Mobile Hardware Security

Vikas Chandra and Rob Aitken

ARM R&D

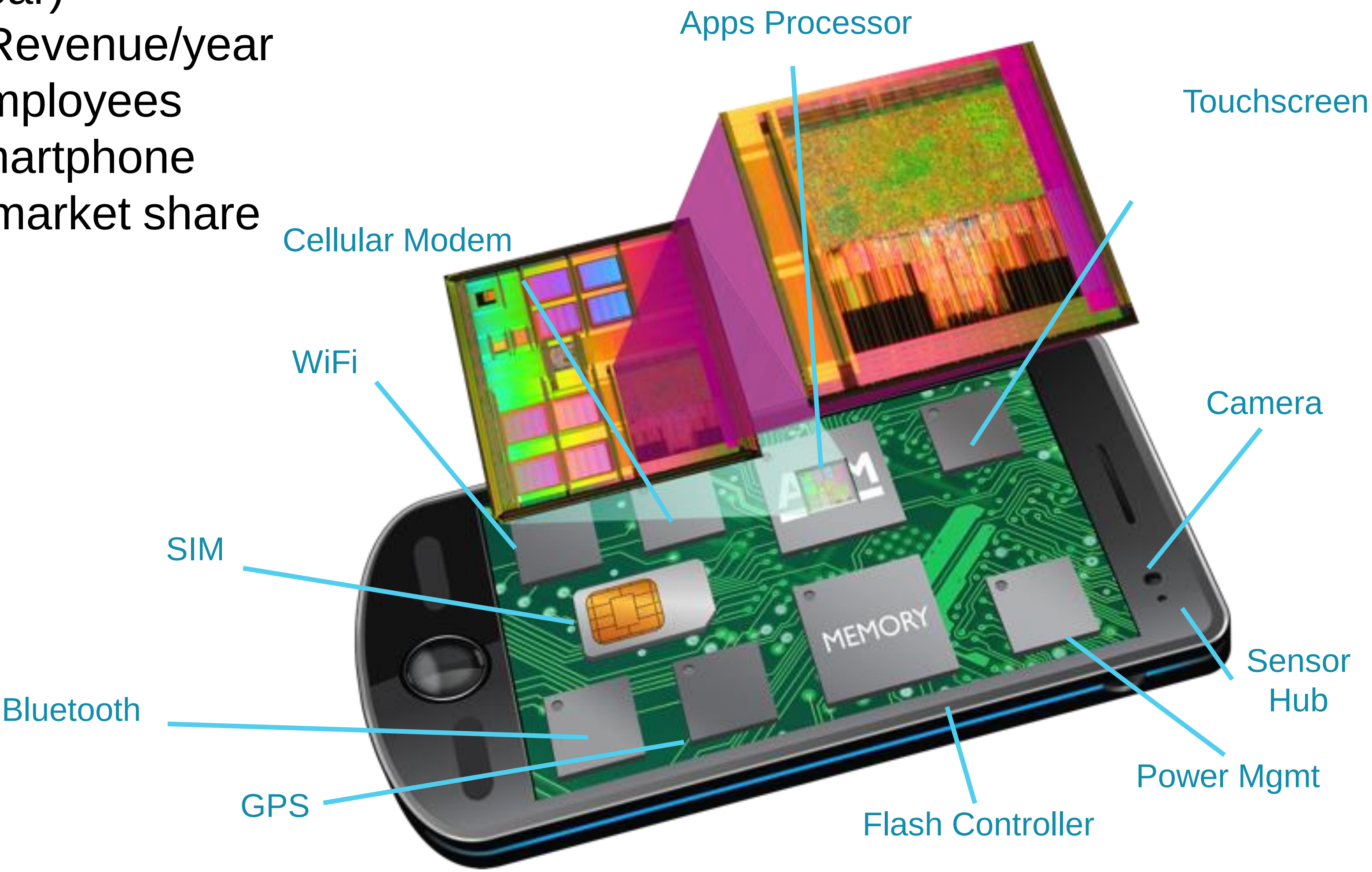
Hot Chips, August 2014

Outline

- Mobile security background
- Trusted Execution Environment
- ARM TrustZone™
- Implementation and use cases
- Authentication

About ARM…

- 50 Billion ARM-based chips shipped (>10B /year)
- ~ $1.2B Revenue/year
- ~3000 Employees
- >95% Smartphone & Tablet market share



ARM

The Mobile Threat Environment

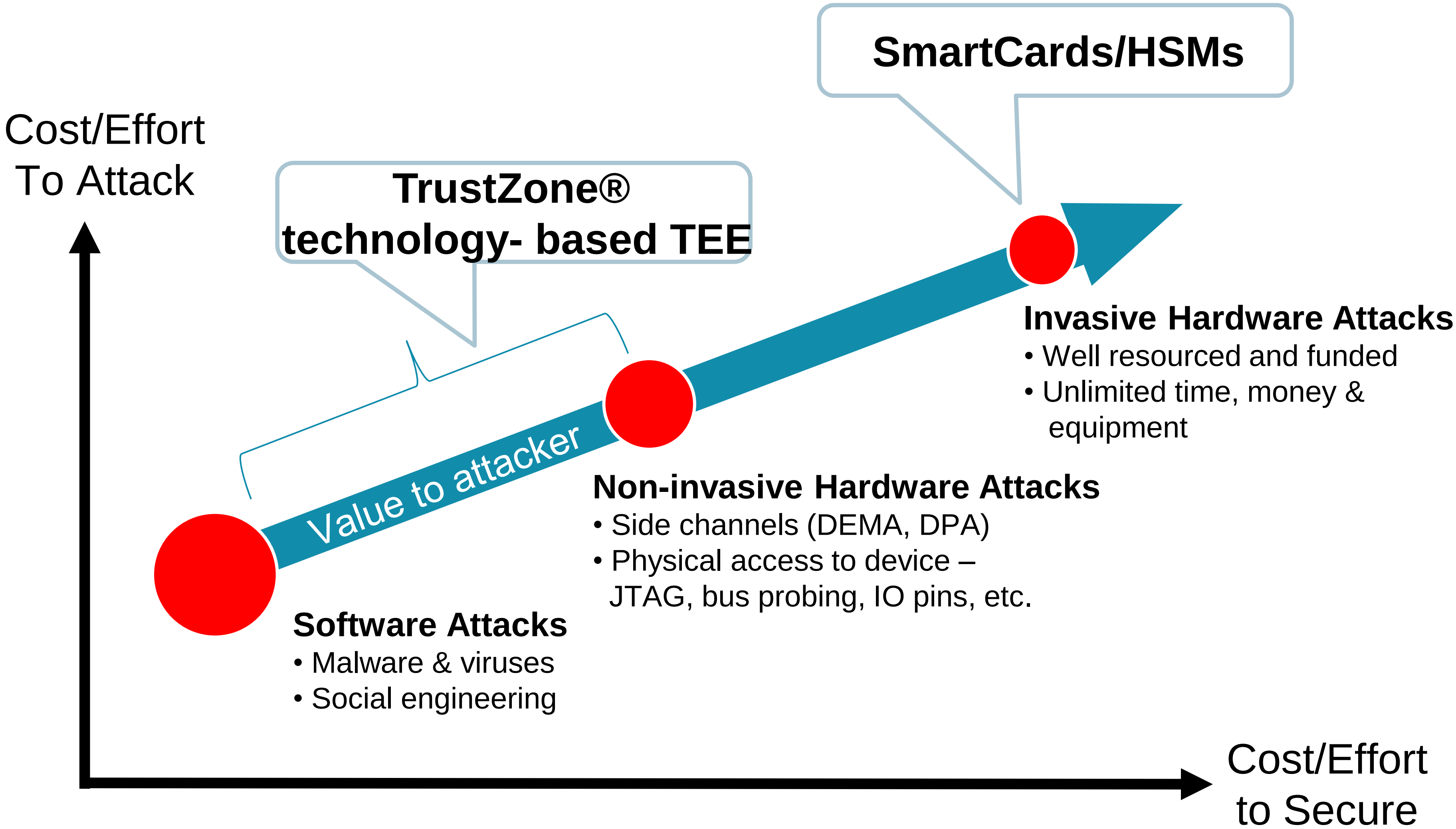
- Increasing risks
  - Social engineering – Trojans, phishing, APT
  - Malware
  - Physical loss or theft leading to risk to data – calendar, phonebook and email
  - Improperly secured devices – no PIN lock
  - User intervention – jailbreaking, unlocking
  - Mobile has become the enterprise security boundary

Need to design in the right system-wide security (not just more security)

Whose Data Is Involved?

- User
  - Personal information, contacts, location, photos, etc.
- Enterprise(s)?
  - Bring your own device (BYOD)
- Carrier
  - Network interface
- Apps
  - Content providers
    - DRM for movies, songs, etc.
  - Finance companies
    - Account data, passwords
  - IoT
    - Home automation, health, etc.

Security Profiles

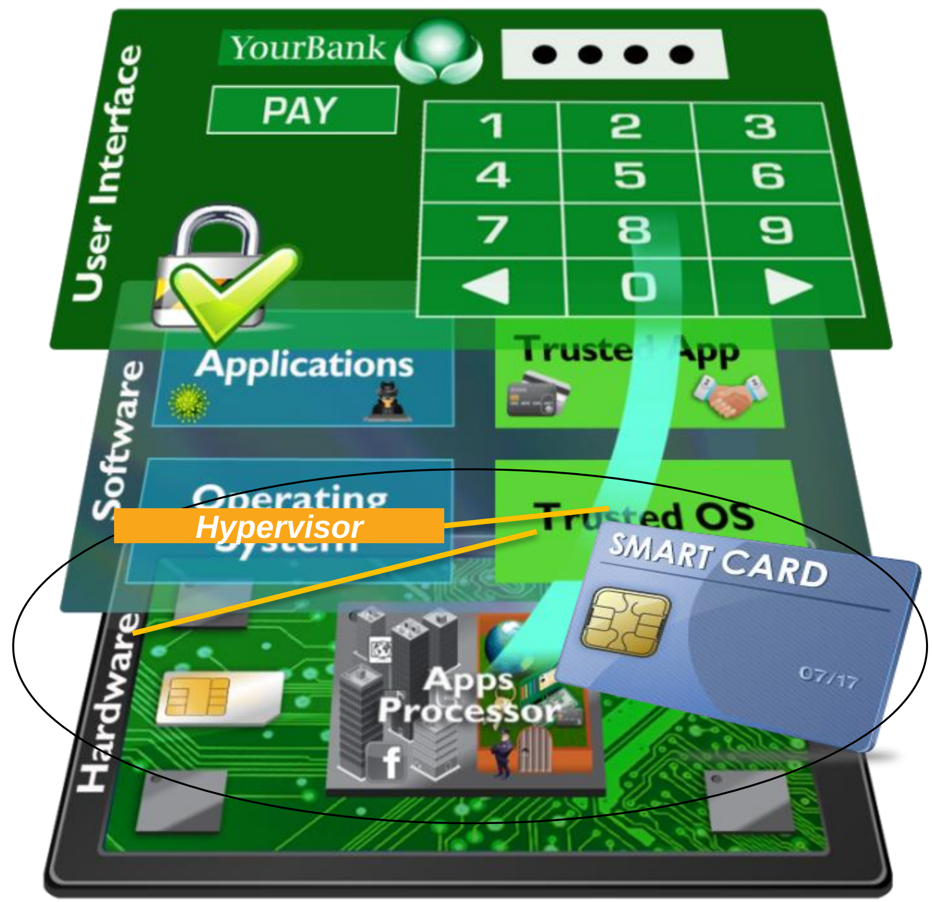


Mobile Solution Is Not PC Solution

- PC-era security
  - Add layers of software security (SSO, etc.)
  - Add hardware security (CVC, key fobs, etc.)
- Too unwieldy and confusing for mobile environment

