

(Payment) Application Protection in Financial Services

James Anderson, EVP, Digital Payment Products



Disclaimer

I work for Mastercard

My views are informed by that fact, but these are my opinions, not necessarily the opinions of Mastercard

Agenda

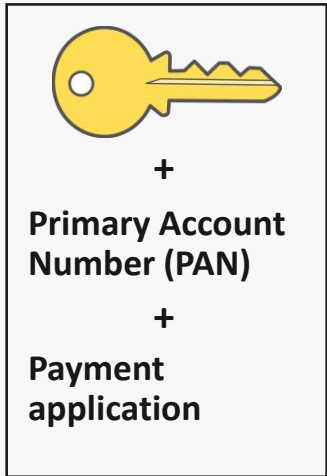
- Why does this all matter?
- A quick primer
- A framework
- A brief history
- Where are we now?
- What would we like?
- How could TEE help?

Why does this all matter?

- Electronic payments is a major source of consumer value and convenience
- Robust security measures are imperative to ensuring the ongoing viability of any specific method for effecting electronic payments
- Static data is intrinsically vulnerable to attack, copying and replay
- The only way to make the data used in electronic payments intrinsically secure is to make it unique for every transaction
- For that we need cryptography...
- ...which means we need to protect cryptographic keys at the very edge of the network, where the transaction takes place

