MatrikonOPC

ENABLING EMBEDDED UA SECURITY AND DISCOVERY

Liam Power

Principal Engineer, Embedded Systems

STATE OF EMBEDDED TECHNOLOGY

Mountain View, CA

Computer History Museum

Bad iPhone photo of me with a Cray - 1

The concept of “marketing” misunderstood?

1976 Cray – 1 Supercomputer

- $5M - 8M in 1976
- Up to $35.3M adjusted for inflation
- 80MHz CPU
- 160 MIPS (theoretical)
- 250 MFLOPS peak performance
- Approximately 100kW power dissipation
- 5.5 tonnes system weight

2014 – ARM Cortex-M4F Microcontroller

- $5 – 6 in volume
- 168MHz CPU
- 210 DMIPS
- 168 MFLOPS peak performance
- Approximately 200mW power dissipation
- 2lbs system weight

* STM32F407. Single precision FPU. Power dissipation for CPU only.

A golden age for embedded systems

1976 – Cray 1

- $35M in 2014 dollars
- 80MHz CPU
- 160 MIPS (theoretical)
- 250 MFLOPS peak performance
- Approximately 100kW power dissipation
- 5.5 tonne system weight

2014 – ARM Cortex-M4F

- Close to $5 in volume
- 168MHz CPU
- 210 DMIPS
- 168 MFLOPS peak performance
- Approximately 200mW power dissipation
- 2lbs system weight

WHY EMBEDDED UA SECURITY?

Secure your process actuation

- Point to point serial (e.g. MODBUS RTU) being displaced by IP based networks
- IP based networks offer new attack vectors compared to legacy systems
- Will you bet the farm on firewall based security?
- Encryption is optional but authentication is critical

Enable new applications for OPC

- Remote Telemetry Unit (RTU) communicating natively via OPC UA over third party or public networks
  - GPRS, 3/4G, LTE
  - Satellite
  - PSTN
  - Wired or wireless Ethernet
- Would not be possible without authentication and encryption

PLATFORM REQUIREMENTS

Platform requirements for security

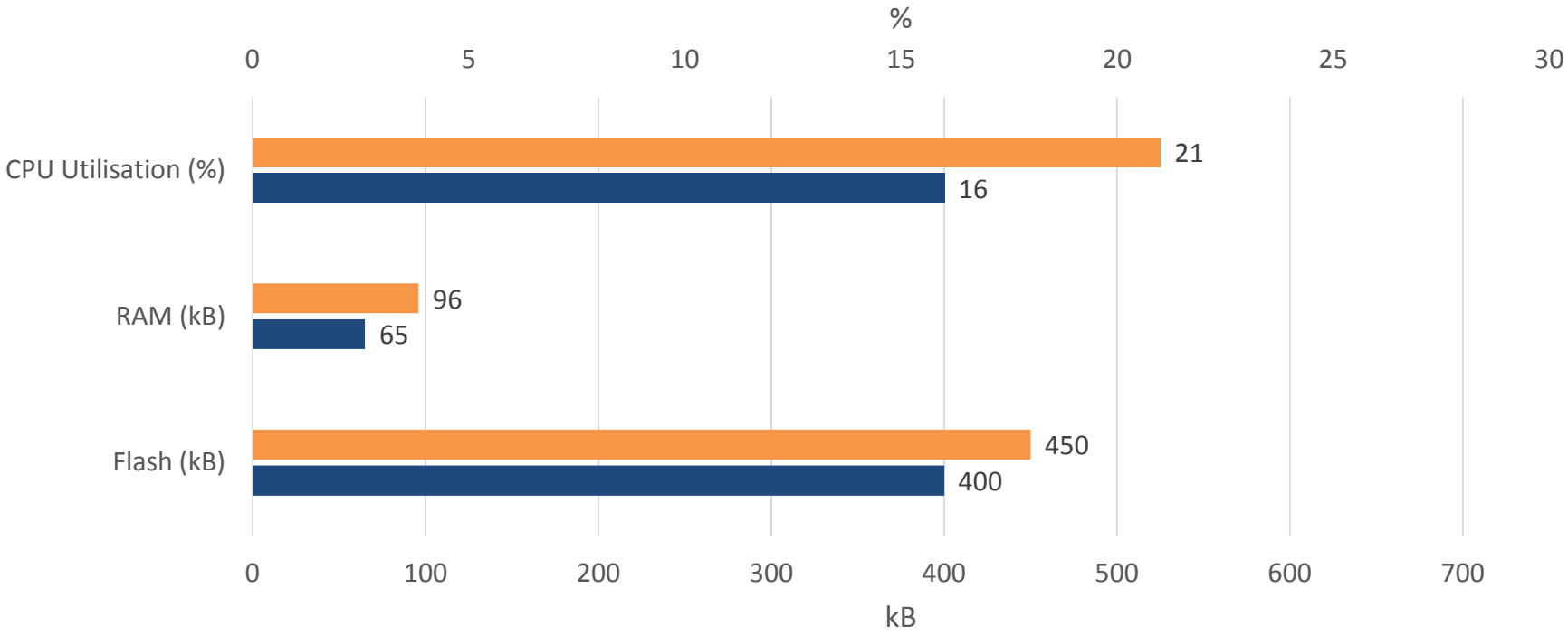
- Typical baseline spec. ARM Cortex-M3 MCU, 100MHz, 1MB Flash, 128kB RAM
- File system recommended but optional (either way private keys must be encrypted)
- Time (via NTP, RTC or other)
- DHCP / Auto-IP support
- For servers an OS is recommended (typically an RTOS as opposed to bare metal)
- Have a good source of entropy

