Seguridad en sistemas embebidos

SISTEMAS EMBEBIDOS - GRADO EN INGENIERÍA INFORMÁTICA

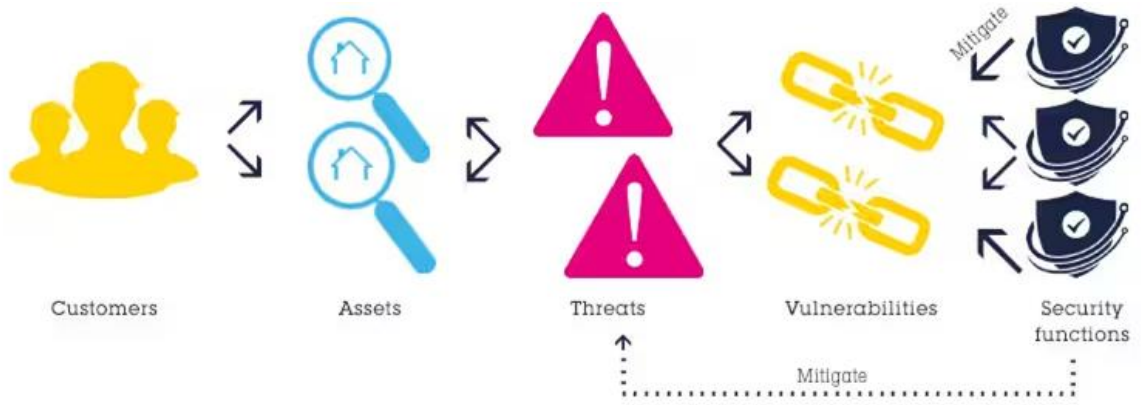
Contenidos

1. Introducción
2. Tipos de ataque
3. Seguridad en ESP32
4. Conclusiones

Introducción

Introducción

La seguridad en sistemas embebidos se ha vuelto una preocupación importante en la era de la Internet de las cosas (IoT). La creciente cantidad de dispositivos conectados a la red y la gran cantidad de datos que se recopilan en ellos ha creado una mayor necesidad de seguridad en sistemas embebidos

