FIPS 140-2 Crypto
In the IoT

Chris Conlon
ICMC17, May 16-19, 2017
Westin Arlington Gateway | Washington DC

Outline

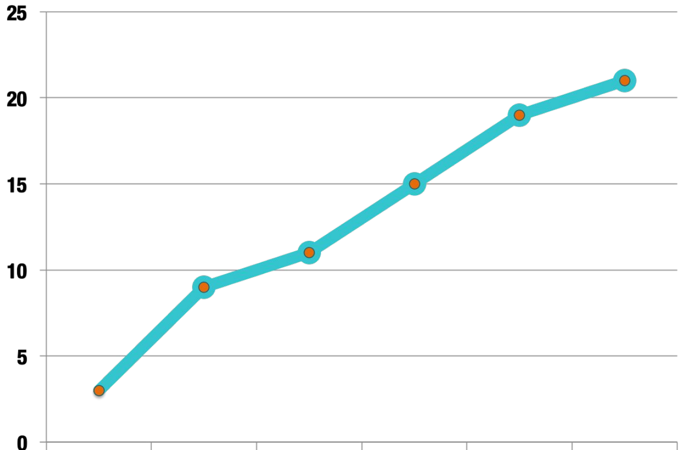
A. Intro to wolfSSL
B. Overview of wolfCrypt FIPS
C. FIPS 140-2 Challenges in the IoT
D. Doing new validations
E. Q&A

Overview

wolfSSL and wolfCrypt FIPS

Introduction to wolfSSL

2004	2004 / 2006	2014
Needs “Clean Room” SSL	2004 - **yaSSL** (C++) 2006 - **CyaSSL** (C)	wolfSSL Name Change



	2011	2012	2013	2014	2015	2017
Employees	3	9	11	15	19	21

450 OEM Customers

15 Resale Partners

2 BILLION

secure connections!