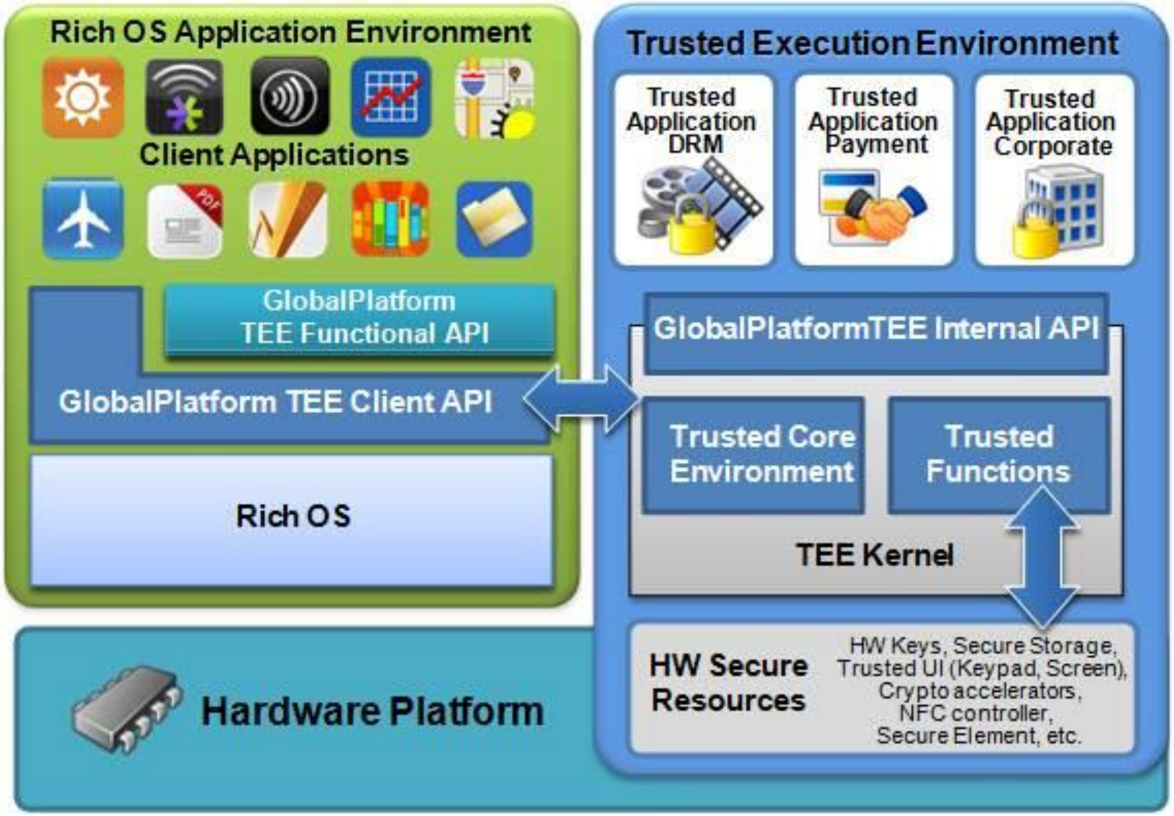
Mobile Security Approach

- Hypervisor (with hardware support) separating large pieces of code
- Small, certifiable Trusted Execution Environment (TEE) inside application processor isolated using ARM TrustZone technology protecting against software attacks
- Secure element for tamper-proof security (where needed)



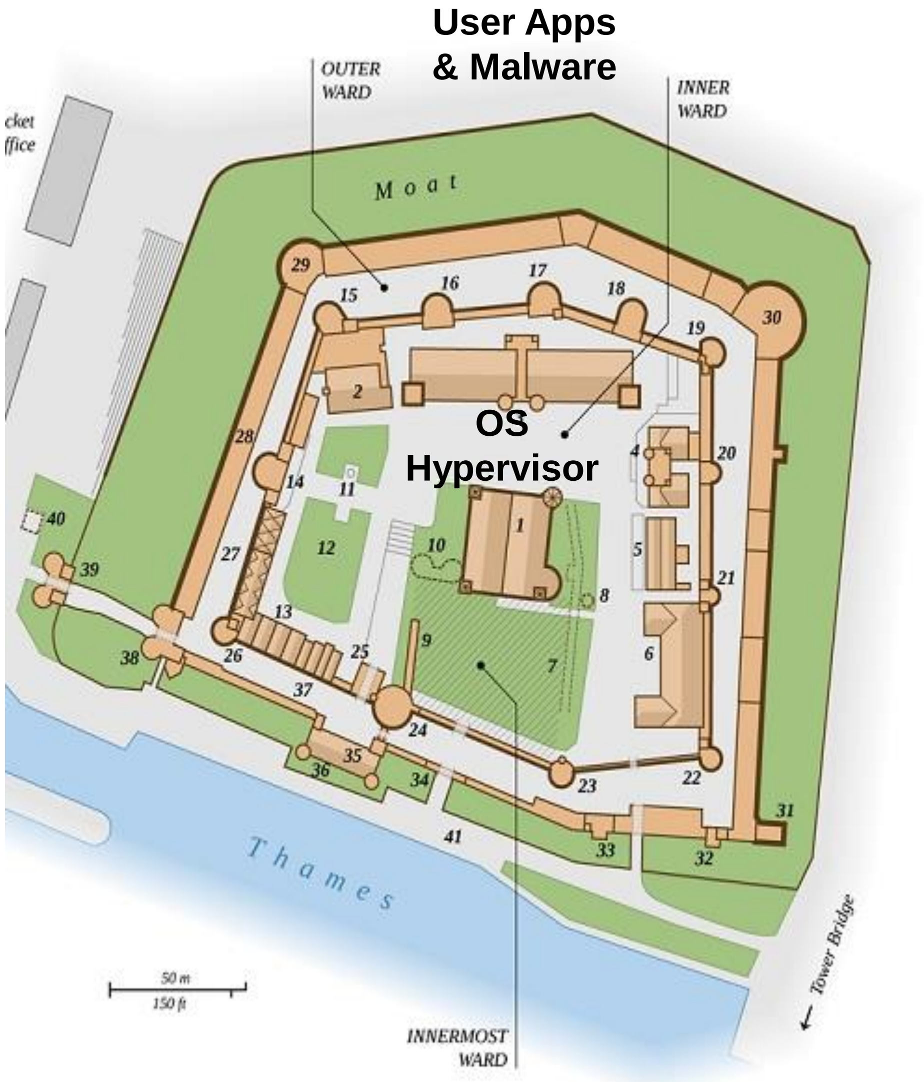
Trusted Execution Environment

- Hardware root of trust
  - A basis for system integrity
- Integrity through Trusted boot
- Secure peripheral access
  - Screen, keypad, fingerprint sensor, etc.
- Secure application execution
- Trust established outwards
  - With normal world apps
  - With internet/cloud apps



Castle Analogy

- Layers of defense
- Reducing attack surface
- Increasing isolation
- Principle of least privilege
- Most precious assets protected by multiple layers of security

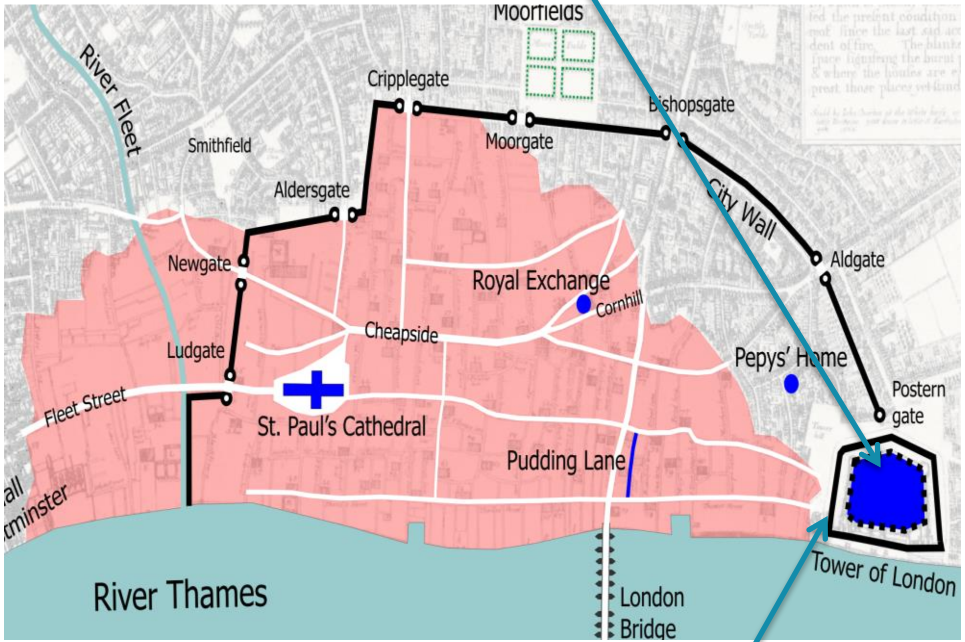


Castle Analogy

But…

- Modern OS/Framework is ~10GB + GBs of apps
- So maybe we should think of a walled city and castle
- Attacks happen
- Everyone knows what the assets are and which room they are in
- Where to put high-value assets such as keys?