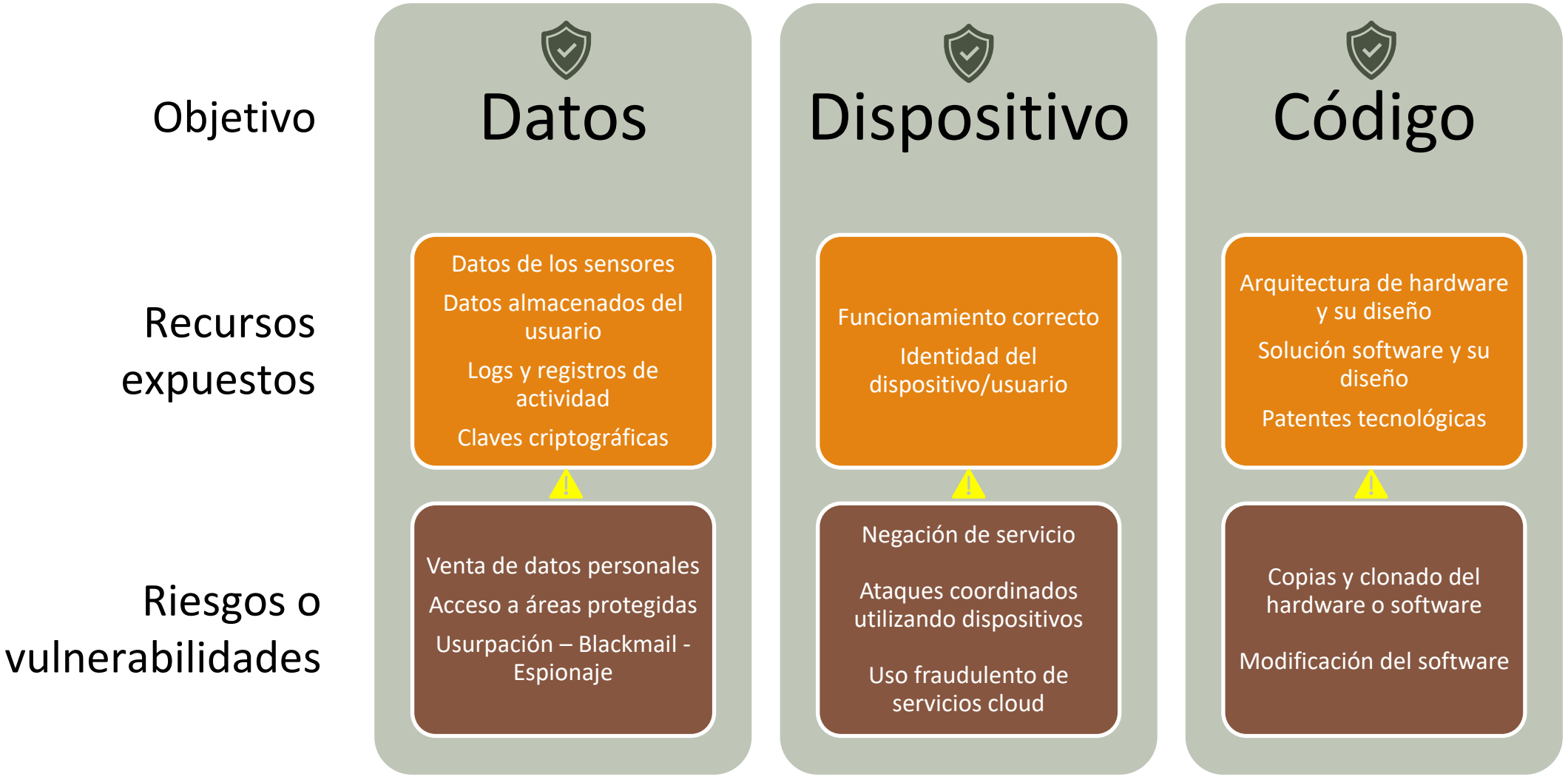


Threats exploit vulnerabilities and damage **assets.**
Protections mitigate vulnerabilities and therefore can mitigate **threats.**

Los gobiernos y las instancias reguladoras están concienciando sobre la importancia de la seguridad de los dispositivos IoT y han establecido normas estrictas para racionalizar el enfoque.

La seguridad de un dispositivo embebido es primordial para impedir que los hackers lo controlen, Los ataques remotos o los ataques locales al son otra forma de obtener información secreta que debemos detener

