The Pros and Cons of Using an Embedded FIPS Module vs. Validating an Entire Product

Acumen Security

Agenda

- What is it?
- What are some considerations?
- Getting started

Imagine This (if you can)

What's the Problem?

- You're non-compliant
- You don't have the time
- Pressure….Lots and lots of pressure

Here's the solution –

Well, maybe!!!

What's to it?

- Specific to the crypto implementation
  - Usually a tool kit
- Tight crypto boundary
- Called on a service-by-service basis

NOT ONE SIZE FITS ALL!!!

Considerations

Acumen Security

Embedded Modules (the case for)

- Time to market
  - Weeks vs months/year
- Cost
  - Order of magnitude less expensive
- Level of Effort
  - Exchanging a component vs numerous bugs/coordinating with labs/Gov't
- Maintenance
  - Product software updates
- Scalability
  - Multiple product lines
  - Small teams
- Sales
  - Meets the requirement to use FIPS crypto

Porting (what can you do)

FIPS 140-2 Implementation Guidance G.5

For Level 1 Operational Environment, a software cryptographic module will remain compliant with the FIPS 140-2 validation when operating on any general purpose computer (GPC) provided that the GPC uses the specified single user operating system/mode specified on the validation certificate, or another compatible single user operating system.

Firmware modules (i.e. Operational Environment is not applicable) that do not require any source code modifications (e.g., changes, additions, or deletions of code) to be recompiled and its identified unchanged tested operating system (i.e. same version or revision number) may be ported together from one GPC or platform to another GPC or platform while maintaining the module's validation.

Why would anyone do it any other way??

Embedded Modules (the case against)

- Coverage
  - Some functionality just may not be there
  - Individual services may not be drop in compatible
- Relevance
  - Historical list
  - Dated algorithms
  - May not meet other certification requirements
- Maintenance
  - Rely on someone else to keep the module up-to-date
  - Bugs!
- Marketing
  - Name on website

Missed Requirement Examples

FCS_CKM.1.1 The TSF shall generate asymmetric cryptographic keys in accordance with a specified cryptographic key generation algorithm: [selection:

- ***RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3;***
- ECC schemes using "NIST curves" [selection: P-256, P-384, P-521] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4;
- FFC schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.1]

What's Missing?

- Key management
  - Most embedded modules are just tool kits
- Key derivation functions
  - Typically handled by protocol implementations outside of the module boundary
- Entropy

Now what?

Acumen Security

Develop Your Strategy

- Why are you considering it?
- What are your competitors doing?
- How much time do you have?
- What services use crypto?
- Other certifications beyond FIPS 140?

Commonly Used Embedded Modules

- Operating System Modules
  - Apple
  - Linux
  - Microsoft
- Open Source Modules
  - OpenSSL
  - NSS
- Private Label Modules
  - Safelogic
  - WolfSSL
  - Mocana
  - RSA

Hybrid Approaches

- Progressive:
  - Start open source
  - Move to full validation
- Roll your own:
  - Privately branded/owned embedded module
- Pick and Choose:
  - Only claim “compliance” on the services customers care about

Summary

- Described what it is.
- Discussed some considerations when deciding to use an embedded module
- Reviewed some steps for getting started

Questions?

Acumen Security

Thank you!

Acumen Security