Introduction to wolfSSL - Products

wolfSSL
Lightweight SSL/TLS

wolfCrypt

SSL Sniffer

wolfCrypt
Crypto Engine

wolfCrypt FIPS

wolfMQTT
Lightweight MQTT Client

wolfSSH
Lightweight SSH Server

wolfSSL JNI
wolfSSL Java Wrapper

wolfSSL C#
wolfSSL C# Wrapper

wolfSSL Python
wolfSSL Python Wrapper

SSL Proxy
On top of Squid Proxy

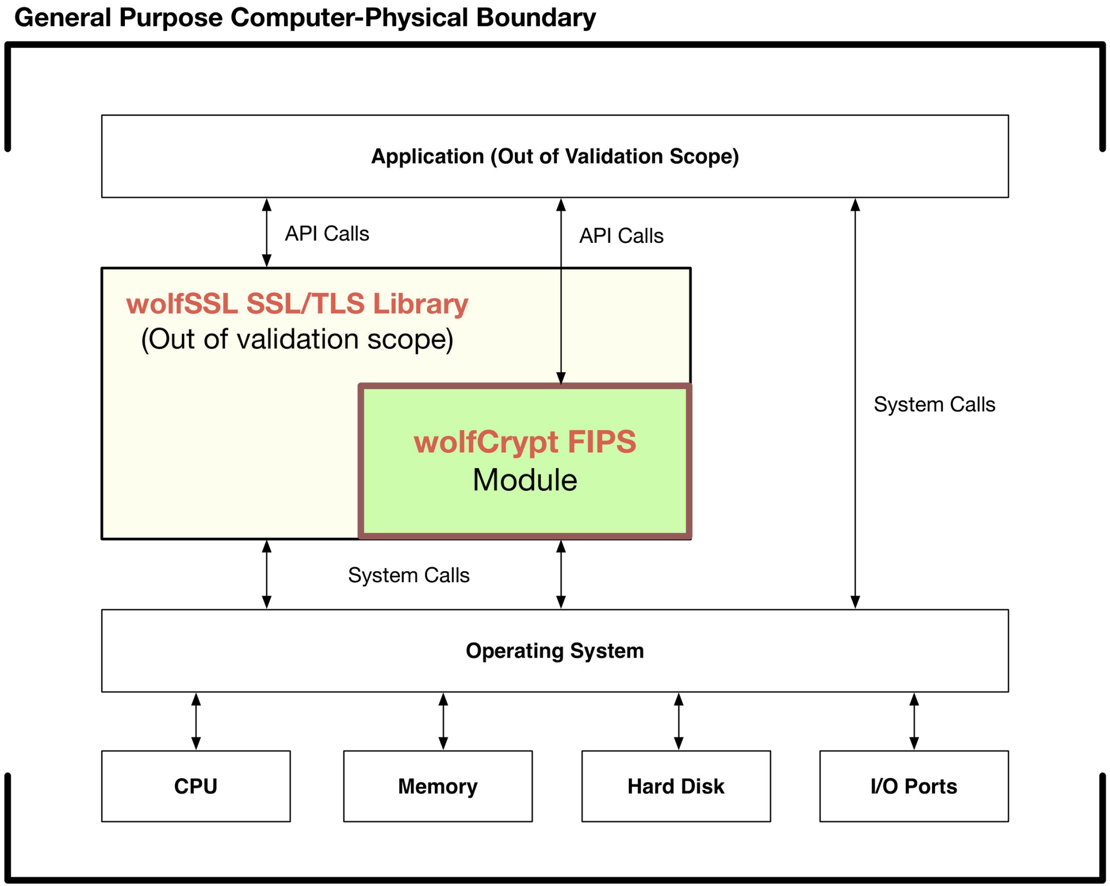
Secure memcached

wolfSCEP

Secure Update

wolfCrypt FIPS Object Module

- **Independent of SSL/TLS**
- **Design simplifies updates**
- **Most bugs and vulnerabilities happen in SSL/TLS, not crypto**



Current wolfCrypt FIPS OE List

Certificate #2425

	Operating System	Processor	Platform
1	**Linux 3.13** (Ubuntu)	Intel® Core™ i7-3720QM CPU @2.60GHz x 8	HP EliteBook
2	**iOS 8.1**	Apple™ A8	iPhone™ 6
3	**Android 4.4**	Qualcomm Krait 400	Samsung Galaxy S5
4	**FreeRTOS 7.6**	ST Micro STM32F	uTrust TS Reader
5	**Windows 7** (64-bit)	Intel® Core™ i5	Sony Vaio Pro
6	**Linux 3.0** (SLES 11 SP4, 64-bit)	Intel® Xeon® E3-1225	Imprivata OneSign
7	**Linux 3.0** (SLES 11 SP4, 64-bit) on Microsoft Hyper-V 2012R2 Core	Intel® Xeon® E5-2640	Dell® PowerEdge™ r630
8	**Linux 3.0** (SLES 11 SP4, 64-bit) on VMWare ESXi 5.5.0	Intel® Xeon® E5-2640	Dell® PowerEdge™ r630
9	**Windows 7** (64-bit) on VMWare ESXi 5.5.0	Intel® Xeon® E5-2640	Dell® PowerEdge™ r630