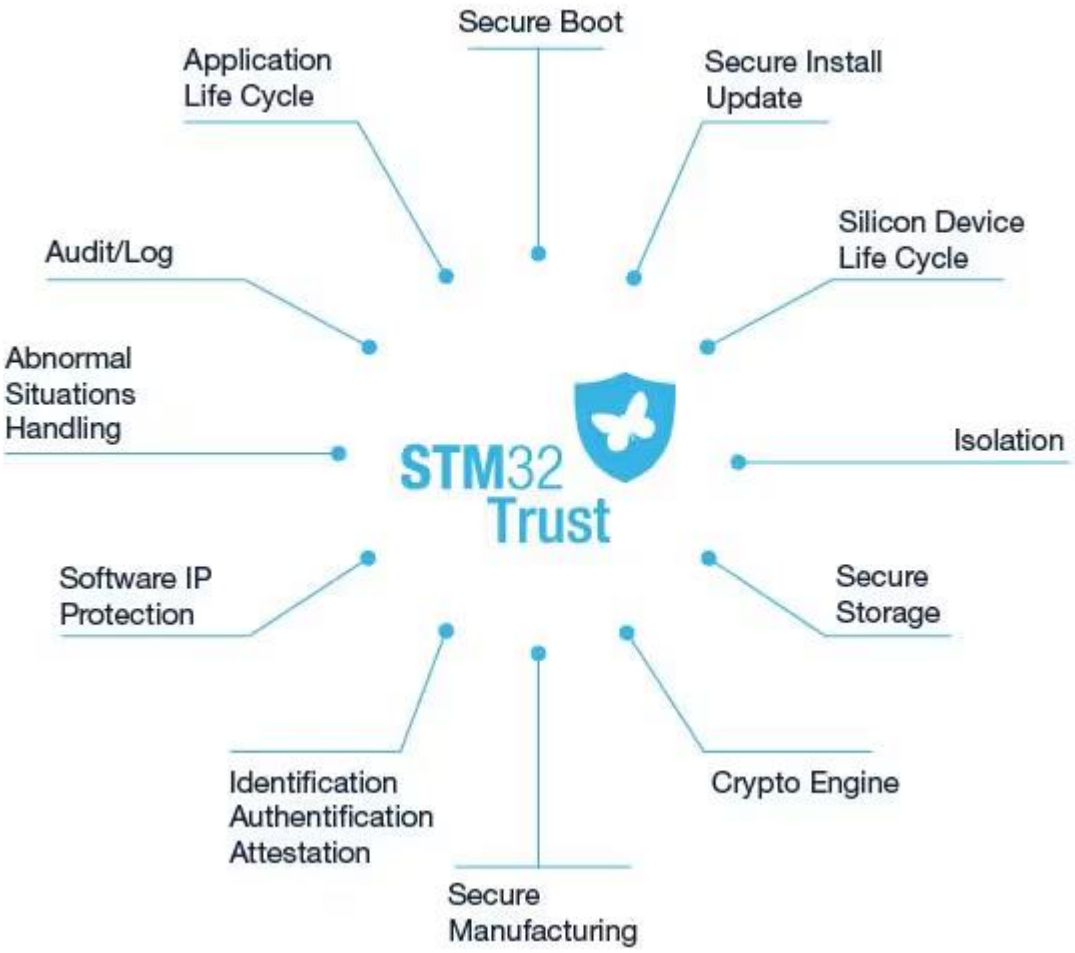
Introducción – ¿Que se debe proteger?



Introducción – Tipos de ataque

Tipos de ataque	Software	Red o comunicaciones	Hardware	Hardware / invasivo
Alcance	Remoto o local	Remoto	Local	Local a nivel de placa
Técnicas	Malware Vulnerabilidades de protocolos Acceso por fuerza bruta Desbordamiento de buffers	Ataque de *man in the middle* Interferencia de señales Cambio de DNS	Escuchas de puertos debug Análisis de señales Inyección de errores Fallos de alimentación Cambios de firmware	Ingenieria inversa Sondeo de dispositivo Focused Ion Beam Laser fault injectión
Coste del ataque	Bajo o medio	Bajo o medio	Elevado. Equipamiento y habilidades requeridas	Muy costos, equipamiento especifico, habilidades específicas
Objetivos	Acceso a información del usuario Denegación de servicio	Acceso a información del usuario Usurpación	Acceso a datos protegidos del dispositivo	Procesos de ingeniería inversa, Jailbreak, desbloqueo de funcionalidades

Introducción – Estrategias de defensa



1. Secure boot: (Arranque seguro) Capacidad de garantizar la autenticidad e integridad de una aplicación que se ejecuta dentro de un dispositivo.

2. Secure Install/Update: (Instalación/actualización segura) Instalación o actualización de firmware con comprobaciones iniciales de integridad y autenticidad antes de la programación.

3. **Secure storage:** (Almacenamiento seguro) Capacidad para almacenar de forma segura secretos como datos o claves (y acceder a ellos sin que sean visibles externamente).

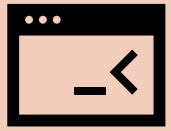
4. **Secure manufacturing:** (Fabricación segura) aprovisionamiento inicial del dispositivo en un entorno no seguro con control de sobreproducción. Posible personalización segura.