Crown Jewels

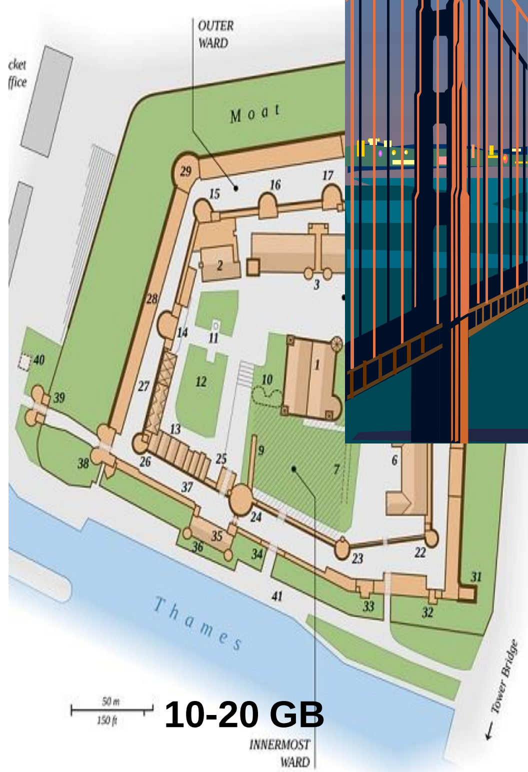


Thief entered here & stole keys!

Implementation details matter!

Castle Analogy with TrustZone Based TEE

- TrustZone technology-based TEE creates a second (much smaller security boundary) castle with only one door, carefully designed entry/exit & APIs
- Keys only used in secure world, protected crypto, encrypted storage, secure execution, secure peripherals
- Offers:
  Integrity (part of trusted boot)
  Confidentiality
- TrustZone TEE castle is invisible to normal world



ARM®

Castle Analogy with TrustZone Based TEE

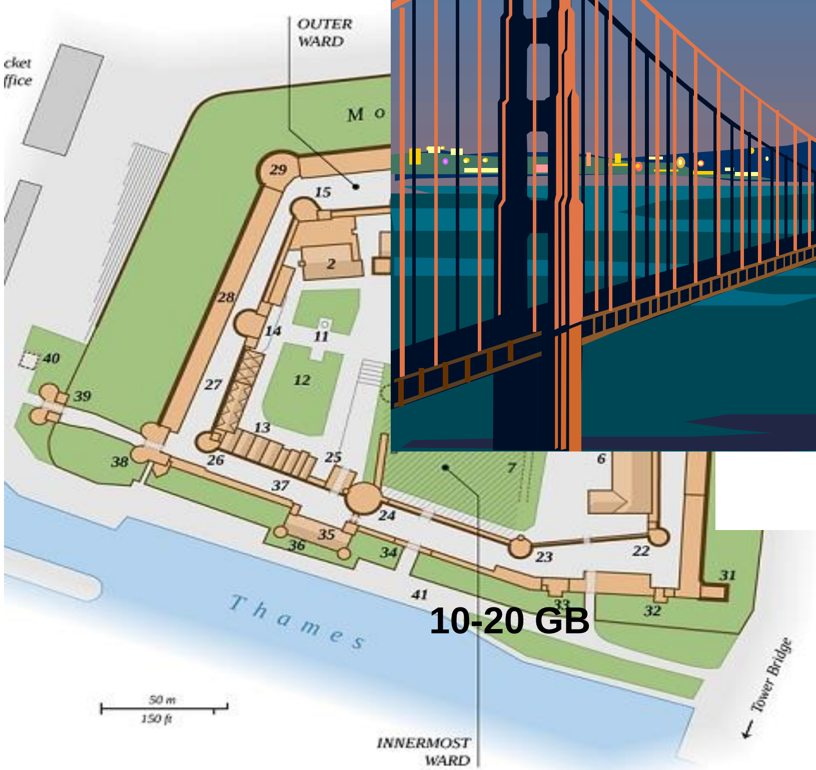
Secure World

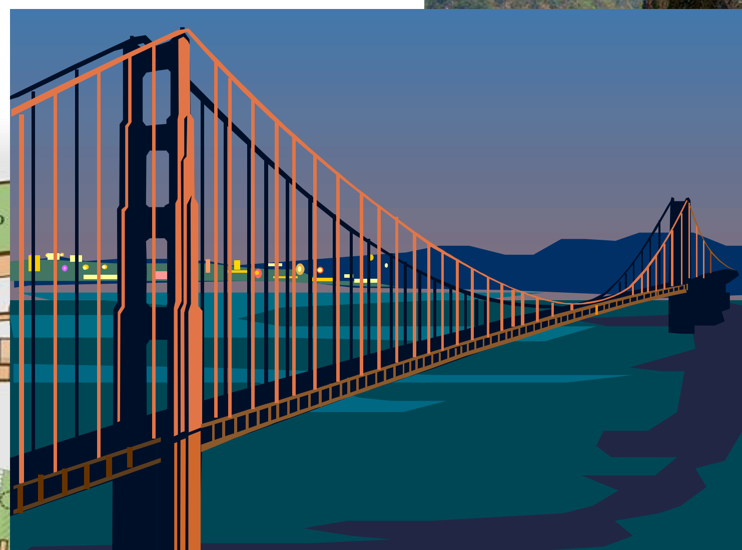
Isolated Trusted Apps

Trusted OS
e.g. Trustonic t-base300

1-2MB

Normal World





10-20 GB

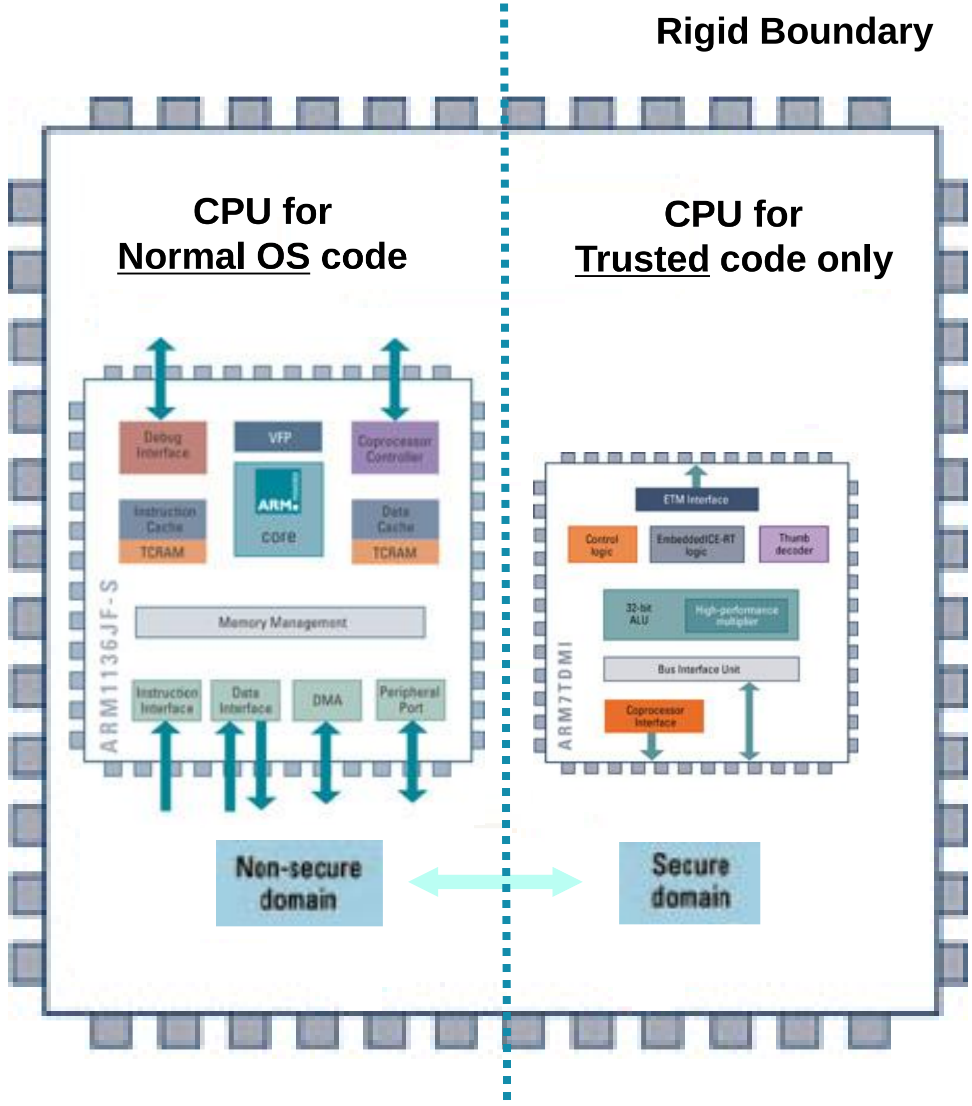
GlobalPlatform Client API
SMC calls at EL3
e.g. ARM Trusted Firmware

ARM®

TrustZone: Two CPUs Virtualized in One

In pre-TrustZone systems:

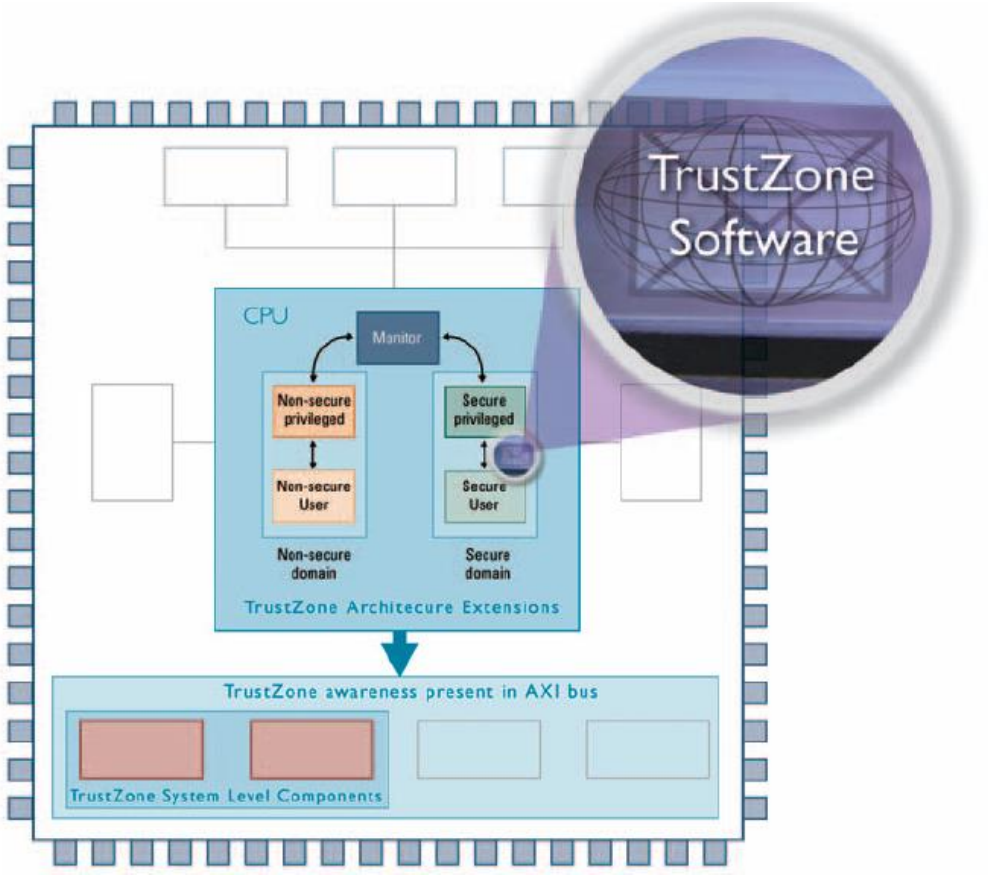
- Rigid allocation of MHz/ resources independent of the application
- Silicon costs with redundant hardware that is idle most of the time
- Complex control logic and deficient performance and power consumption