Approved and Validated Crypto Functions

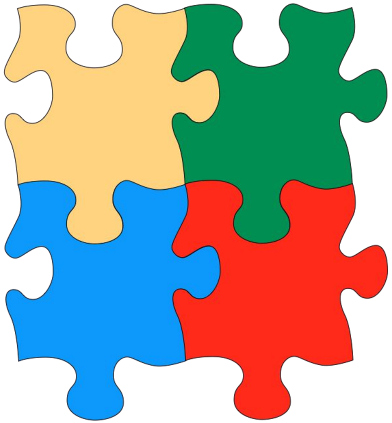
Algorithm	Description	Cert #
AES	[FIPS 197, SP 800-38A] (Encryption, Decryption) Modes: CBC, CTR, Key sizes: 128, 192, 256 bits	3157, 3330, 3417, 3490, 3508
DRBG	[SP 800-90A] (Hash_DRBG) Security Strengths: 256 bits	650, 775, 821, 863, 875
HMAC	[FIPS 198-1] (Generation, Verification) SHA sizes: SHA-1, SHA-256, SHA-384, and SHA-512	1990, 2121, 2175, 2228, 2241
RSA	[FIPS 186-4, and PKCS #1 v2.1 (PKCS1.5)] (Signature Generation, Signature Verification) Key sizes: 1024 (verification only), 2048	1602, 1710, 1749, 1791, 1803
SHA	[FIPS 180-4] (Digital Signature Generation, Digital Signature Verification, non-Digital Signature Applications). SHA sizes: SHA-1, SHA-256, SHA-384, SHA-512	2614, 2763, 2823, 2882, 2893
Triple-DES (TDES)	[SP 800-20] (Encryption, Decryption) Modes: TCBC, Key sizes: 3-key	1800, 1901, 1928, 1966, 1972

FIPS 140-2

Challenges in the IoT

FIPS 140-2 Challenges in the IoT

- **Predominant challenges include:**
  - Porting default shared library entry point
  - Running CAVP test vectors
  - Fitting FIPS module into available memory
  - Porting library to target environment



Porting Default Entry Point

- **When library is first loaded, two things need to happen:**

  1. Power-On Integrity Check
  2. Run Known Answer Tests

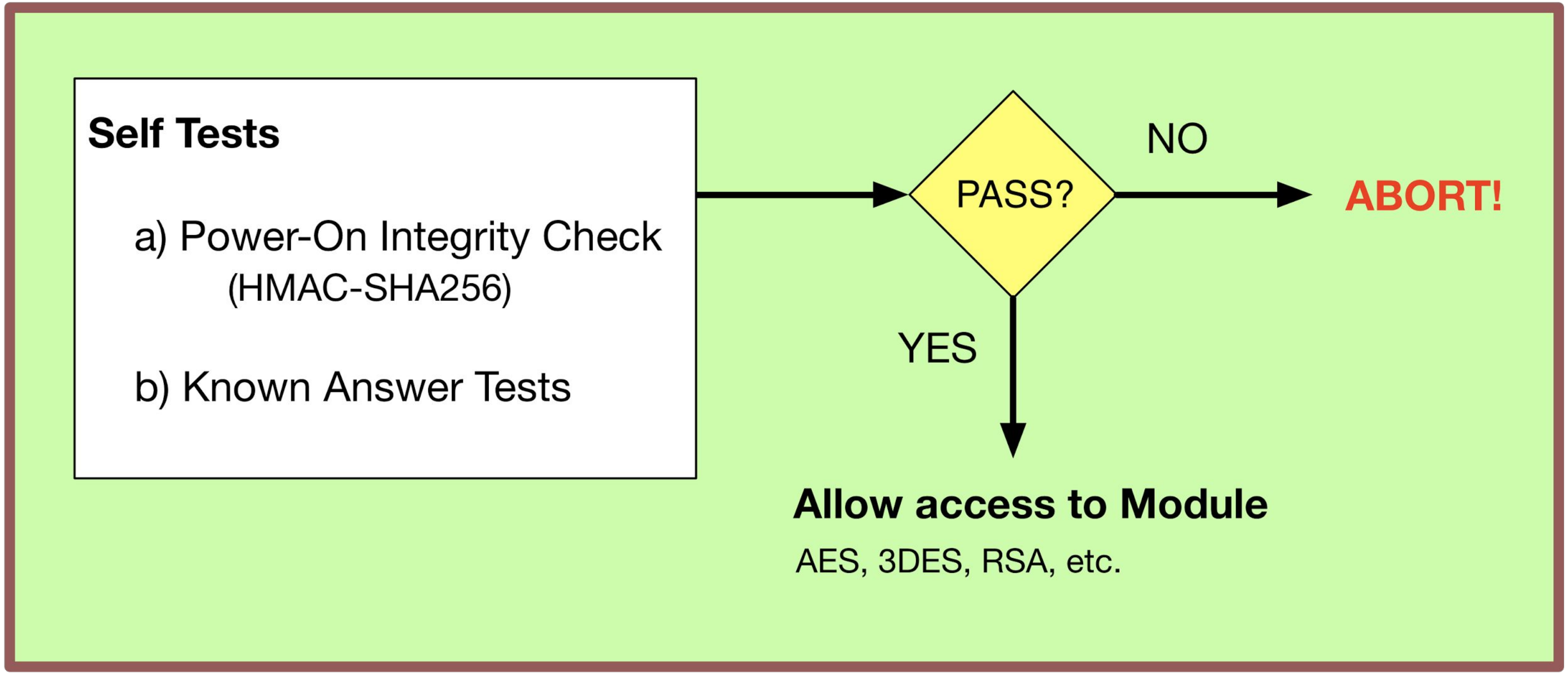
- **Shared library default entry point is used for this**

```
#define INITIALIZER(f) static void __attribute__((constructor)) f(void)
```

