# CrypTech

## Building a More Assured HSM with a More Assured Tool-Chain

# Hardware Security Module

# The Need

- Every week a new horror about Crypto/Privacy
- *der Spiegel's* revelation of the "SpyMall Catalogue"
- Compromises of most network devices, servers, firewalls, ...
- We are relying on HSMs which are designed and made by government contractors
- Many people are not comfortable with this

# HSMs Are Used For

- Principally, Lock-box for Private Keys for
  - DNSsec
  - RPKI
  - PGP
  - Your use goes here
- Also,
  - Encryption / Decryption
  - VPNs
  - Source of Randomness

# Origins

- This effort was started at the suggestion of Russ Housley, Stephen Farrell, and Jari Arkko of the IETF, to meet the assurance needs of supporting IETF protocols in an open and transparent manner.

- But this is NOT an IETF, ISOC, ... project, though both contribute. As the saying goes, we work for the Internet.

# Goals

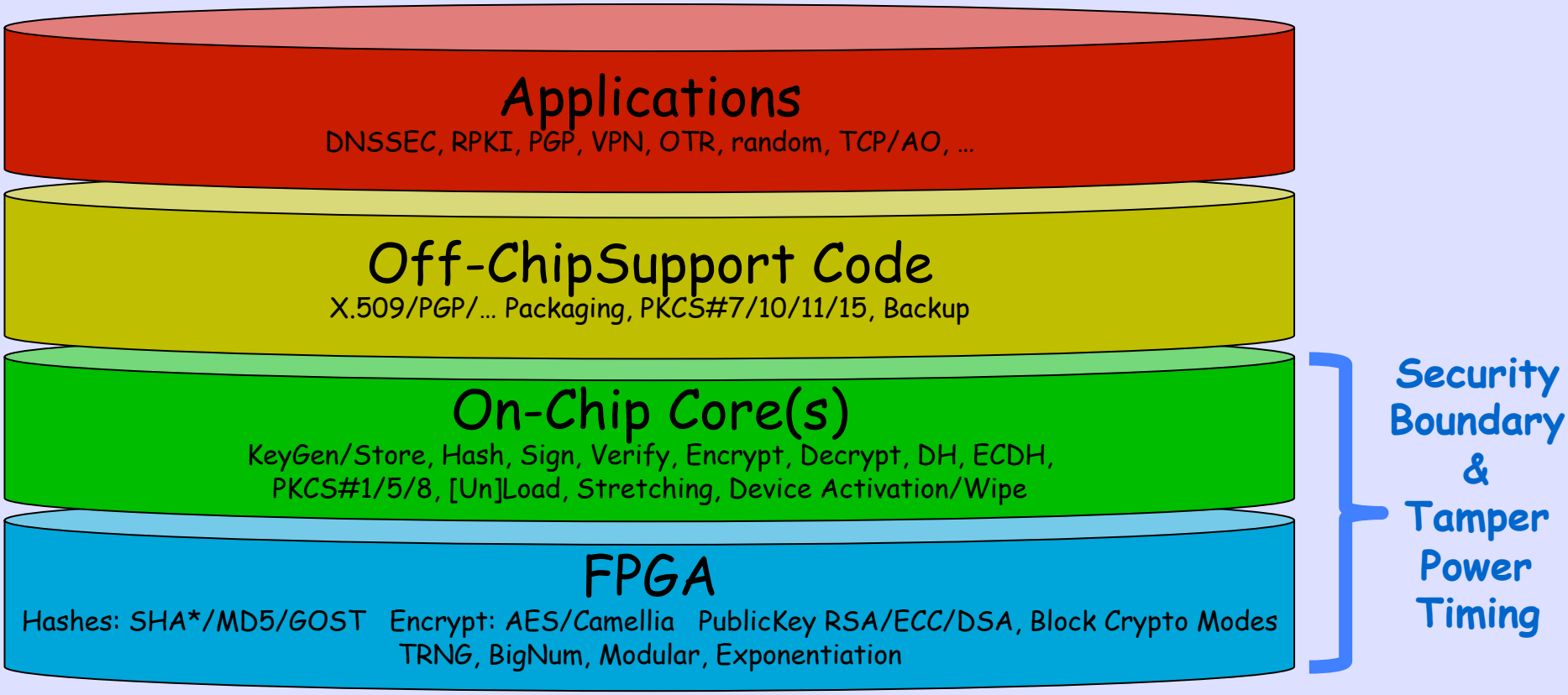
- An open-source reference design for HSMs
- Scalable, first cut in an FPGA and CPU, later allow higher speed options
- Composable, e.g. "Give me a key store and signer suitable for DNSsec"
- Reasonable assurance by being open, diverse design team, and an increasingly assured tool-chain

# Open and Transparent

- The project is being run in a maximally open, transparent manner with traceability for all decisions etc.

- We do this in order to build trust in the project itself

# Layer Cake Model



# The Tool Chain

- When my laptop's fan goes on, I think it is the NSA, GCHQ, Israelis, Chinese, … fighting to see who will own me
- We have NO ASSURANCE of our tool set, from CPU to Kernel to Compiler to …
- When constructing assurance-critical tools, we need to maximize assurance in the tools used to build them

# The Compiler

- Ken Thompson's 1984 Turing Award paper *Reflections on Trusting Trust*

- A self-reproducing trap in the C compiler which "would match code in the UNIX "login" command. The replacement code would miscompile the login command so that it would accept either the intended encrypted password or a particular known password." You have been owned!

# Double-Diverse Compilation

- In his 2009 PhD dissertation, David Wheeler explained how to counter the "trusting trust" attack by using the "Diverse Double-Compiling" (DDC) technique
- We can use this on GCC and clang to get somewhat assured compilers
- But you still have to inspect the source!

# Some Phases

- First Year: Tool-chain, Basic Design, not all cyphers, not all protocols, prototype implementations on FPGAs and boards
- Second Year: Better Tool-chain, all needed cyphers, hashes, crypting, … and integration with some apps, DNSsec, RPKI, TLS, PGP, Tor
- Third Year: Solid packaging, ability to compose designs for use cases, etc.

# We Seek Review

- We seek, expect, and encourage any form of open and transparent review
- We will not (soon) seek certification as cost/benefit is high
- But we encourage/expect others to take the designs down that path
- We will document all security claims

# Minimal Organisation

- Finances at a non-profit in Sweden associated with NORDUNET
- Administration at SUNET, Maria Hall and Leif Johanssen
- Technical: cooperative of very senior folk with Randy Bush as cat herder
- Fund Raising – All of Us, You Too!

# Diversity Improves Trust

- Diverse Technical Team - from diverse countries / environments
- Transparent Development - code, designs, documentation all public
- Auditable and Audited – Please help audit

# Diverse Funding

- Multinational and Multi-Stakeholder
- Industry, Academe, Social, …
- Diversity is critical, no donor > 10%

https://cryptech.is/