A quick primer on chip card security

We need to protect the key –
to ensure the cryptogram is
valid

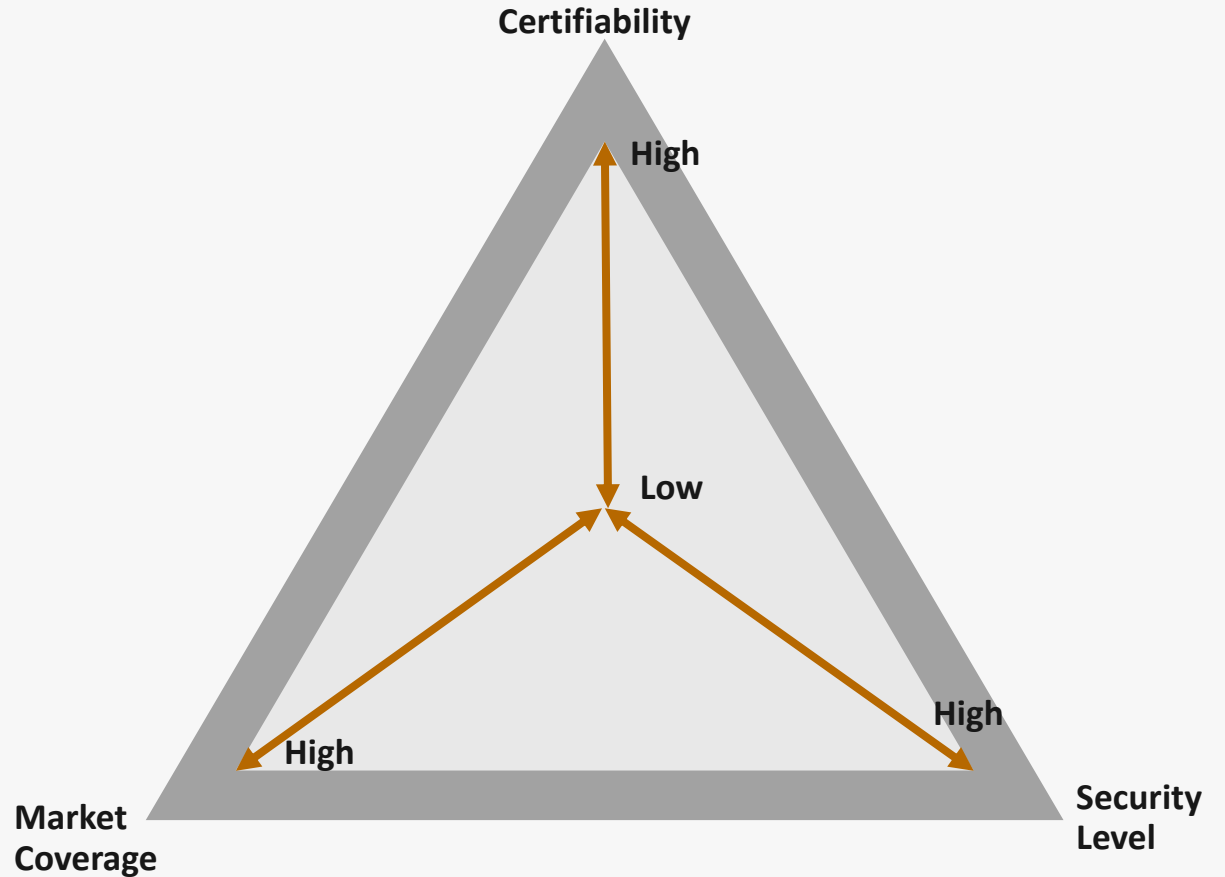


+
Transaction
information

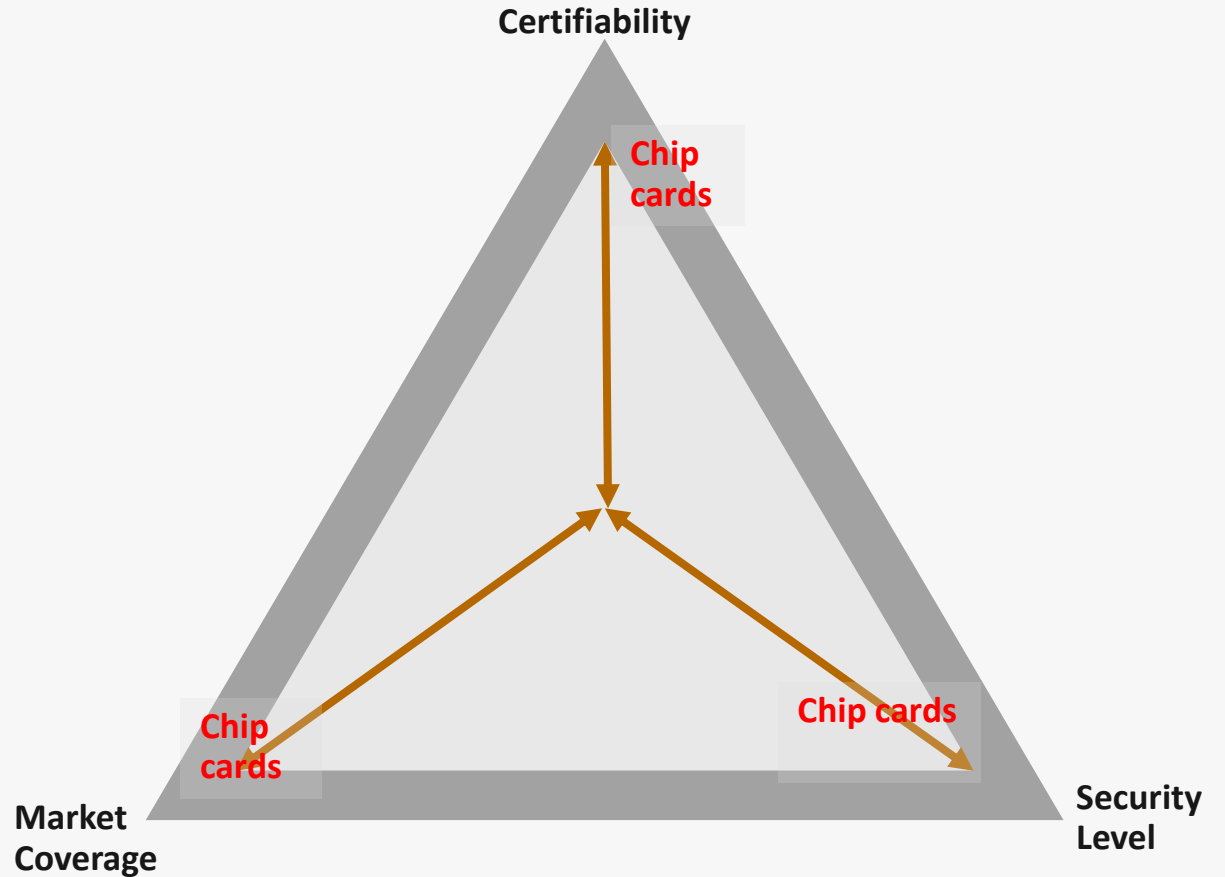


Transaction
cryptogram

A frame to compare security options



A frame to compare security options



Embedded SE



Pros

- A direct emulation of the card model
- Proven security model
- Mature technology
- Easy to certify and establish assurance to our ecosystem

