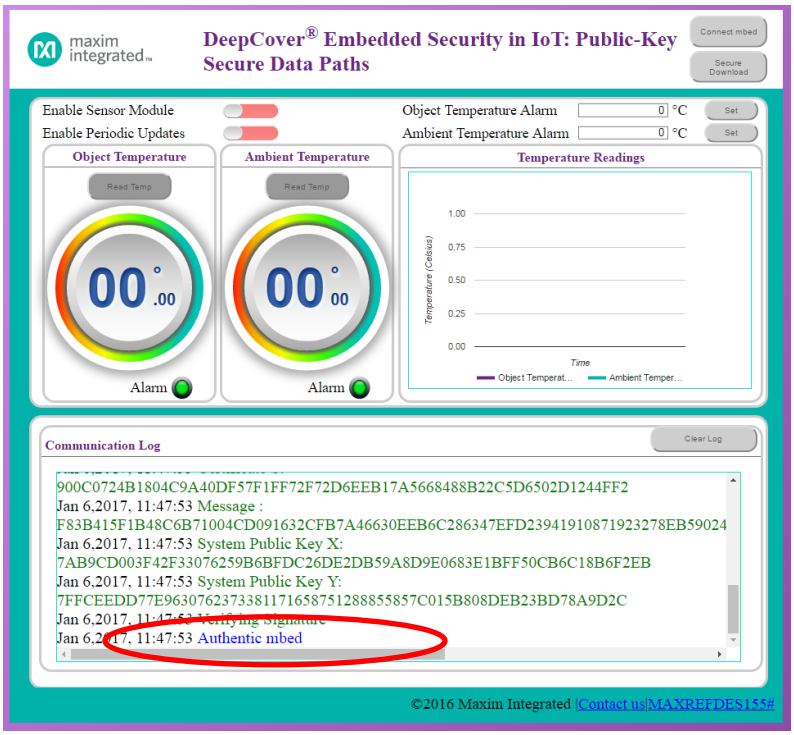

Figure 11. Web server verification that the mbed is Authentic.

11. To summarize, if everything is correct the following (as illustrated in **Figure 12**) shows what occurred to establish a secure connection to the Web Server:

    a. mbed connects to the web server through Wifi.

    d. mbed sends ROMID, certificate, and device public key to the web server.

    e. Web server uses the system public key to verify mbed is part of the system by checking the certificate.

    f. The web server now sends the challenge with a request for mbed's digital signature.

    g. The mbed sends the digital signature and the web server verifies the signature by using the additional ingredients of the known challenge, device public key, and ROMID. If the mbed is authenticated, then the web server allows the connection. If not authenticated, the web server drops the mbed connection and does not accept any incoming data.