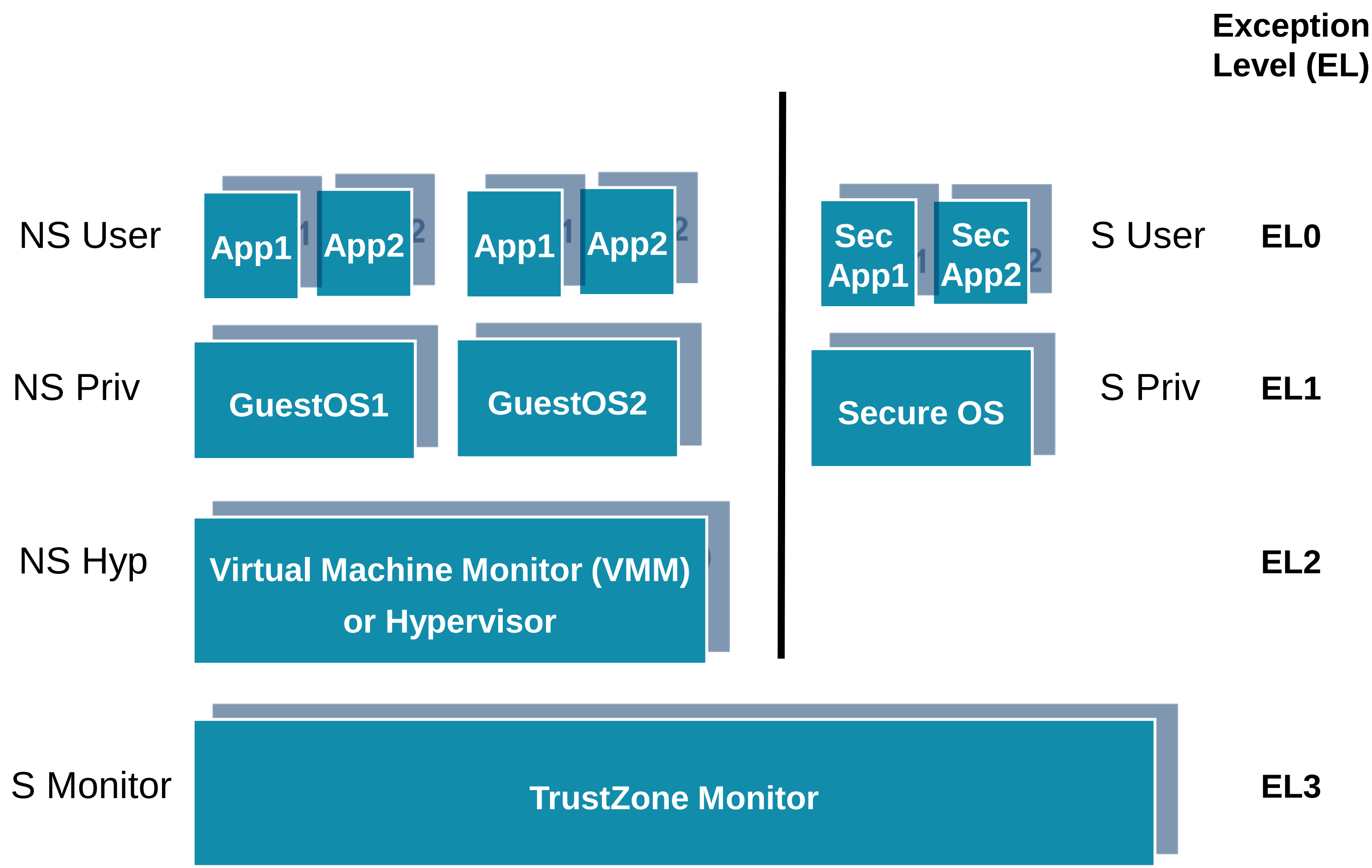
TrustZone Basics

Key advantages over separate secure processor solutions:

- CPU MHz/resources are dynamically shared
- Two domains in same machine
  - Difficult to give precise “overhead” values since secure and non-secure tightly integrated from design standpoint
- Use exceptions to move between modes



AArch64 Exception Levels



AArch64: Exception Model

- 4 exception levels: EL3-EL0
  - Forms a privilege hierarchy, EL0 the least privileged (user mode)

- Exception link register written on exception entry
  - Interrupt masks set on exception entry
  - 32-bit to 64-bit exception zero-extends the link address

- Exceptions can occur to the same or a higher exception level
  - Different vector base address registers for EL1, EL2, and EL3

- Vectors distinguish:
  - Exception type: synchronous, IRQ, FIQ or system error
  - Exception origin (same or lower exception level) and register width

http://www.arm.com/files/downloads/ARMv8_Architecture.pdf
http://www.arm.com/products/processors/armv8-architecture.php