5. **Identification:** Iidentificación única de un dispositivo y/o paquete de software, y capacidad de detectar su autenticidad, desde el interior del dispositivo o externamente.

6. **Abnormal situation handling:** Gestión de situaciones anómalas Capacidad para detectar situaciones anómalas (tanto de hardware como de software) y tomar decisiones adaptadas, como la eliminación de datos secretos.

7. **Crypto Engine**: Funcionalidades de criptografía disponibles para la programación de comunicaciones seguras

Introducción – Escenarios



Venta de firmware. Aislamiento del firmware frente al código del cliente.

Rutina segura de instalación
Rutina segura de actualización
Aislamiento de funcionalidades

Venta de dispositivos.

Actualizaciones necesarias, solo firmware original.

Rutina segura de instalación
Rutina segura de actualización
Arranque seguro

Recolección y comunicación de datos personales.

Arranque seguro
Cifrado de comunicaciones
Sistemas de identificación

Tipos de ataque

Tipos de ataque - software

Malware

Los ataques de malware a sistemas embebidos funcionan de la misma manera que con cualquier otro sistema: un hacker despliega un fragmento de código malicioso que intenta interceptar los datos almacenados en el interior del sistema, tomar el control del sistema víctima o dañarlo.

Por lo general, los hackers falsifican actualizaciones de firmware, controladores o parches de seguridad para distribuir malware.

Estrategias de defensa:

Para protegernos de actualizaciones maliciosas podemos utilizar las funcionalidades de *secure boot* asegurando que nuestros dispositivos solo ejecuten código firmado.

Tipos de ataque - software

Ataque de fuerza bruta

Consiste en intentar acceder a los credenciales de acceso a base de multiples comprobaciones de los mismos. Muchos sistemas embebidos no incluyen *timeouts* o limite de intentos para evitar comprobaciones de credenciales demasiado frecuentes

Estrategias de defensa:

Timeouts o limite temporal de intentos de acceso

Desbordamiento del búfer

Consiste en un tipo de ataque en el se intenta desbordar manualmente el búfer de memoria asignado para contener los datos que se mueven dentro de un sistema embebido. En este caso, el sistema operativo integrado grabará algunos de esos datos en secciones de memoria situadas junto al búfer. Los datos grabados pueden contener shellcode u otros exploits que ayuden a los hackers a obtener credenciales y elevar sus derechos de acceso.

Tipos de ataque - red

Man in the middle (MITM)

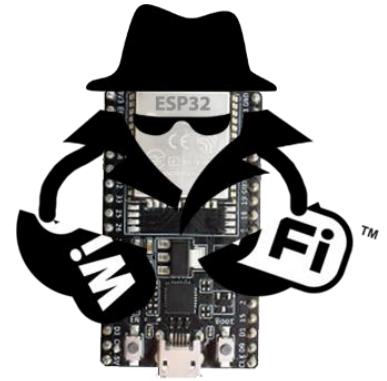
Un ataque de este tipo se utiliza para interceptar o alterar los datos transmitidos por un sistema embebido. Para ejecutarlo, los hackers modifican los parámetros de conexión de dos dispositivos con el fin de interponer un tercero entre ellos.

Si un tercero consigue obtener o alterar las claves criptográficas utilizadas por ambos dispositivos, pueden espiar de una forma muy difícil de detectar, ya que no causa ninguna interrupción en la red.

Estrategias de defensa:

Cifrado de datos transmitidos

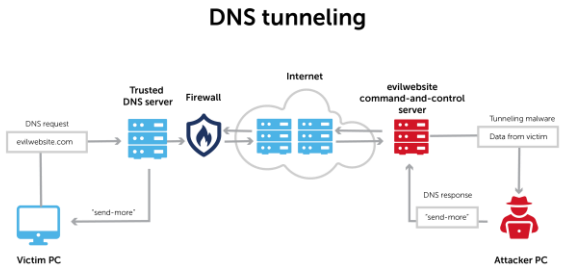
Utilización de protocolos de seguridad para transmission de claves



Tipos de ataque - red

Cambio de DNS

Si la red ha sido comprometida, es posible forzar al gatway a utilizer un servidor especifico de DNS con registros corruptos. Aprovechando las vulnerabilidades de un servidor DNS y envenenando su caché, los hackers pueden redirigir el tráfico de un sitio web objetivo a cualquier otra dirección.