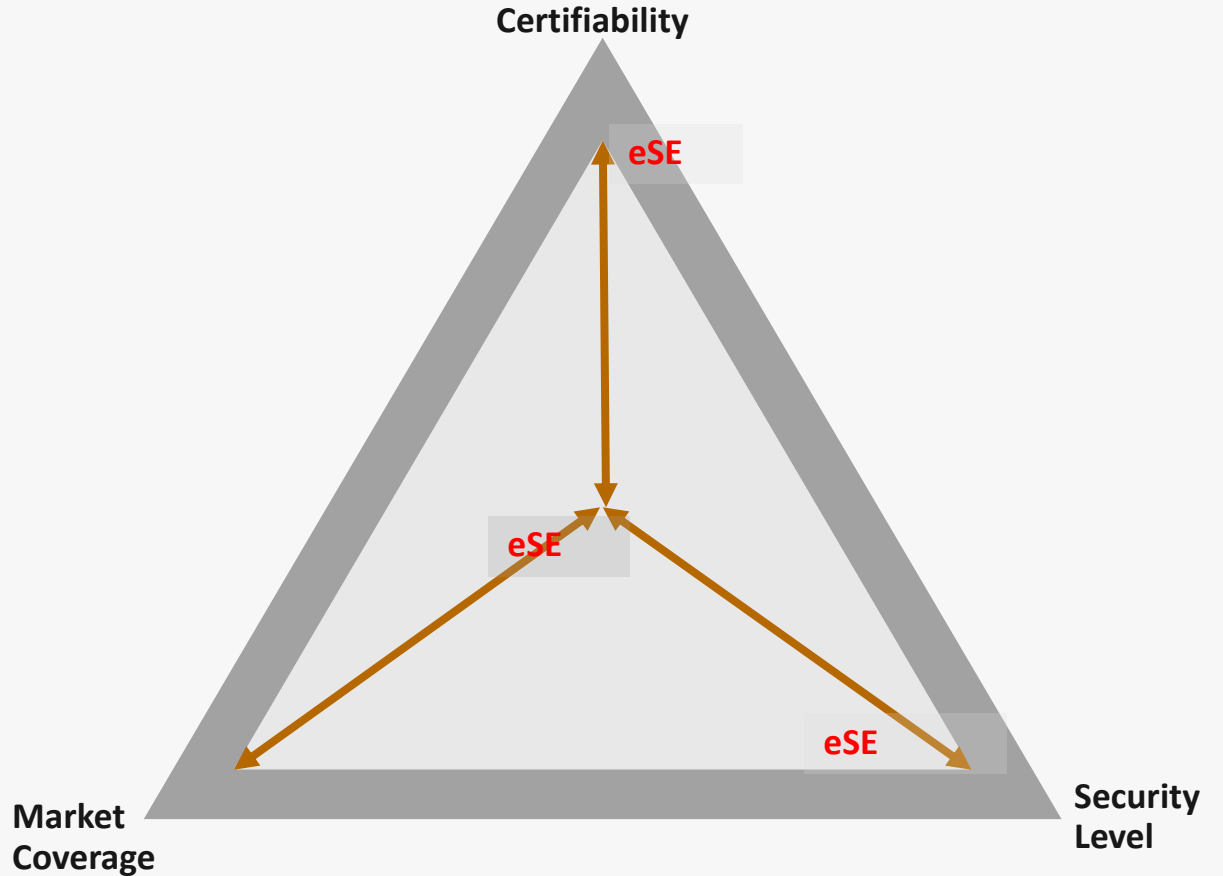
Cons

- Economic problem between device OEMs, and would be payment providers

Embedded SE

Model is only used for Apple Pay today, since Apple controls the hardware, software and payment service

Model is coming back around for passive devices (wearables, IoT) for contactless use case



SIM-based



Pros

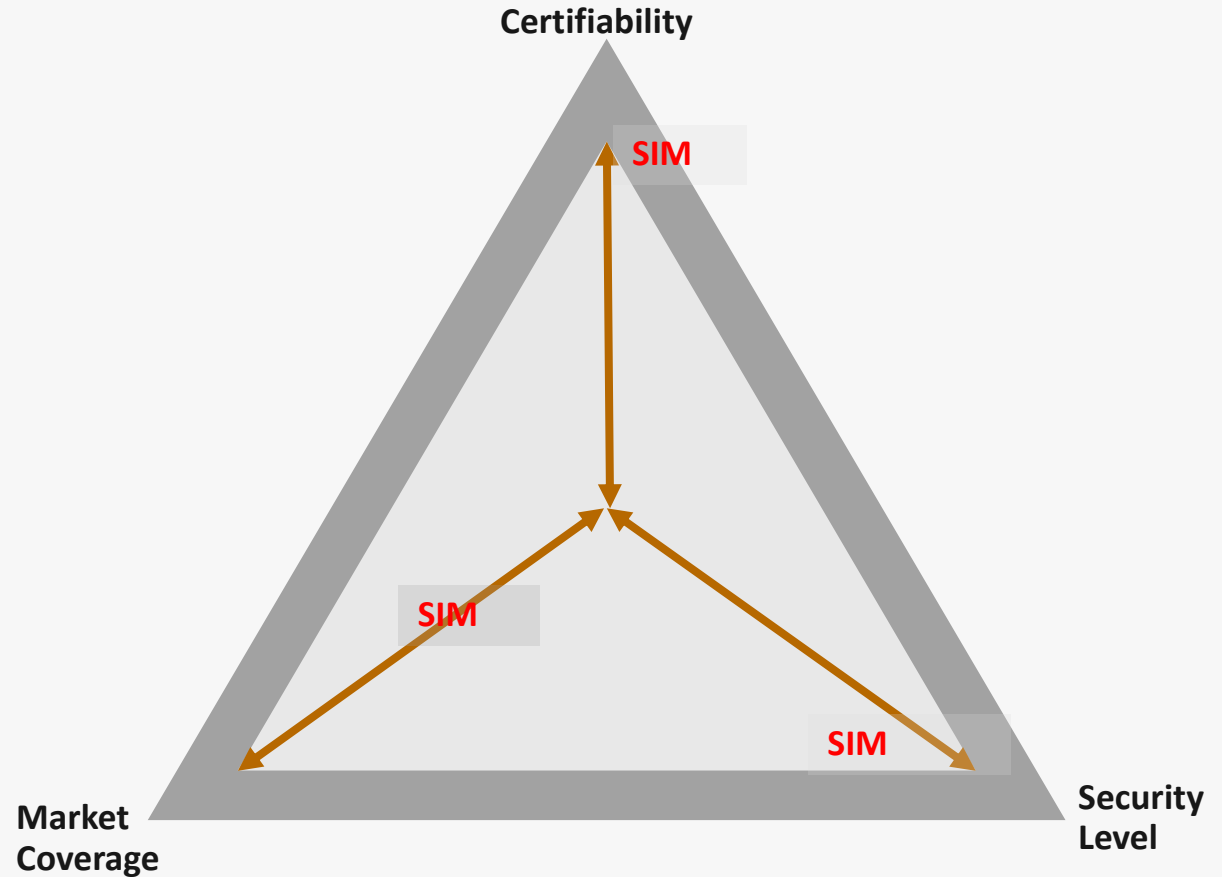
- Same basic technology as chips used in cards
- Certifiable through standard processes