ARM®

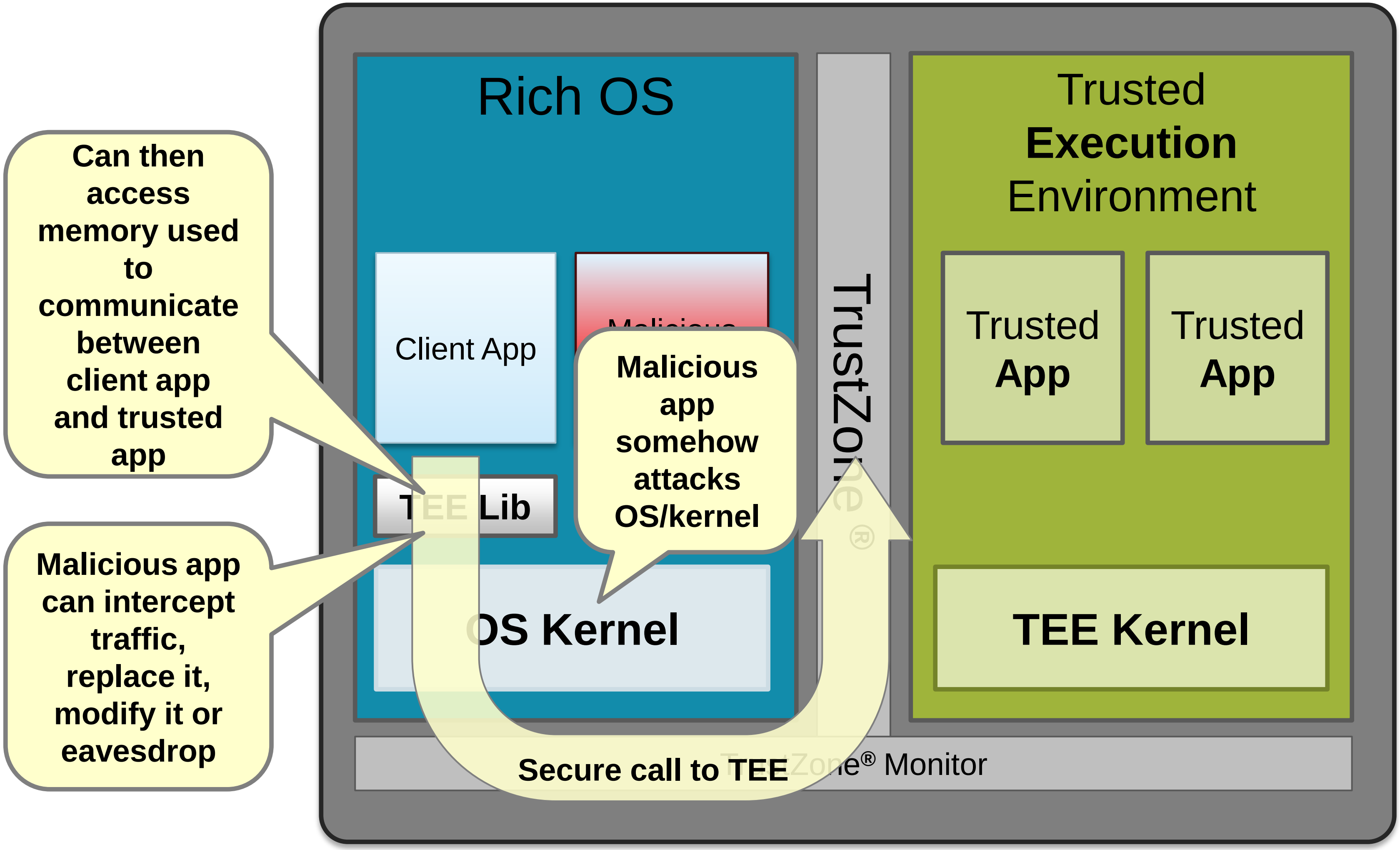
AArch64 Registers

X0	X8	X16	X24
X1	X9	X17	X25
X2	X10	X18	X26
X3	X11	X19	X27
X4	X12	X20	X28
X5	X13	X21	X29
X6	X14	X22	X30*
X7	X15	X23	

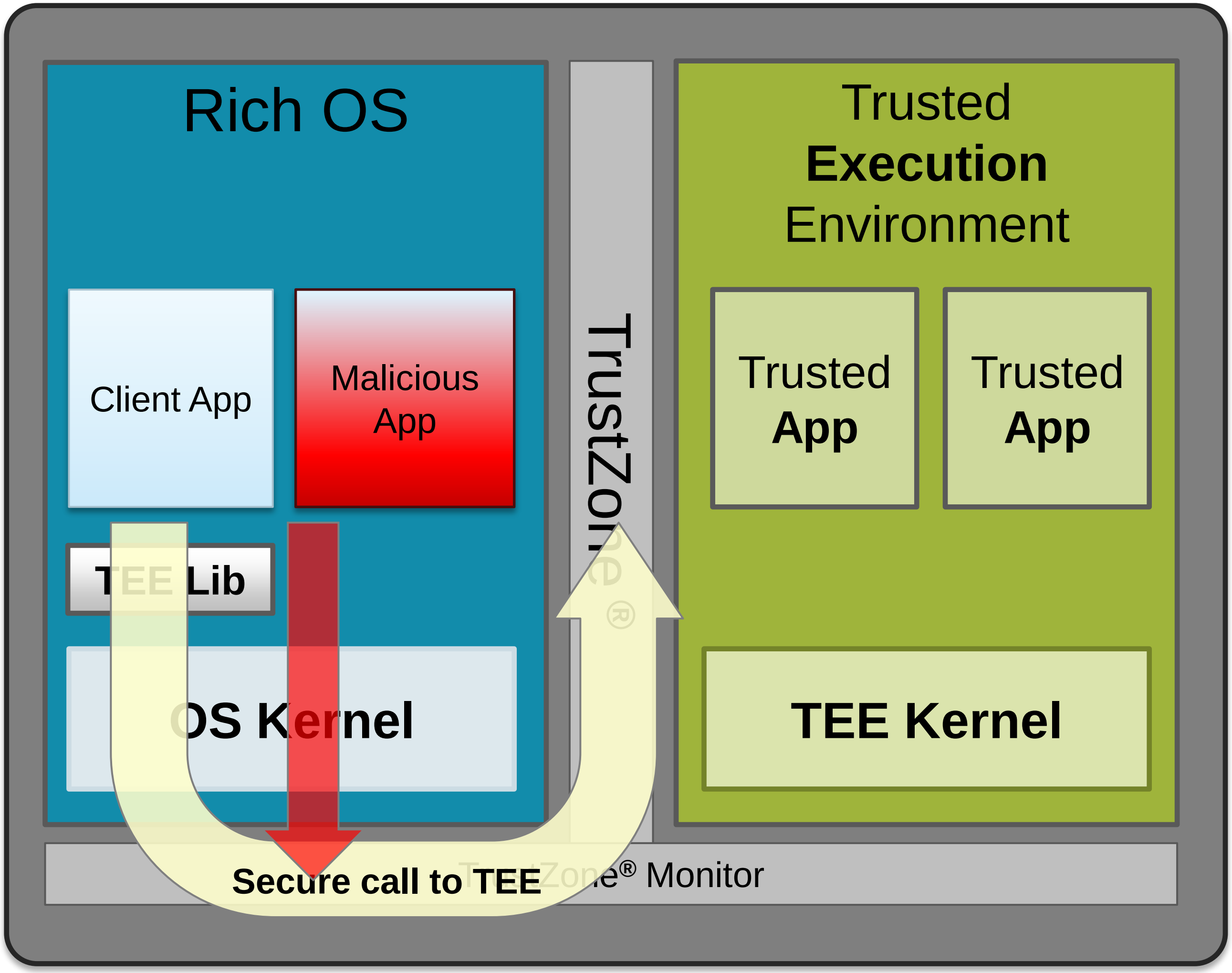
* _procedure_ LR

	EL0	EL1	EL2	EL3	
SP = Stack Ptr	SP_EL0	SP_EL1	SP_EL2	SP_EL3	(PC)
ELR = Exception Link Register		ELR_EL1	ELR_EL2	ELR_EL3	
Saved/Current Process Status Register		SPSR_EL1	SPSR_EL2	SPSR_EL3	(CPSR)

Attack Approach: Man In The Middle

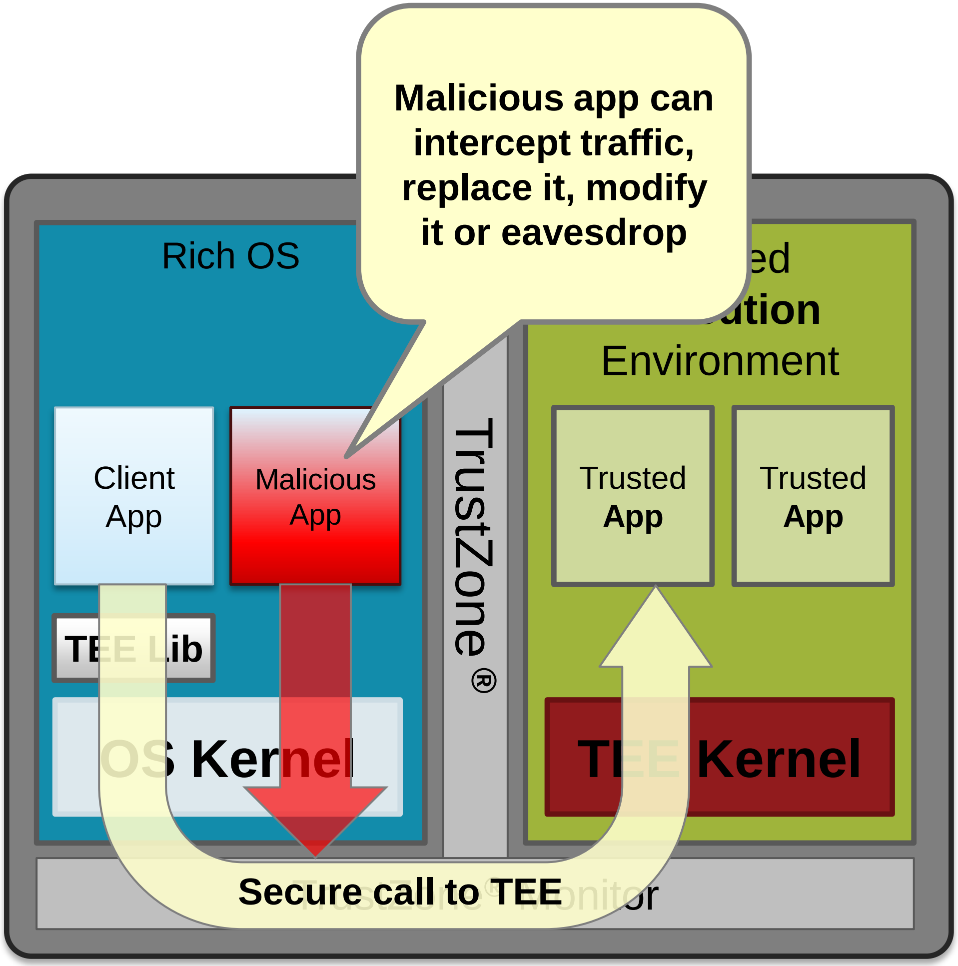


Side-Channel Attacks

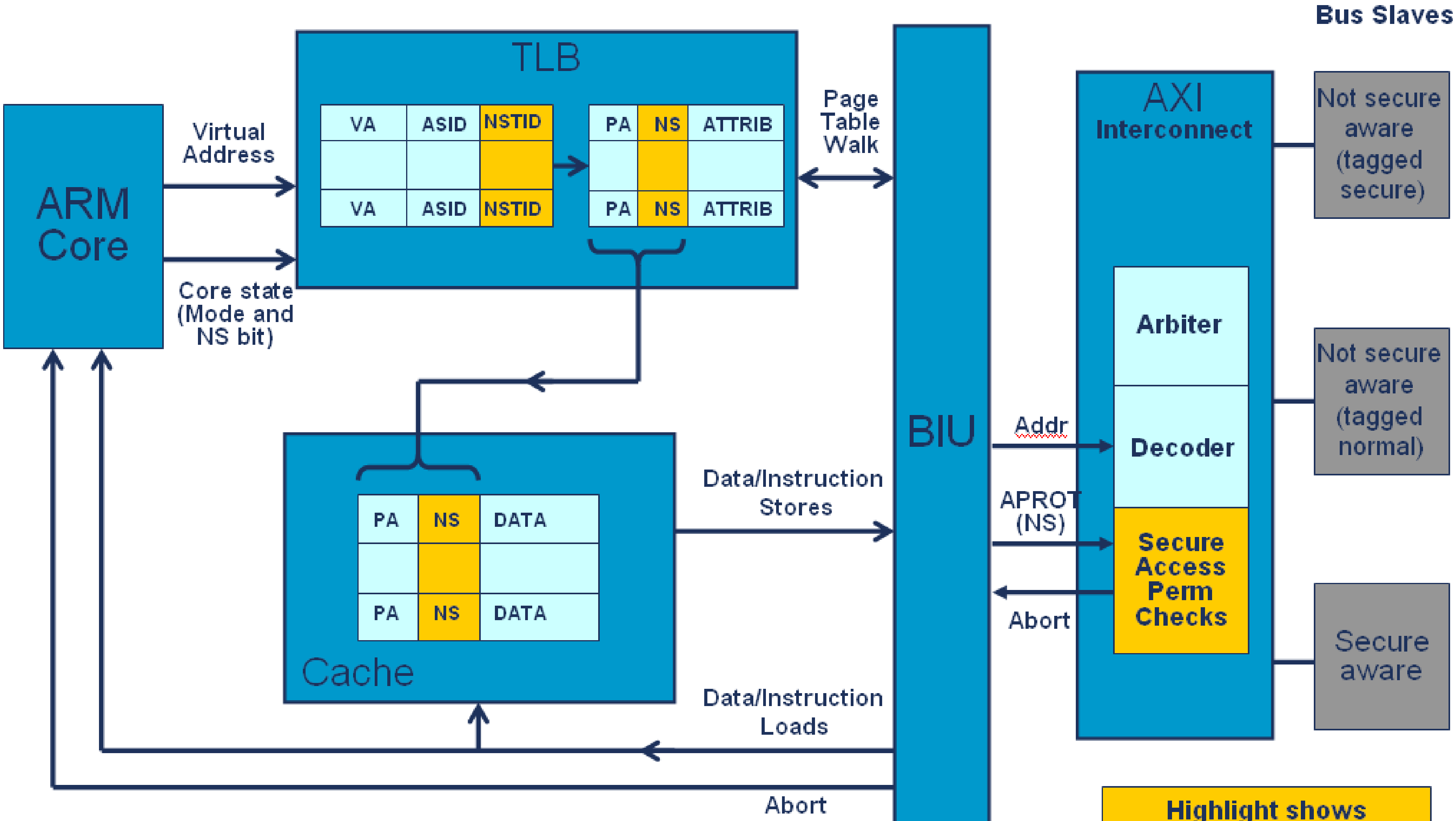


Defenses

- Normal World to Secure World communications are always exposed and vulnerable
- Mitigation
  - Don’t design systems that rely on secure communications between Normal World and Secure World
  - Always use trustworthy components – crypto library, TEE and protocols