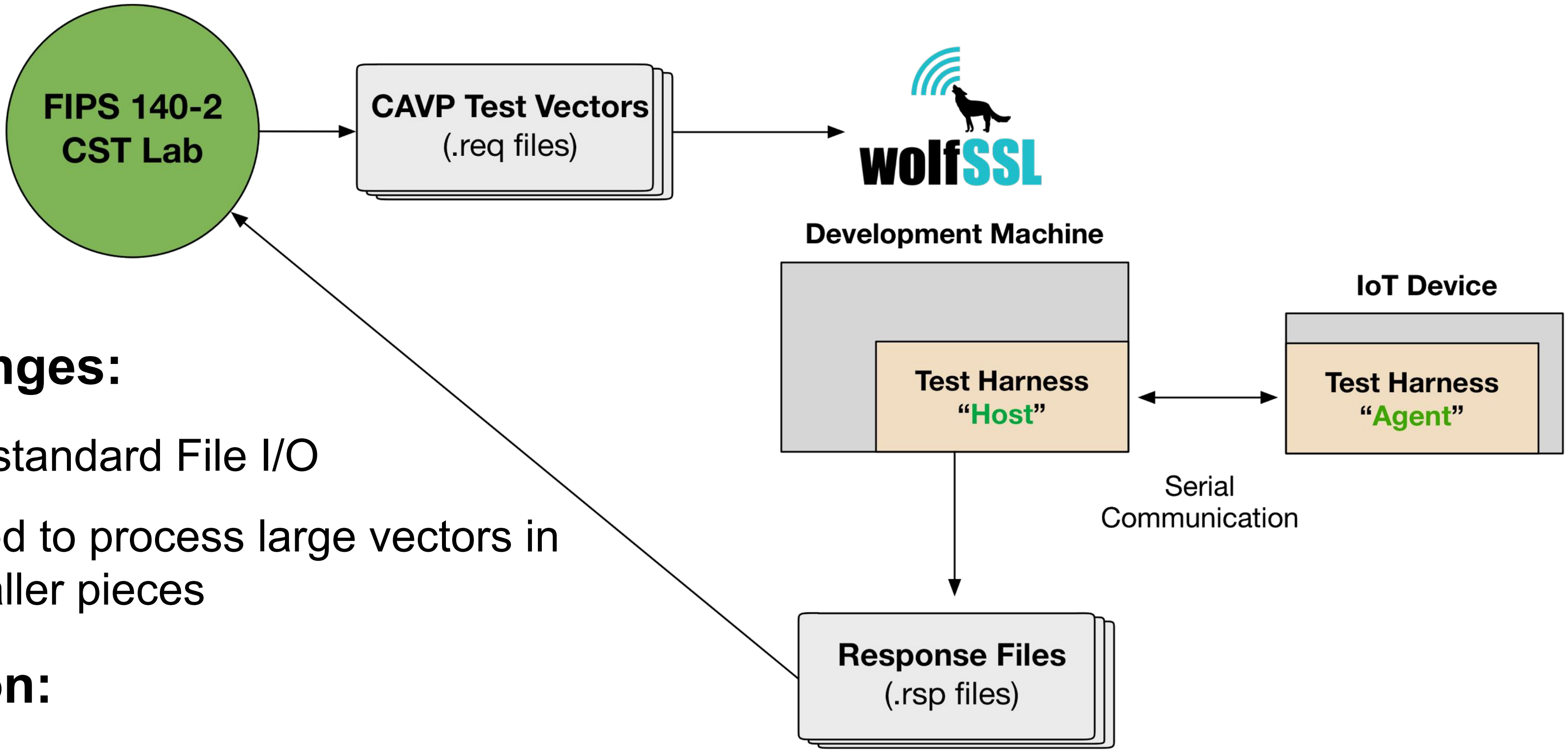
- **Needs to be ported on new compiler/linkers** (gcc, VS, etc)