Cons

- Cost to MNOs
- Logistical complexity of deployment
- MNOs challenged as payment service providers

SIM

Model is all but unused today after demise of Softcard



Frustrations of hardware-based approaches led to the search for alternatives

„welcome to the world of HCE

Fundamental realization that the protection of issuer keys could be accomplished different ways

- Hardware protection on the device needed to support offline authorized transactions
 - A declining use case – but occasionally very important (e.g., transit)
- We could shift the hardware to the server-side in an HSM
 - Send single use keys to the device for “redemption” as part of every transaction
 - Device would replenish their key inventory when it connects to the network and could reach the server
- Needed support in the OS
 - To allow NFC traffic originating from a place other than a Secure Element to perform a transaction
 - Named Host Card Emulation (HCE) by Android
 - Because the Host (main processor) is performing NFC Card Emulation transactions
- Great solution for players that did not control hardware
 - Google – Android Pay
 - Wallets deployed by issuers

Software-only



Pros

- Software only - can be used on all devices with right OS
- No toll collector keeper/external control point

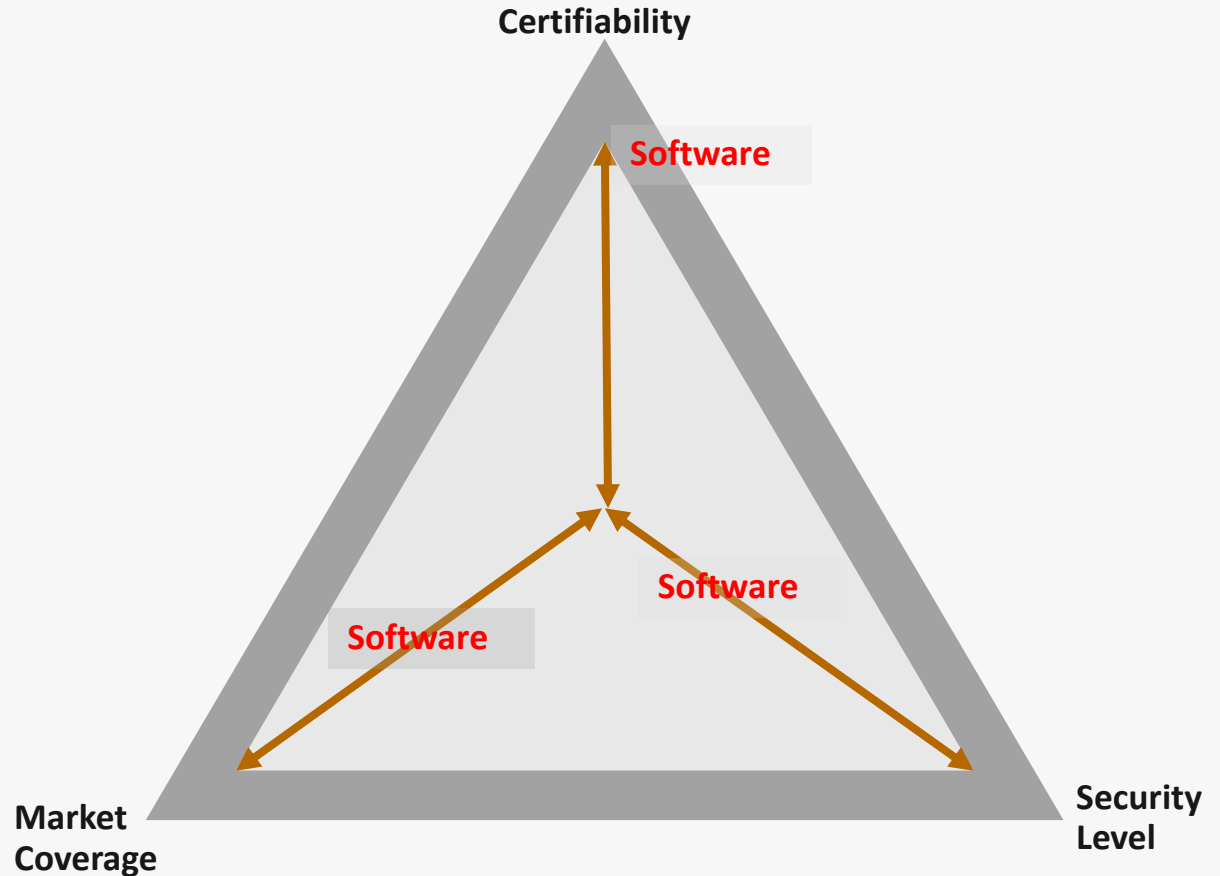
Cons

- Software only – security intrinsically lower than hardware
- Normally needs white box crypto applied to protect the software from attack

Software-only

Widely used today by Android Pay and Issuer Wallets

Demonstrates that accessibility trumps absolute security level – at least in the short term





Pros

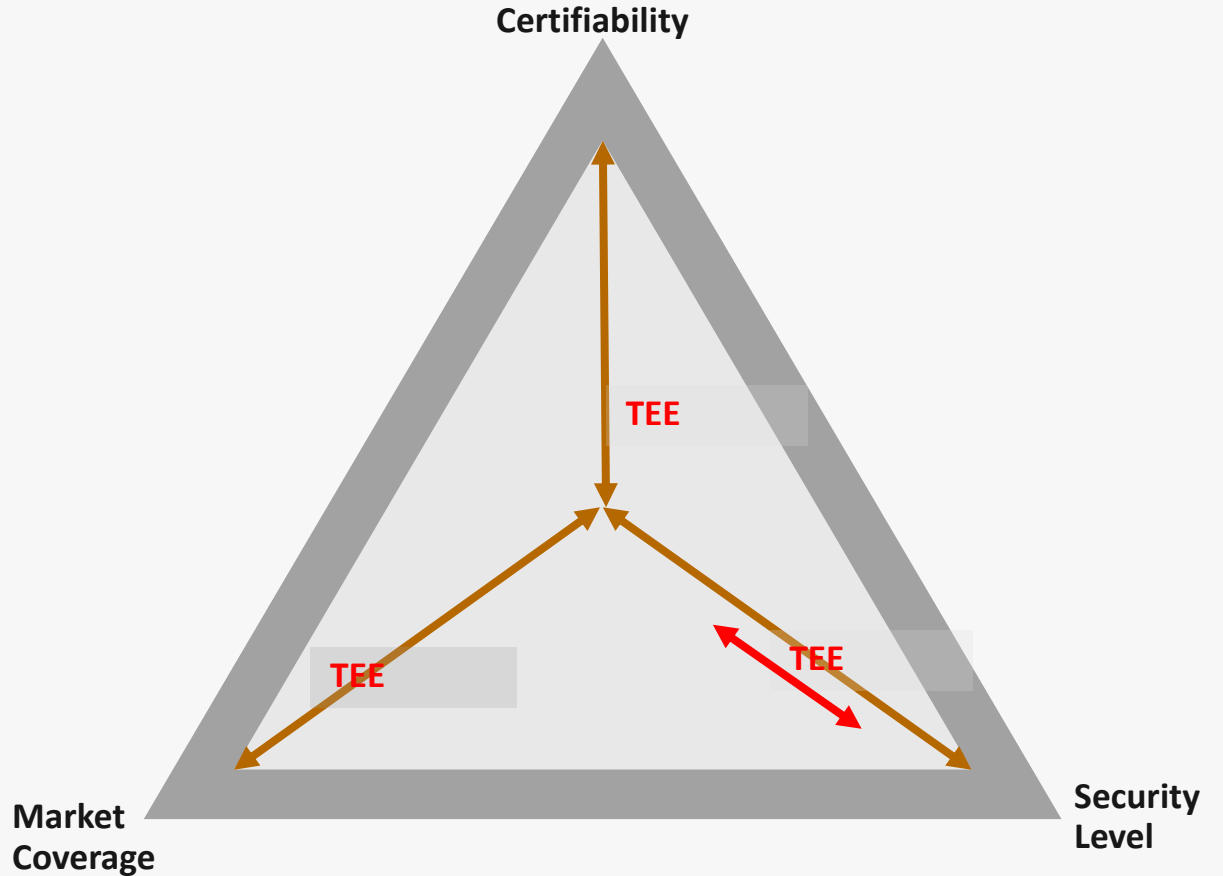
- Hardware-based
- No need for white-box crypto
- Wide access across Android devices