Propagating System Security



NS : NOT Secure, treated like an address line

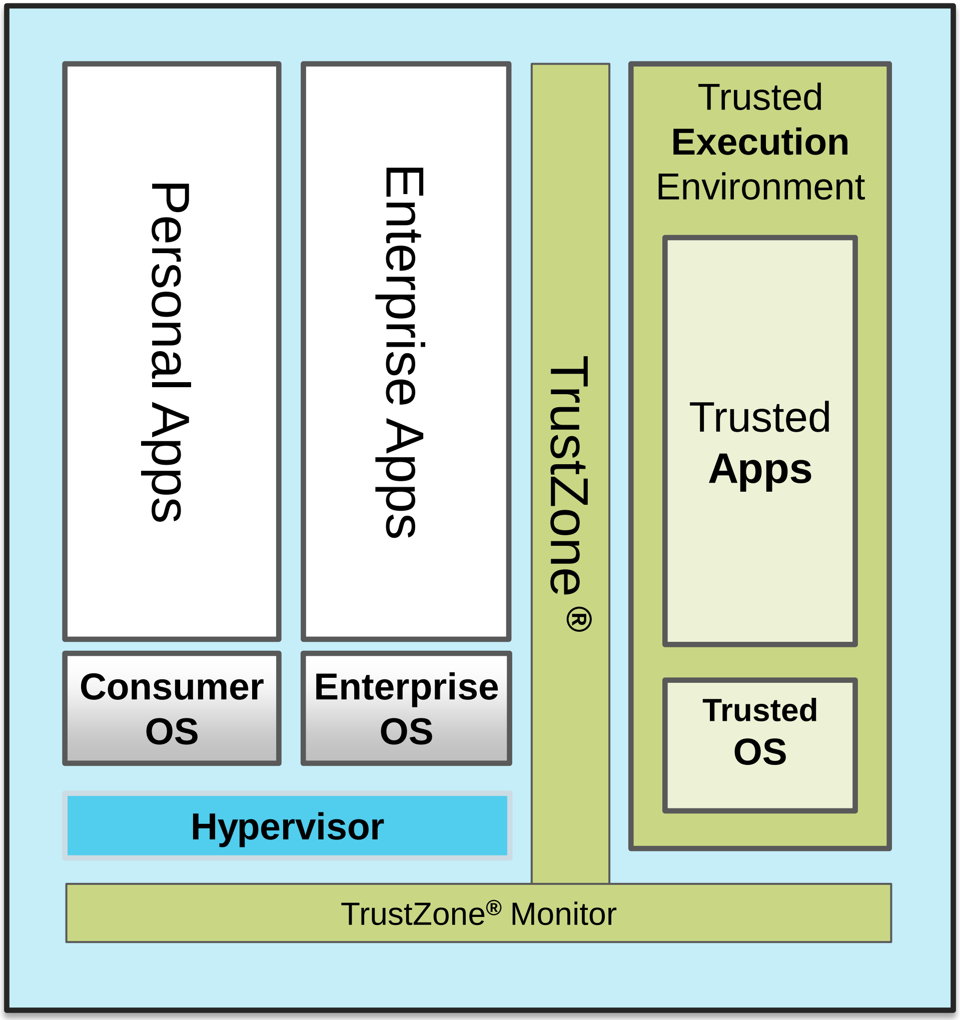
TrustZone Controllers – Vital Statistics

Code	Product	Main Function	Key Features	Size
TZC-380	TrustZone Address Space Controller	Partition external DRAM into secure and non-secure regions	Configurable up to 16 regions of size 32K to 4G, each with 8 sub-regions (down to 4K). Configurable registering to meet timing constraints with minimum latency. AXI interface for compatibility with NIC-301 and DMC-34x.	10-100k gates
BP141	TrustZone Internal Memory Wrapper	Protects internal SRAM	Manages a single secure region within the SRAM. AXI interface.	<1k gates
BP147	TrustZone Protection Controller	Prevents non-secure accesses to peripherals	Allows peripherals to be safely shared by the Secure and Non-Secure worlds. APB interface.	<1k gates

Application of Hypervisor for BYOD

Two personas

- Mutual distrust model between OSs
- Ensuring enterprise OS Security, while protecting consumer OS privacy
- Enabling enterprises to have control of their own assets in case of loss

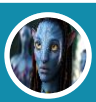


Secure Content Path: SoC Requirements

Premium Content

DRM

Video Codec



Display

Firmware protected against tampering

- Any software component directly used in setting up protected memory path
- Decoders, mixers, renderers, DRM
- Critical components placed in secure processing space
- Integrity checked at boot time

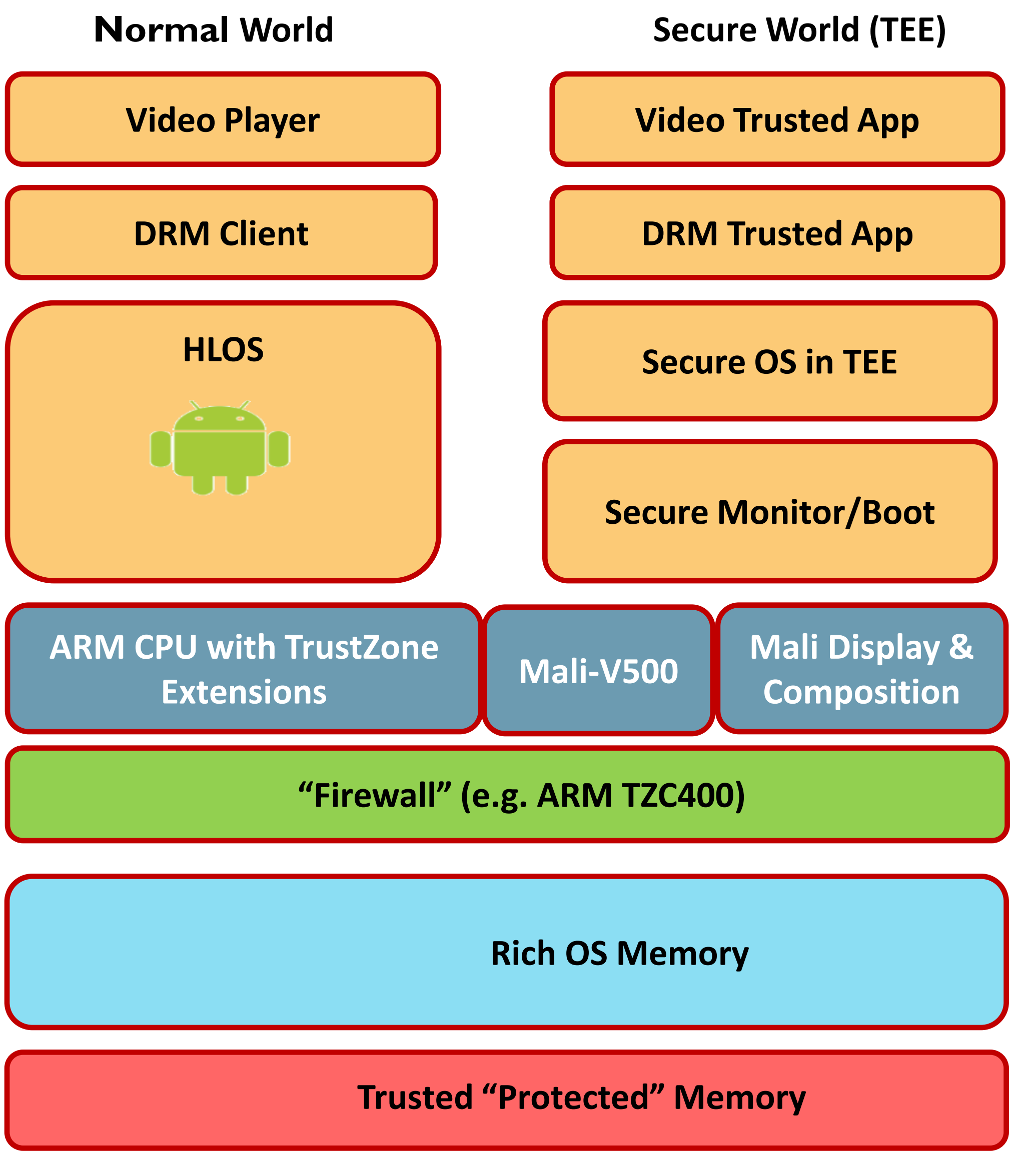
Unencrypted content protected

- After DRM protection removed
- Unencrypted content never accessible to processes running in HLOS
- Unencrypted content only ever written to protected memory

Memory buffers protected by hardware control

- All memory used in processing, decoding, mixing and rendering
- Sufficient memory for video bitstream and frame buffer
- Not accessible by HLOS or unauthorized HW or SW
- Output only to internal display or via protected export clients such as HDCP and DTCP

Secure Implementation Example



Low cost and complexity

- Secure CPU, bus fabric and Video from a single source
- System IP designed to work together
- Simple SW integration – create a secure session then manage scheduling/control as normal

Minimal memory fragmentation

- Major issue for HD content
- Video MMU can be used for secure sessions by TEE
- No need to assign large, contiguous secure buffers

Increased flexibility and protection

- Simultaneous protected and un-protected video streams
- Additional protection of video firmware (read-only) and data (non-executable)