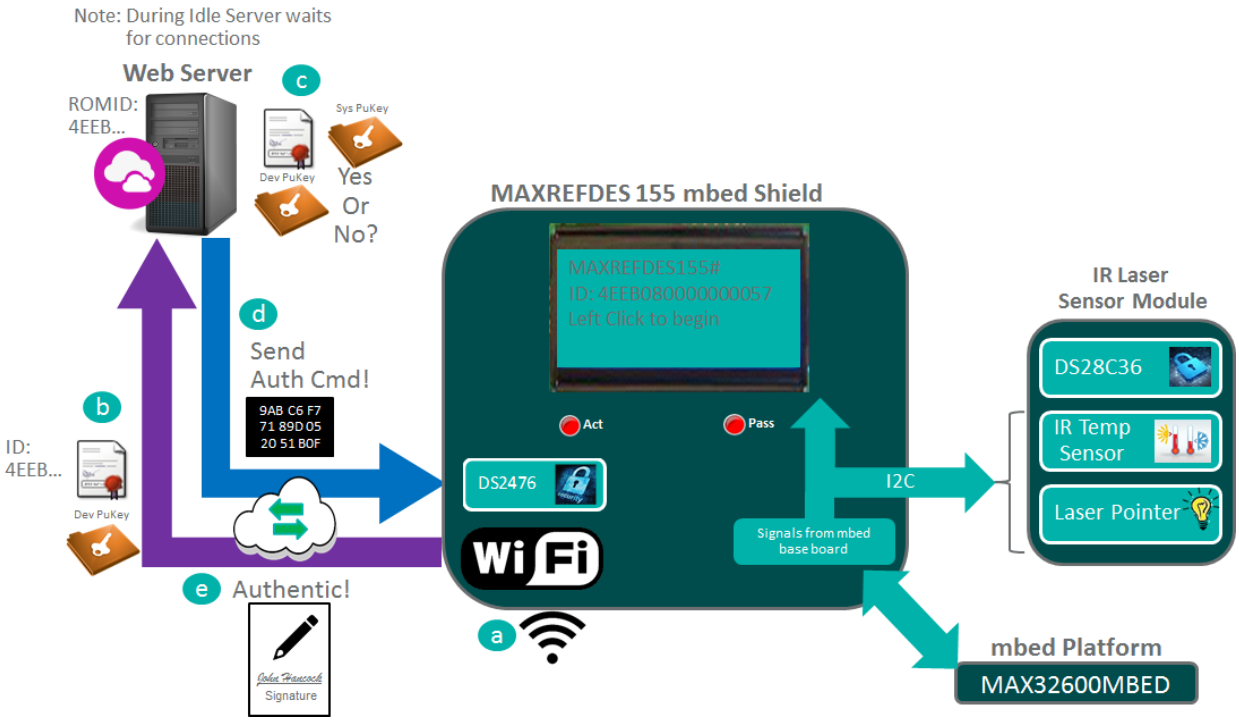

Figure 12. Establishing the setup connection to the web server.

Run the Demo

Temperature Demo

A web client runs the demo by executing the mbed commands and displaying the results of temperature that was authenticated by the web server. After the **Demo Setup** has successfully completed do the following to run this demo:

1. Click on the **Enable Sensor Module** so the laser-pointer power is enabled securely. Note: For safety, when the mbed receives the enable sensor module, the DS2476 authenticates the DS28C36. If the DS28C36 is found authentic then the DS28C36 GPIO is enabled by a write authentication. This in turn enables the laser-pointer power. Otherwise, the DS28C36 GPIO is kept disabled which in turn keeps the laser pointer in a safe state.