Ataque DDos

DDoS es un conocido ataque que hace que un sistema no esté disponible al desbordarlo con peticiones de varias fuentes. Estos ataques son difíciles de detener porque las peticiones proceden de un gran número de IP. No existe una protección universal contra los ataques DDoS, pero sí algunos métodos eficaces de protección DDoS

Tipos de ataque - Hardware

Escucha de puertos debug

Algunos sistemas embebidos son comercializados con las rutinas y puertos debug expuestos, provocando una vulnerabilidad en el dispositivo. Es importante recorder que antes de desplegar un sistemas hay que desactivar todas aquellas funcionalidades y servicios no esenciales que no esten siendo usadas.

Analisis de consumo

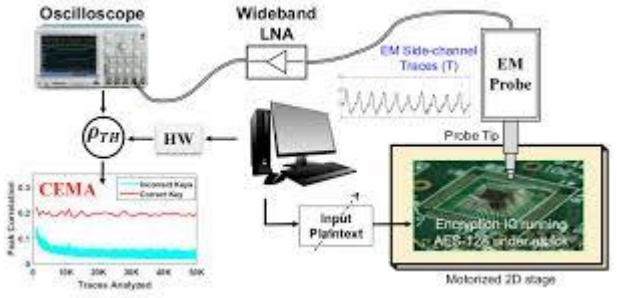
Un ataque de análisis de energía requiere acceso físico a un sistema integrado para sondear sus conexiones y detectar cambios en el consumo de energía. Estos cambios dependen de los datos procesados por el sistema, por lo que los hackers pueden detectar cuándo un sistema procesa un tipo concreto de información e interceptarla.

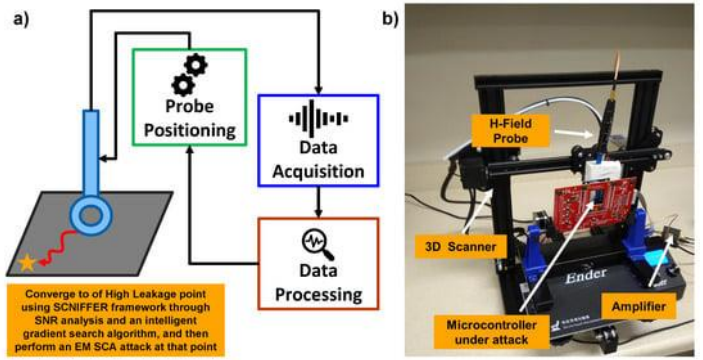
Tipos de ataque - Hardware

Sondeo electromagnético

El análisis electromagnético ofrece a los piratas informáticos otra forma de mirar dentro del software integrado. Utilizar el análisis electromagnético para grabar y analizar las emisiones de un dispositivo, averiguar sus operaciones criptográficas e incluso extraer claves secretas.

Se trata del tipo de ataque que requiere más tiempo y dinero, ya que exige proximidad física y amplios conocimientos del diseño del dispositivo





Estrategias de defensa:

Añadir protección física (por ejemplo, utilizando ciertos tipos de pegamento o blindaje de circuitos y cables).