Developing Security – Hierarchy of Trust

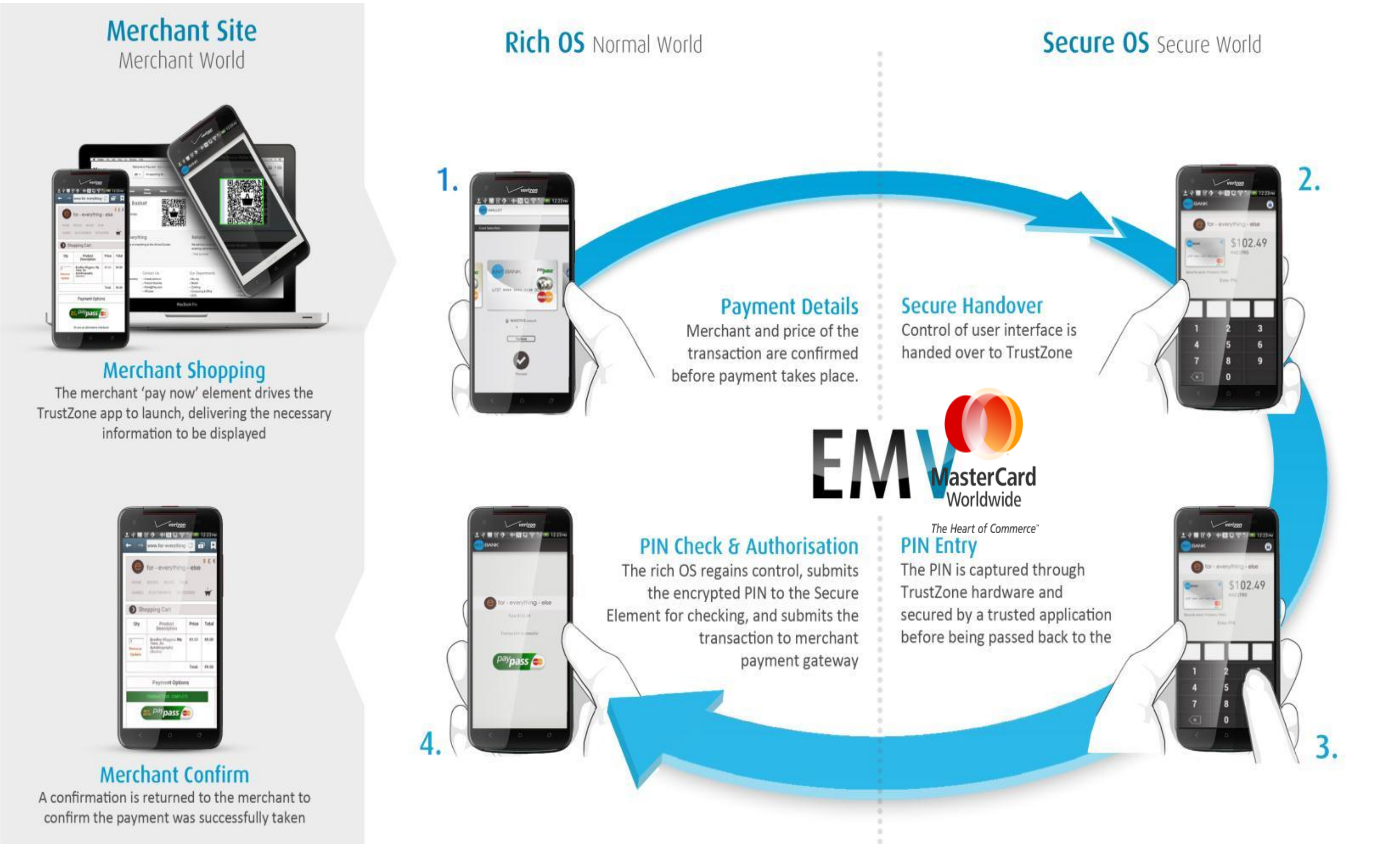
Rich Domain	User Applications	Android or other OS	
Protected Domain	Protected Video Path BYOD System Mgmt	Virtual machines and bus masters isolated by a hypervisor	Hypervisor HYP
Trusted Domain	Secure Firmware Device Management Key Management	Trusted applications executing from a Trusted Execution Environment	TrustZone® TEE
Secure Domain	Secure Element Smartcard SIM & TPM	Tamper proof, physically isolated, EAL certified	SecurCore™ SEE

Use Cases for Hierarchy of Trust Domains

	Client Asset Protection	Content Protection	Mobile Payment		Enterprise	Government	Automotive	Server	Certification
Secure	UICC for smartphones	Secure Storage (CA)	Credit Card Payment Wallet NFC	mPOS (TEE + SE)	Strong Authentication of user credentials	Strong Authentication of user credentials (eID)			GlobalPlatform EMVCO
Trusted	MDM SIM Lock Trusted Storage Trusted Boot Trusted Apps	Trusted Storage (Keys) DRM, CA Trusted video playback path	Cloud Payment Strong Authentication - Trusted UI Wallet		MDM BYOD Trusted UI Integrity Trusted VPN Trusted FOTA	MDM Trusted UI Key Store Trusted Services (eTS)	Infotainment Integrity	Trusted boot Integrity	GlobalPlatform
Protected	Apps isolation	SW DRM	Isolation		BYOD	Dual Persona	Isolate Infotainment and CAN	VMs KVM, XEN…	
Rich	SW Crypto	SW DRM	Web Remote Payment (SSL)		SW BYOD SW FOTA				FIPS 140.2

ARM®

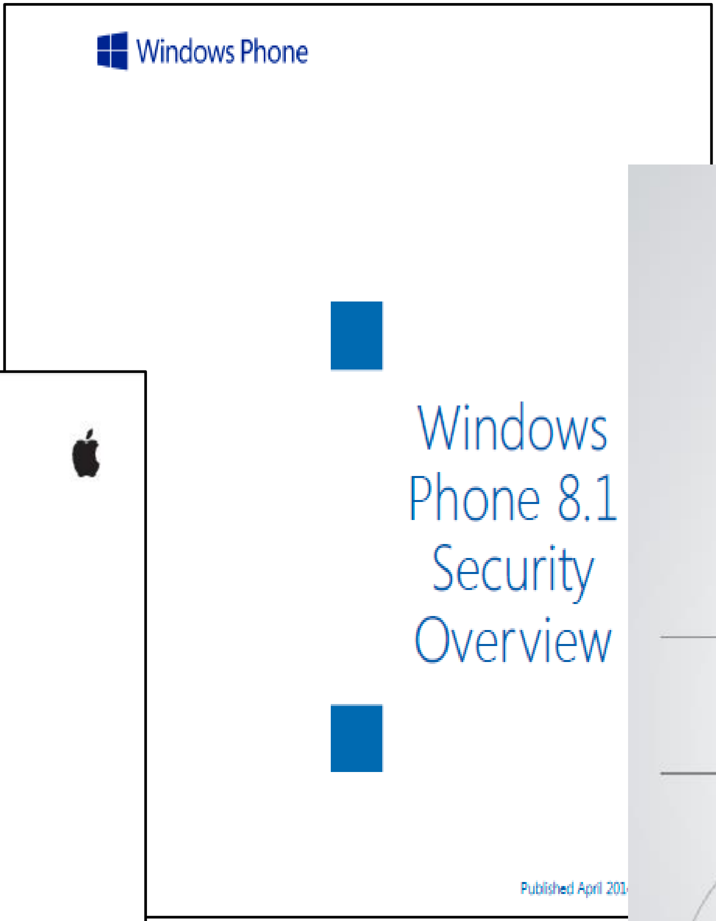
Web Payment Example: MasterPass

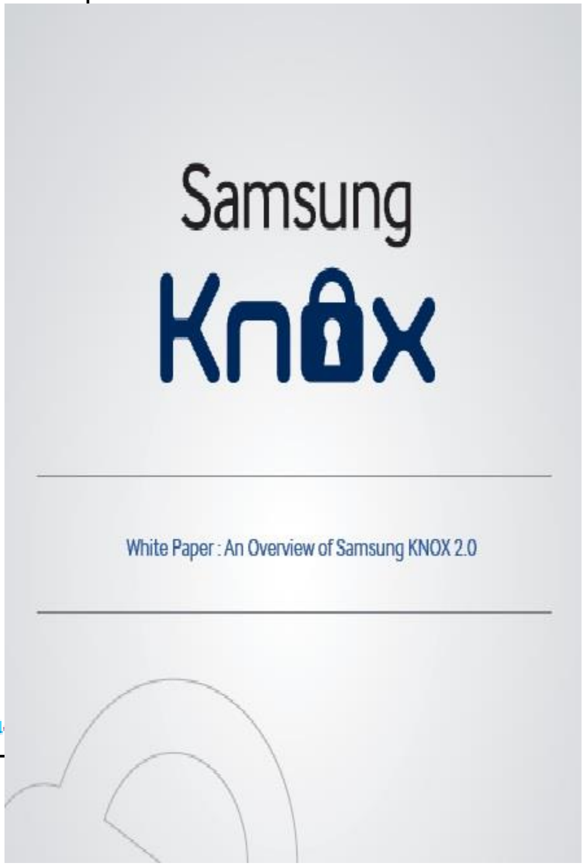


ARM®

Current Practice

Three recent whitepapers from Apple, Samsung and Microsoft give good insight into modern mobile security practice





http://www.apple.com/ipad/business/docs/iOS_Security_Feb14.pdf
http://www.microsoft.com/en-us/download/details.aspx?id=36173
http://www.samsung.com/global/business/mobile/resource/white-paper

CESG General Security Recommendations

- Assured Data at Rest
- Assured Data in Transit
- Authentication
- Secure Boot
- Platform Integrity
- Application white listing
- Malicious Code Detection
- Security Policy Enforcement
- External Interface Protection
- Device Update Policy
- Event Collection
- Incident Response

https://www.gov.uk/government/publications/end-user-devices-security-guidance-general-security-recommendations

Encrypted Data at Rest and Data in Transit

- Assured data at rest: Data at rest should be suitably encrypted
  - Typical ARM SoC has a crypto hardware engine to encrypt/decrypt files
    - Also crypto extensions in AArch64
  - Hardware Unique Key available only to Trusted OS, fused into silicon can be used to derive other keys
  - Key material can be kept on Secure World side or encrypted "wrapped" and stored as metadata
  - System Integrity as determined by Trusted Boot can be verified before the data is decrypted

- Assured data in transit: IPSec VPN of "assured foundation grade" & configured appropriately
  - TrustZone technology-based TEE can add strong 2-factor authentication for remote working

ARM®

Authentication and Secure Boot

- Authentication:
  - User to Device, User to Service, Device to Service
  - Trusted peripherals are handled only by the Secure World
  - Protocols such as FIDO will simplify the silo nature of the authentication status quo

- Secure Boot:
  - Should not be modifiable by unauthorized entity and attempts should be detected
  - Device boots into Secure World and runs only cryptographically verified boot loaders
  - Device starts Trusted OS before main OS is started
  - Measurements of boot process can be made to test for tampering