2. Click on the **Read Temp** button (**Figure 13**) and verify a temperature was read and is authentic.

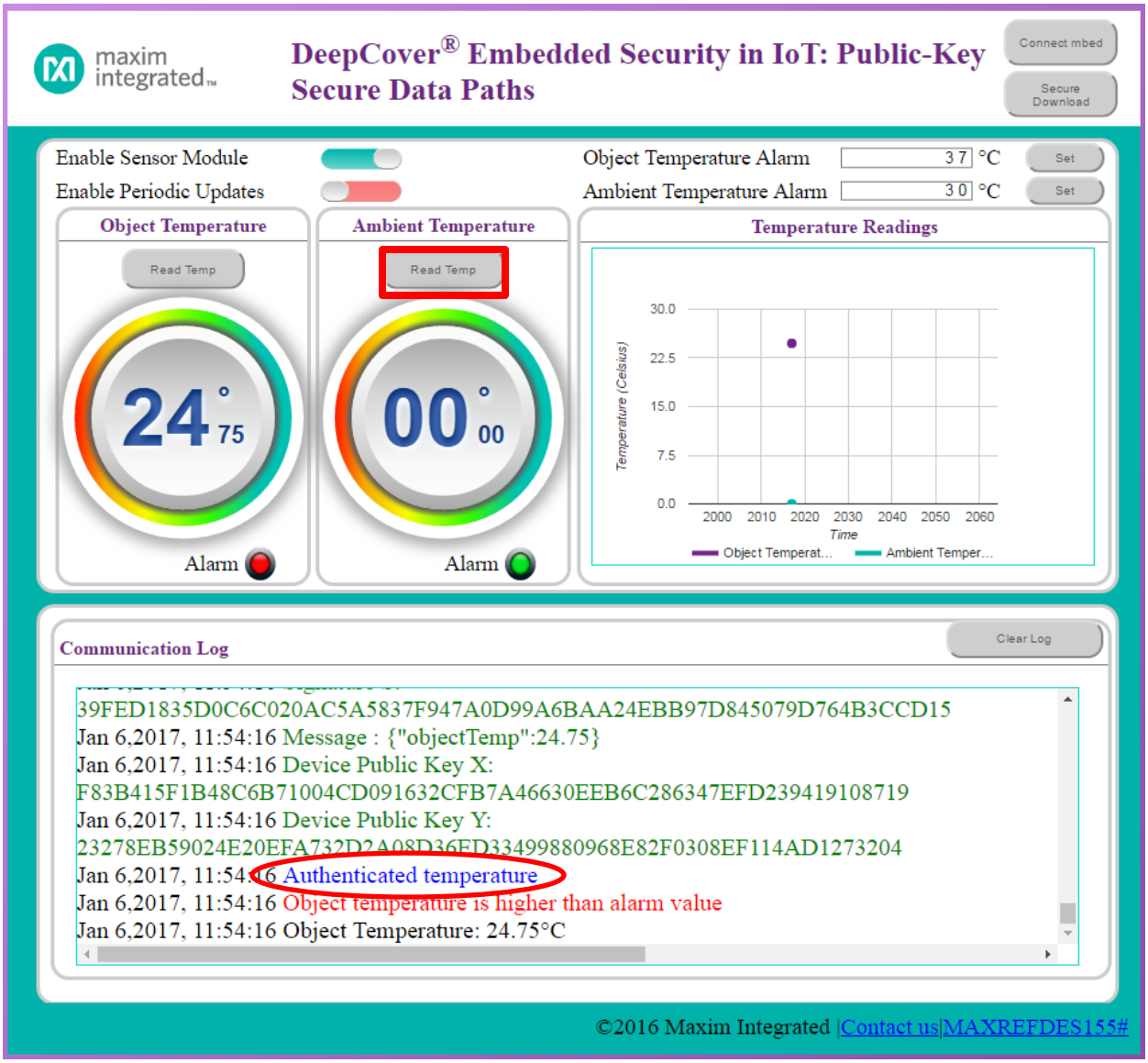

Figure 13. An authenticated temperature reading.

3. Slide the **Enable Periodic Updates** and this continually requests and receives an authenticated temperature reading while graphing a log of the results in the **Temperature Readings** panel. The increments of the temperature readings occur every five seconds. Each reading can be verified to be authentic as listed in the **Communication Log** panel as shown in **Figure 14**.

