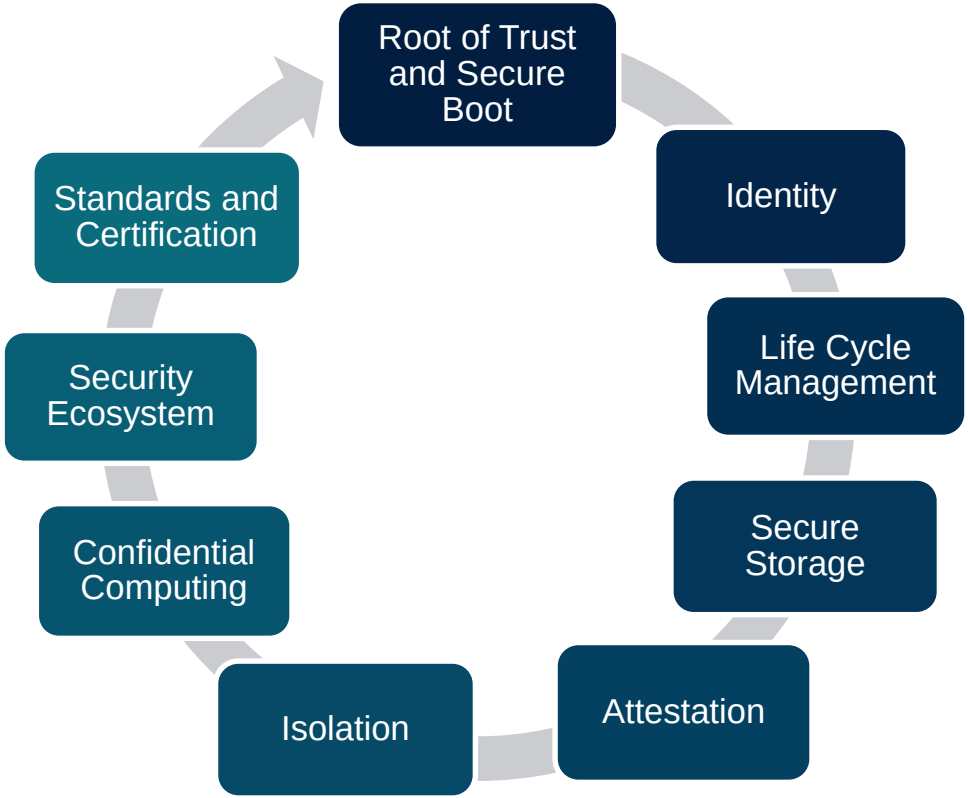
Risc-V Security

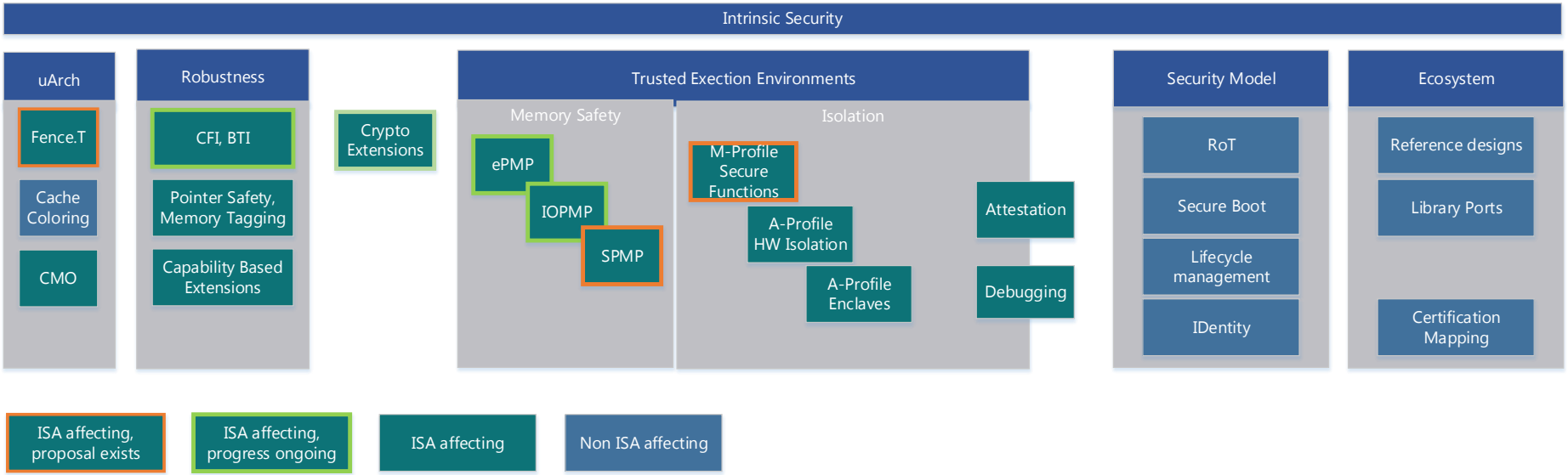
Intrinsic Security

Zero Trust Model

- Security as a basic feature of HW, SW Firmware
- Support security through entire lifecycle
- Guidelines matched to profiles



RISC-V®


Intrinsic Security
uArch
Fence.T
Cache Coloring
CMO
Robustness
CFI, BTI
Pointer Safety, Memory Tagging
Capability Based Extensions
Crypto Extensions
Trusted Exection Environments
Memory Safety
ePMP
IOPMP
SPMP
Isolation
M-Profile Secure Functions
A-Profile HW Isolation
A-Profile Enclaves
Attestation
Debugging
Security Model
RoT
Secure Boot
Lifecycle management
IDentity
Ecosystem
Reference designs
Library Ports
Certification Mapping
ISA affecting, proposal exists
ISA affecting, progress ongoing
ISA affecting
Non ISA affecting

Security Model

- State Goals of Security and Rationale
- Abstracted from implementation specifics
- Platform specifications can reference appropriate sections
  - By Profile, By Vertical
- Reuse Existing standards when appropriate

Security Model

Secure Boot

- Immutable Root of Trust
- Secure Boot Chain
- Mutable Root of Trust
- Attestation through every stage
- Temporal Isolation
- *Certificate Format*
- *Binary Format*
- ….