Cons

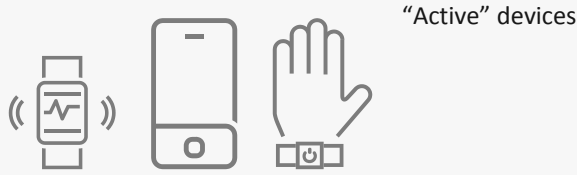
- Actual security level depends on the quality of each deployment
- Hard to certify – OEMs not interested in expense – most don't have payment programs themselves

TEE

Only used by Samsung Pay today -
Samsung willing and able to certify
their own devices for their own
program




Today's Market Landscape




"Active" devices

M/Chip Mobile




Hardware (eSE)

Mastercard TEE Based Payments




Hardware (TEE)

Mastercard Cloud Based Payments (MCBP)



Software

Issuer Wallets



Software + white box crypto

MC payment app specification

Wallet provider

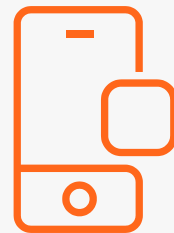
Security model



Tap



In-App



Payment Use Cases

What would I like to see?

Demonstrably high security level capability accessible on as many devices as possible for as many market participants as possible

- Hardware-based solutions will not get us there => fundamental business model problem
- Software solutions working for now (demonstrable security level across all devices) but have some limitations (white box crypto intrinsically proprietary, hard to compare, relies on staying ahead of bad guys in defending remotely distributed software)
- TEE could be the resolution
 - Available across a majority of devices
 - Simple economic deal terms
 - Technically easy to access

So what is holding it back?

- Main hurdle remains certifying an acceptable level of security
 - Security certification is neither simple nor cheap
 - Different implementations of the same TEE approach can have materially different levels of security
 - Who will submit a TEE for certification?
 - Who will pay for the certification?
 - OEMs should, since they implement the TEE
 - But they don't see an RoI on the expenditure
 - Nobody buys a specific Android device because of a more secure payment functionality
 - Can Trustonic?
 - And recoup its investment via charges
 - Will require finding a way to reduce/eliminate variations between how different OEMs implement their TEEs

Other Ways TEE could be useful

As a persistent device identifier

Two potential use cases:

1) For contactless/in-app

- Payment application runs in software
- Place a key in the TEE when the payment application is loaded that uniquely and persistently identifies a device
- Check the key prior to replenishing single-use tokens to device

2) For web payments

- Use as a complement to cookies to persistently identify a device
- When a device is identified the consumer experience is much better

Q&A