La aleatorización y las interrupciones de los procesos de cifrado también son contramedidas eficaces contra el análisis electromagnético

Introducción

Tipos de ataque

Seguridad en ESP32

Conclusiones

Seguridad en ESP32

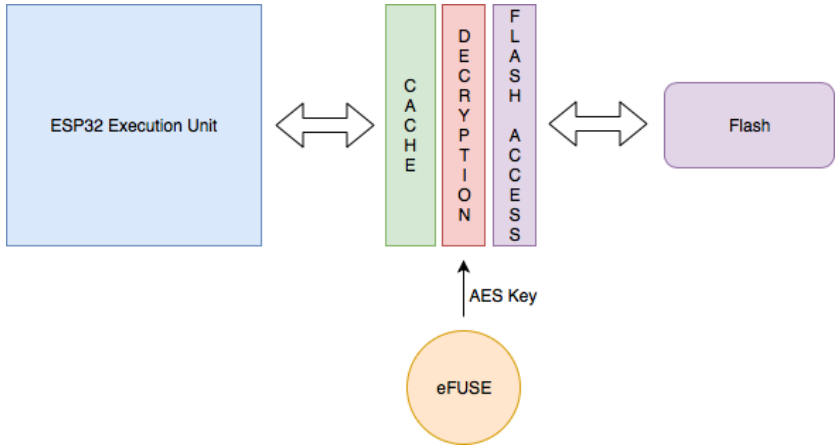
Seguridad en ESP32 – eFUSE

El eFUSE desempeña un papel importante en el funcionamiento de estas funciones de seguridad. Así que vamos a ver rápidamente en el eFUSE antes de entrar en las características de seguridad.El ESP32 tiene una eFUSE de 1024 bits, que es una memoria programable de un solo uso. Esta eFUSE se divide en 4 bloques de 256 bits cada uno.

Los bloques 1 y 2 son los que más nos interesan ahora. Estos bloques almacenan las claves para el cifrado flash y el arranque seguro respectivamente. Además, una vez que las claves se almacenan en el eFUSE, se puede configurar de tal manera que cualquier software que se ejecuta en ESP32 no puede leer (o actualizar) estas claves (desactivar la lectura de software). Una vez habilitado, sólo el hardware ESP32 puede leer y utilizar estas claves para garantizar el arranque seguro y el cifrado flash.

Seguridad en ESP32 - Flash Encryption

La encriptación flash está pensada para encriptar el contenido de la memoria flash externa del ESP32. Una vez activada esta función, el firmware se carga como texto sin formato y, a continuación, los datos se cifran en el primer arranque. Como resultado, la lectura física de la memoria flash no será suficiente para recuperar la mayor parte de su contenido.



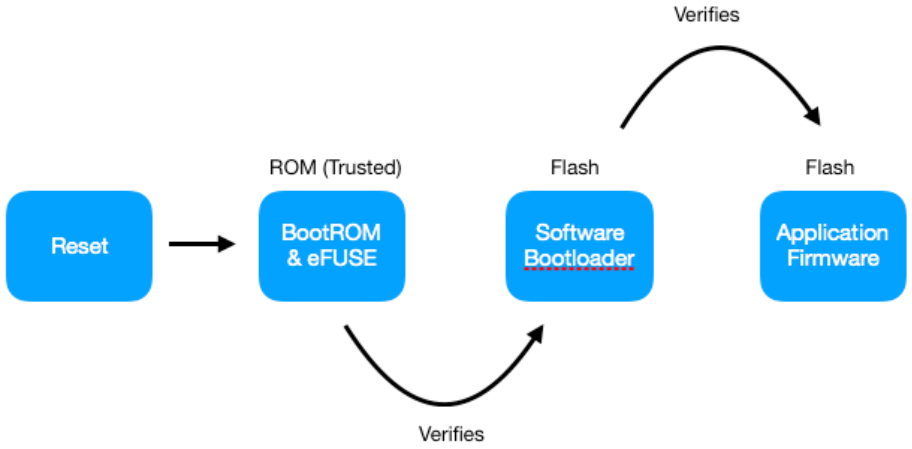
El soporte de encriptación flash garantiza que cualquier firmware de aplicación que se almacene en la memoria flash del ESP32 permanezca encriptado. Esto permite a los fabricantes incluir firmware cifrado en sus dispositivos.