FIPS 140-2 Module Runtime Requirements

wolfCrypt FIPS Module



Running CAVP Test Vectors



Challenges:

- No standard File I/O
- Need to process large vectors in smaller pieces

Solution:

- Test Harness (Host/Agent)

Fitting FIPS Module into Memory

- **IoT Device Memory Constraints Pose a Challenge**

- **Mitigations / Resolutions**
  - Configure algorithms differently (speed vs. size)
  - Configure memory usage differently (stack vs. heap)
  - Shrink module boundary

Porting Library to Target Environment

- **IoT Devices Pose Portability Concerns**
- **Platform Details Can Vary:**
    - Variety of RTOS's
    - Different toolchains / compilers
    - Memory configurations (stack vs. heap preference)
    - Threading / Mutexes
    - Seeding PRNG / sources of randomness
- **wolfCrypt platform-dependencies have been abstracted out**

FIPS 140-2

Doing New Validations

New FIPS 140-2 Validations

- **Validation Options:**
  - Adding new Operating Environment (OE)
  - Rebranding Validation
  - Growing (or Shrinking) Module Boundary

- **Timeframe dependent on scope, platform, and lab/CMVP**

Adding a New Operating Environment

Step 1: CAVP Testing and Algorithm Certificates

1. Define desired cryptographic module boundary
    - From customer:
        - **Exact platform** (hardware, OS version)
        - **Example app** demonstrating I/O from device (for test harness)
2. **Port** and **test** module and harness on desired validation target
3. **Request** test vectors from FIPS Lab
4. **Run** test vectors through module, return to FIPS Lab
5. Obtain **Algorithm Certificates** from CAVP

Adding a New Operating Environment

Step 2: CMVP and FIPS 140-2 Certificate

1. Update **Security Policy**, send to FIPS Lab
2. **On-site** testing at FIPS Lab with module
3. FIPS Lab submits to **CMVP… wait…**
4. **FIPS 140-2 Certificate Issued**, or **Existing Updated**

CMVP Main Page

Validated FIPS 140-1 and FIPS 140-2 Cryptographic Modules

Historical, 1995-1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017

All

wolfCrypt FIPS Rebranded Validations

- **Rebranded wolfCrypt FIPS validations possible**
- **One recent IoT-based example - reduced FIPS boundary**

Operating System	Processor
OpenRTOS v9.0.0	ATSAM4L

Algorithm	Description
AES	[FIPS 197, SP 800-38A] (Encryption, Decryption) Modes: CBC, CTR, Key sizes: 256 bits
HMAC	[FIPS 198-1] (Generation, Verification) SHA sizes: SHA-256
SHA	[FIPS 180-4] (Message Digest) SHA sizes: SHA-256

Summary

A. **FIPS 140-2 Challenges in the IoT**
   a. Porting default shared library entry point
   b. Running CAVP test vectors
   c. Fitting FIPS module into available memory
   d. Porting library to target environment

B. **Doing new validations**
   a. Adding a new OE
   b. Rebranded Validation
   c. Growing (Shrinking) Module Boundary

Thanks!
Questions?

info@wolfssl.com
www.wolfssl.com