LifeCycle Management

- Development and Debug
- Provisioning
- IDentity
- Update
- Anti Roll back
- Decommisioning

- Intention is to reference from Platform Specifications
  - Shall - Should - May in platform spec
  - Security Model itself is Non-ISA affecting

Trusted Execution

- Memory Safety
  - PMP based TEEs
- Hardware based Isolation
  - M-Class secure functions
  - A-Class Maintaining privilege levels in TEE
- Confidential Compute
  - Mutually untrusted applications
  - Isolated Application Enclaves
  - Encrypted Enclaves
- Attestation of entire software chain from initial boot
- Security aware Debugging

Maybe ISA affecting / **ISA Affecting**

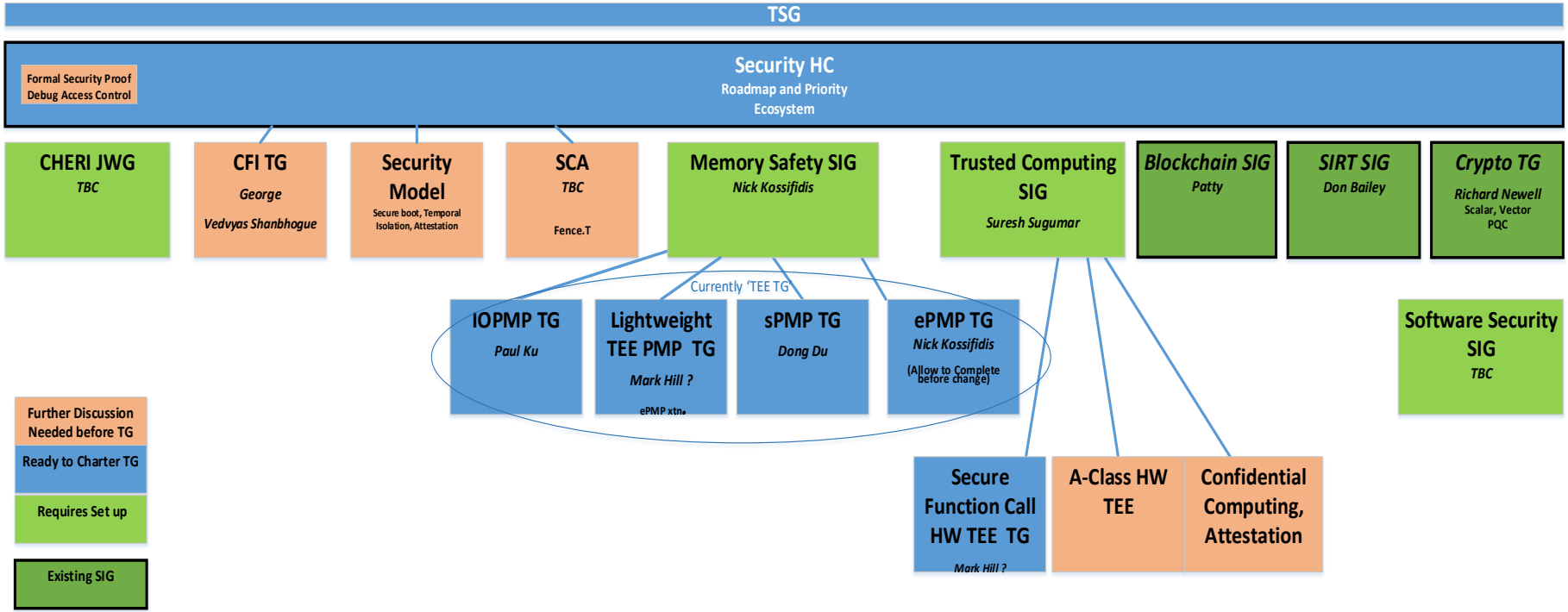
Robustness

- Control Flow Integrity
  - BTI, PAC etc
- Side Channel Leakage Protection
  - ISA extensions to control uArch ?
    - Fence.t
- Capability Based
  - CHERI JWG
  - Alternatives ?

Security Ecosystem

- Security Services
  - Reference implementations
    - Secure Boot
    - Attestation
    - …
  - Optimised library ports
    - OpenSSL
    - mbed Crypto
  - OS ports
  - APIs
- Standards and Certification
  - Formal Security Proofs
  - Certification mapping
- Verticals
  - Any Specific requirements ?

Proposed Reorganisation of Security in Risc-V



Proposed Reorganisation of Security in Risc-V

- Start *Security Model* team under HC to create high level recommendations around RoT, secure boot, attestation chain, temporal isolation, updateability, identity, and the rationale.
- Move most of the Current TEE TG work to a new SIG ‘Memory Safety SIG’ -
  - Meetings structure unchanged, chair remains Nick
  - Create individual TGs for each task with specific charter
- Refocus the Security Tech SIG
  - Suresh remains chair
  - Trusted Execution and Confidential Computing
  - Initial work is to define the requirements, aim to charter TG(s)
- Create Joint Working Group (JWG) for CHERI discussion
- CFI ready to charter TG
- Restart SCA discussion on fence.T and other proposals
- Also need focus on Security SW and Ecosystem – any volunteers ?