Platform resource utilization example

Authentication & Encryption (Basic128Rsa15)

No Security

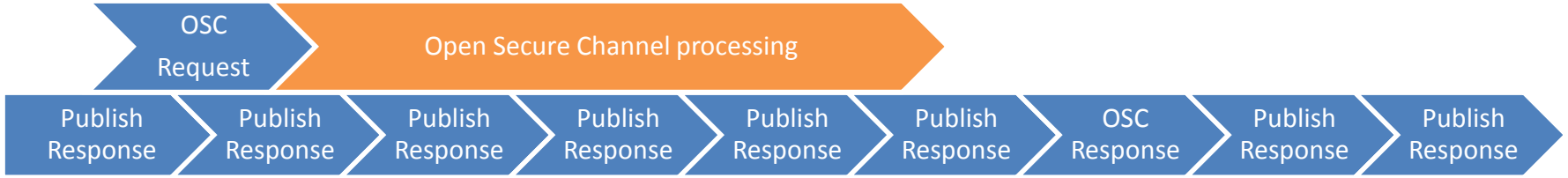
ARM Cortex-M4F STM32F407 @ 168MHz, GCC –O2, executing from internal flash.
CPU utilisation while sampling and publishing 100 continuously changing tags every 100ms



PKI and network jitter

PKI blocks execution when performed in a single thread

No blocking when offloaded to a worker thread



*Not to scale.

VENDOR CONFIGURATION

Names and certificates

- URI and default Hostname must be unique
- URI "urn:dev:mac:0024befffe804ff1"
- Hostname "matrikonopc-sensor-0024befffe804ff1"
- Label the device for the commissioning engineer
- Default application instance certificate provisioning
  - Can be installed in the factory, installed in the field or generated on the device
  - Should be bound to a default IP address or a default hostname
  - If using discovery the IP address is irrelevant

Server secure channels & sessions

- Set bounds for the secure channel lifetime
  - Lower limit restricts the frequency of PKI private key operations (decrypt and sign)
  - Upper limit restricts the window available for an attack on the symmetric keys
  - Typical limits may be 1 – 24 hours depending on your application
  - A theoretical supercomputer that could check 50 $*10^{18}$ AES keys per second would require about 3×10^{51} years to exhaust the 256-bit key space.
- If your application allows, keep session timeouts much smaller than secure channel lifetimes as the UA stack will recycle secure channels that have no associated session

DEPLOYMENT AND COMMISSIONING

Discovery

- Install device and power on
- Device acquires an IP address via DHCP or Auto-IP
- Device announces itself over the network and responds to mDNS network discovery requests
- A device server is now available to connect to at “matrikonopc-sensor-0024befffe804ff1.local”
- A device client now has a list of discoverable servers on the network

Server security provisioning

- Change the device hostname to a more meaningful name if required
- Update the server application instance certificate if necessary or if the device does not have a default certificate install one
- Connect securely to the server from an OPC UA configuration client (or GDS)
- Install self signed client certificates and/or CA certificates in the trust list using the configuration client or GDS push

Client security provisioning

- Change the device hostname to a more meaningful name if required
- Update the client application instance certificate if necessary or if the device does not have a default certificate install one
- Connect securely to a GDS or provision the client via an out of band configuration interface
- Install self signed server certificates and/or CA certificates in the trust list using the configuration interface or GDS pull

DEVICE MANAGEMENT

Self-signed versus CA issued certificates

- Self-signed
  - No need for CRLs (Certificate Revocation Lists)
  - Device trust list must be updated for every new application that requires a connection
  - Device trust list can become very large over time increasing Flash and RAM requirements
- CA issued
  - Only a single CA certificate required to be installed in the trust list
  - Any number of applications can now connect to the device with no additional configuration
  - Bounded Flash and RAM requirements
  - CRLs now become important and the device CRL(s) should be updated regularly

Security error suppression

- Based on a risk assessment set management policies as follows
  - How to handle expired certificates (both our application certificate and those of applications we are connected to)
  - Whether or not to enforce hostname validation
  - Whether or not to require use of CRLs
- Incorrectly set policies can have consequences
  - Security breach due to a revoked client certificate and no CRL lookup on a server device
  - Y2K type problem caused by forgotten RTU with an expiring certificate

Key Takeaways

- Secure & Discoverable Embedded UA is an incredibly powerful tool that can
  - Reduce system installation and commissioning cost
  - Secure existing infrastructure
  - Enable new applications that were not practical before
- Security & discovery are perfectly suited to low cost chipsets
- Always authenticate but only encrypt if it makes sense
- Embedded UA Security can be easy to deploy and manage
- PKI security requires some administration. Ensure end users clearly understand what they need to do to keep their infrastructure secure over time