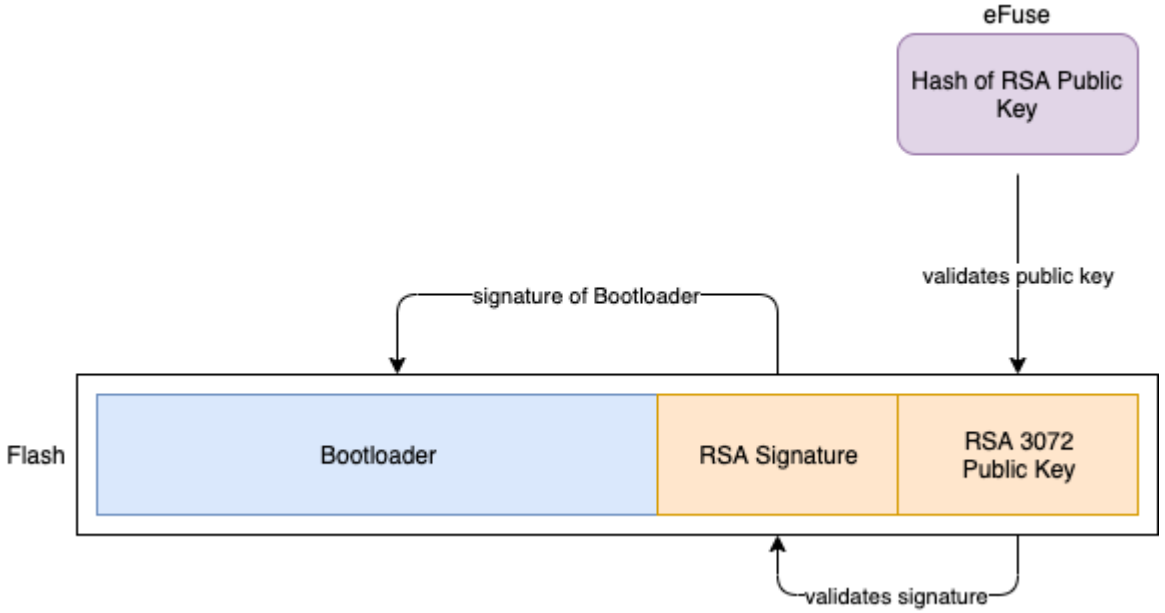
Cuando la encriptación flash está activada, todos los accesos de lectura a la memoria flash se desencriptan de forma transparente y en tiempo de ejecución. El controlador flash utiliza la clave AES almacenada en la eFUSE para realizar el descifrado AES

Como la clave está bloqueada en la eFUSE, sólo el hardware puede utilizarla para descifrar el contenido de la memoria flash.

Seguridad en ESP32 – Secure boot

El soporte de arranque seguro garantiza que cuando el ESP32 ejecuta cualquier software desde la memoria flash, ese software es de confianza y está firmado por una entidad conocida. Si se modifica incluso un solo bit en el cargador de arranque de software y el firmware de la aplicación, el firmware no es de confianza, y el dispositivo se negará a ejecutar este código no fiable.