Need for Authentication

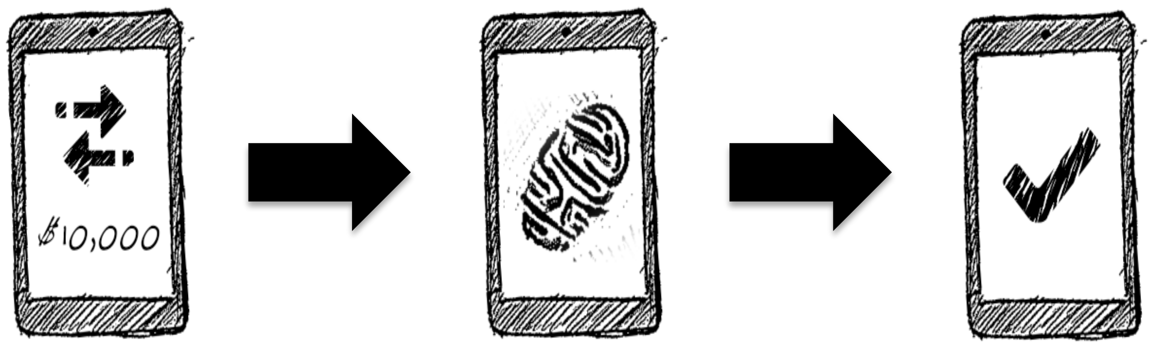
- People use the same simple passwords (Analysis of 6m accounts showed that 10k common passwords would give access to 98.8% of the accounts)
    - 1k passwords give access to 90% of the accounts see https://xato.net/passwords/more-top-worst-passwords
    - 10k passwords give access to 98.8% see https://xato.net/passwords/more-top-worst-passwords/
- People reuse them
    - In 2007: People had 25 accounts and used 6.5 passwords (see Large Scale Study on Web Password habits)
    - 73% of the users shared their online banking password with at least one non-financial site

FIDO Functionality

- Discovery of authenticators on the client
- Registration
- Authentication
- Transaction confirmation

FIDO User Experiences

PASSWORDLESS EXPERIENCE (UAF standards)



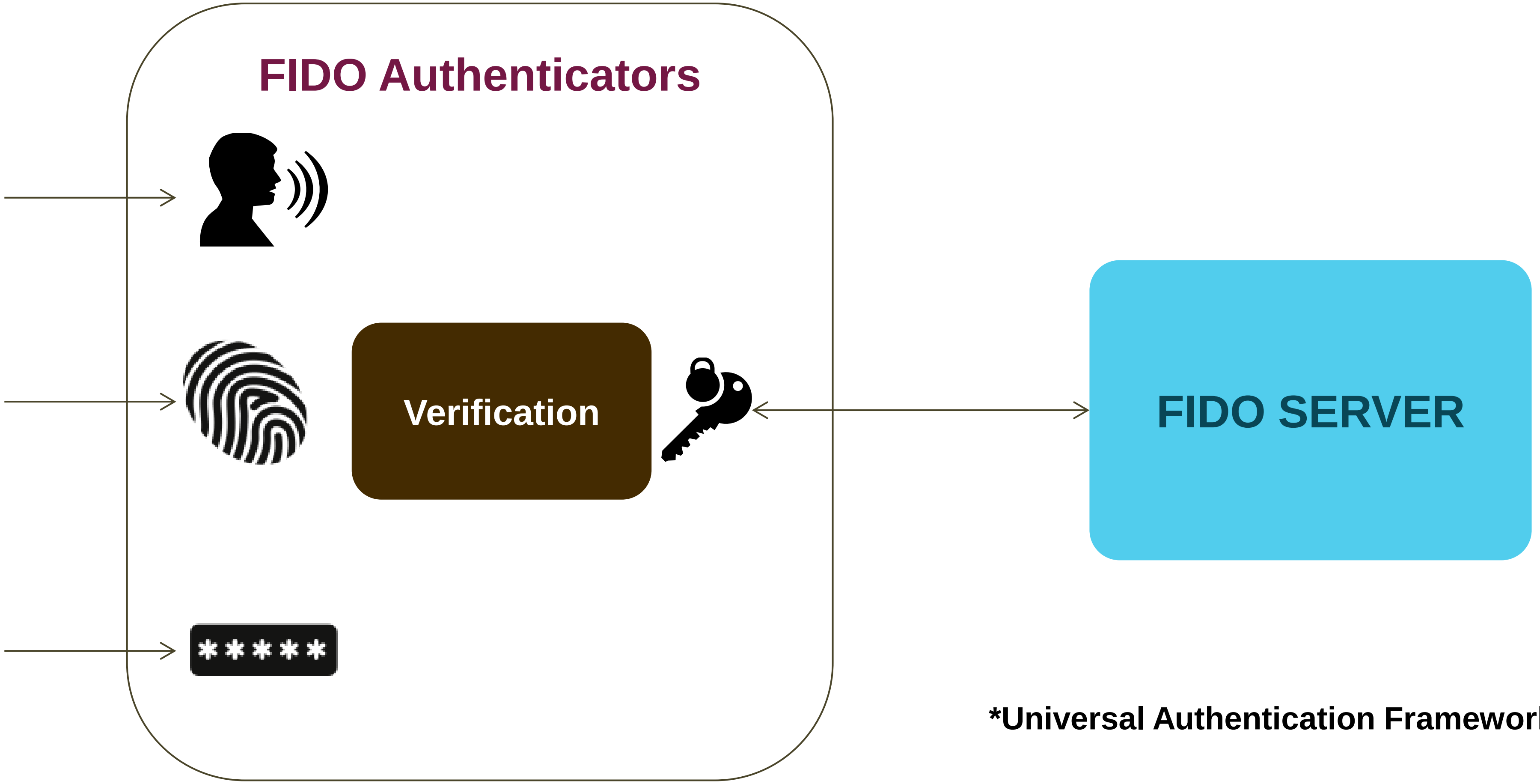
Transaction Detail → **Show a biometric** → **Done**

SECOND FACTOR EXPERIENCE (U2F standards)

Login & Password → **Insert Dongle, Press button** → **Done**

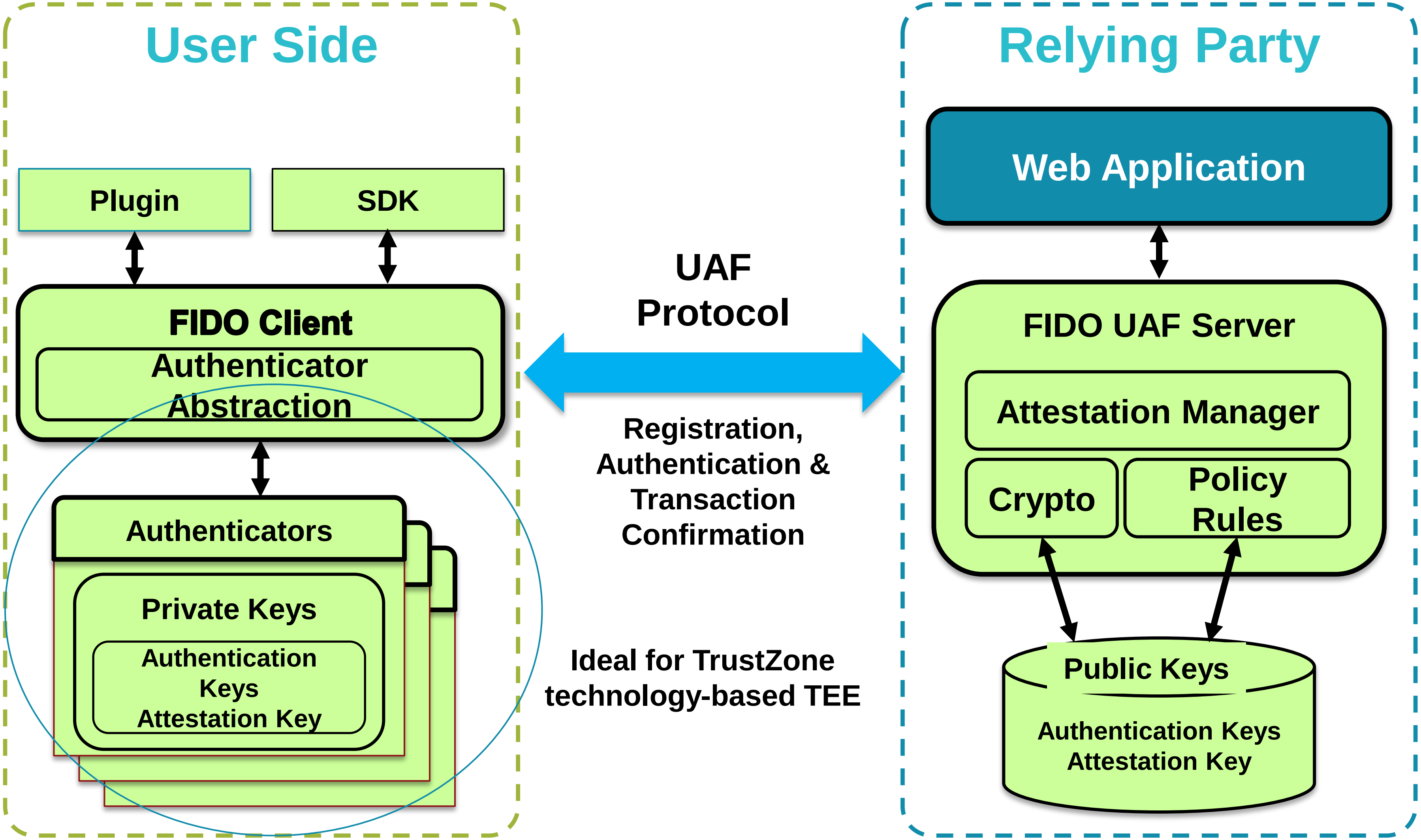
ARM

How Does FIDO UAF* Work?



*Universal Authentication Framework

FIDO - Universal Authentication Framework



Conclusions - What Is Needed?

- Mobile security is built on hardware root of trust
- Requires hardware, software and services to work together
- Not all security is equal – consider ARM's Hierarchy of Trust model
- CESG recommendations and OEM security whitepapers useful for orientation
- FIDO can help us move beyond passwords

Acronyms

ASID = address space identifier

BIU = bus interface unit

BYOD = bring your own device

CA = certificate authority

CESG = Communications-Electronics Security Group (UK)

CVC = card verification code (or card verifiable certificate for smart cards)

DNSSEC = DNS security extensions

DPA = differential power attack

DEMA = differential electromagnetic attack

DTCP = digital transmission content protection

FIDO = Fast IDentity Online

FIQ = fast interrupt request

FOTA = firmware over the air (secure updates)

HDCP = high bandwidth digital content protection

HLOS = high level operating system

HSM = hardware security module

IRQ = interrupt request

mPOS = mobile point of sale

NS = not secure

NSTID = nonsecure table identifier

PA = physical address

SMC = secure monitor calls

SSO = single sign-on

TEE = trusted execution environment

TLB = translation lookaside buffer

TLS = transport layer security

UAF = Universal authentication framework

UICC = universal integrated circuit card

VA = virtual address