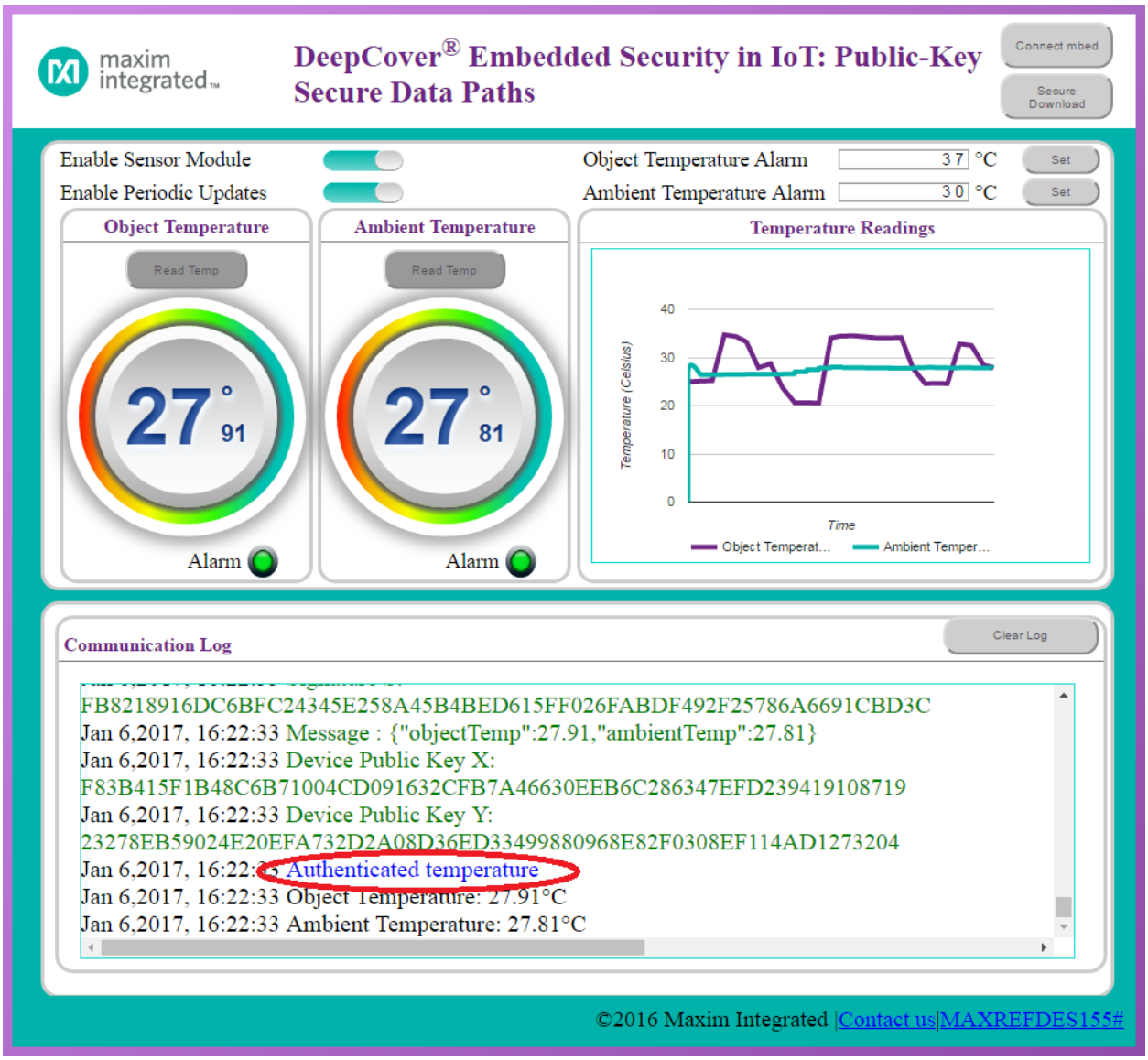

Figure 14. An Authenticated temperature reading with history graphed.

4. In summary, the **Temperature Demo** authenticates temperature each time by doing the following (**Figure 15**):

    a. The mbed receives a get temperature command along with the challenge from the web server (i.e., it is instructed by the web client) for a temperature reading.

    h. On the mbed, DS2476 locally authenticates (i.e., by using a locally symmetric-based authentication with HMAC SHA-256) the DS28C36 on the sensor module and locally reads the temperature.

    i. Assuming the sensor module is declared authentic, the mbed reads and signs the temperature reading with the device private key in DS2476 and sends the data to the web server.

    j. The web server uses the device public key obtained beforehand during the **Demo Setup**, the challenge, and the just received temperature reading/signature to verify authenticity.