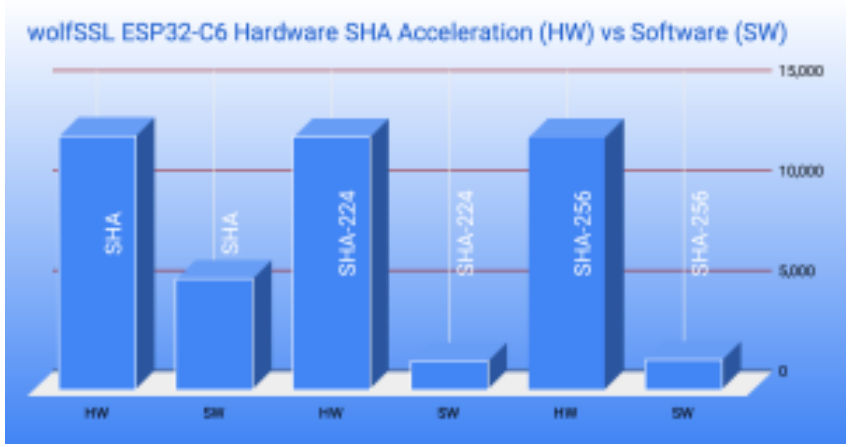


Seguridad en ESP32 – Acelerador criptográfico

Recientemente Espressif ha actualizado el diseño de sus dispositivos, ESP32-v3 para soportar las capacidades de aceleración criptográfica por hardware en las placas RISC-V

Estos modelos son: ESP32-C2, ESP32-C3 y la ESP32-C6.

Estas nuevas funcionalidades permiten utilizar claves y cifrados que incluyen SHA (hash), RSA (matemáticas de números grandes) y cifrado AES.



Esta nueva funcionalidad puede ser utilizada por los desarrolladores para desarrollar sistemas robustos de comunicación, servidores HTTPS y otras medidas de seguridad que antes era imposible realizar en tiempo real.

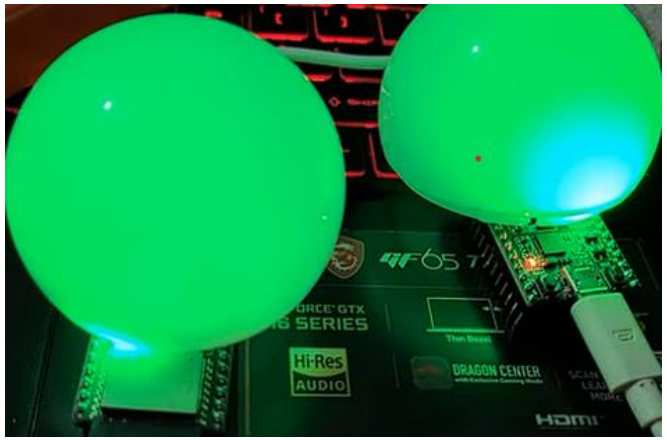
Seguridad en ESP32 – Secure ESPNOW

ESP-NOW es un tipo de protocolo de comunicación Wi-Fi sin conexión definido por Espressif. En ESP-NOW, los datos de la aplicación se encapsulan en una trama y, a continuación, se transmiten de un dispositivo Wi-Fi a otro sin conexión.

ESP-NOW utiliza el método CCMP, que se describe en IEEE Std. 802.11-2012, para proteger el marco de acción específico del proveedor. El dispositivo Wi-Fi mantiene una Clave Maestra Principal (PMK) y varias Claves Maestras Locales (LMK). Las longitudes de PMK y LMk son de 16 bytes.

```
static esp_err_t register_peer(uint8_t *peer_addr)
{
    esp_now_peer_info_t esp_now_peer_info = {};
    memcpy(esp_now_peer_info.peer_addr, peer_addr, ESP_NOW_ETH_ALEN);
    esp_now_peer_info.channel = ESP_CHANNEL;
    esp_now_peer_info.ifidx = ESP_IF_WIFI_STA;
    esp_now_peer_info.lmk = ESP_NOW_LMK;
    memcpy(esp_now_peer_info.lmk ,ESP_NOW_LMK,ESP_NOW_KEY_LEN);
    esp_now_peer_info.encrypt = true;

    esp_now_add_peer(&esp_now_peer_info);
    return ESP_OK;
}
```



Conclusiones

Conclusiones

- ✓ La seguridad en sistemas embebidos empieza a cobrar fuerza e interés político, con regulaciones y protocolos en desarrollo.
- ✓ Existen diferentes tipos de ataque, hemos visto su categorización y algunas estrategias mitigadoras.
- ✓ ESP32 pone a disposición del desarrollador una serie de funcionalidades de seguridad fáciles de implementar y robustas.