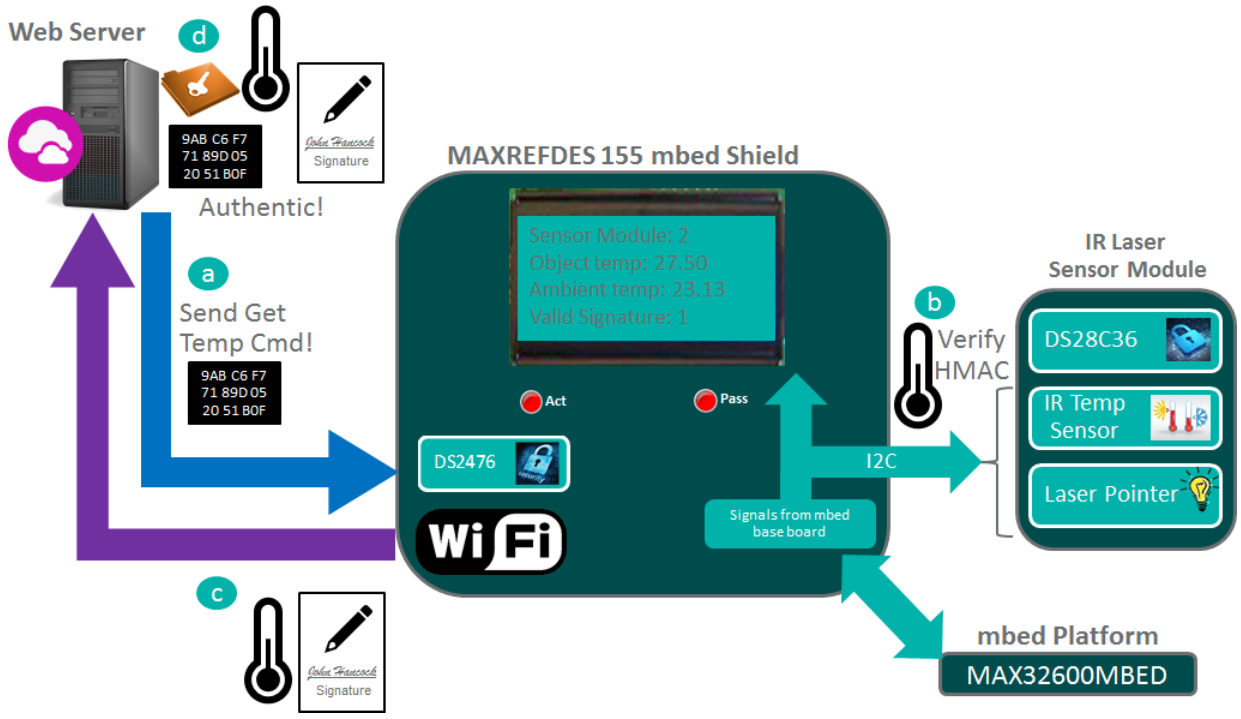

Figure 15. An illustration of Authenticated temperature readings.

Image Demo

The web client can also be used to send an image securely to the mbed platform. After the **Demo Setup** has completed successfully, do the following to run this demo:

1. In the web client, click on the **Secure Download** button and a popup window opens.
2. Select an image to download, check the **Valid** box so the image has a digital signature generated/used and click on **Download** to send the image to the LCD display of the mbed shield (**Figure 16**).

   Note: the web client contains the color images and the correlating black & white LCD image.

Figure 16. Select image to download.

3. Verify the MAXREFDES155# mbed shield shows the selected iButton® image in black and white on the LCD display.
4. In summary, the **Image Demo** authenticates the image each time by doing the following:
   a. The web client sends the LCD image bytes and settings to the web server. (Figure 16)
   b. The web werver checks the **Valid** flag.
      - If checked, the web server uses system private keys and signs the LCD image data. Next, it sends the LCD image data and signature to mbed.
      - If not checked, the web server uses a different private key and signs the LCD image data. Then sends the LCD image data and invalid signature to mbed.
   c. The mbed gets the LCD image data plus the correlating signature and verifies the signature.
      - If the signature passes, then the mbed LCD shows the image.
      - If the signature fails then the mbed LCD does not show the image but an error message.

Trademarks

DeepCover is a registered trademark of Maxim Integrated Products, Inc.
iButton is a registered trademark of Maxim Integrated Products, Inc
mbed is a registered trademark of ARM Limited.