

Platform Security Architecture One Year On

Eric Wang

Director of Security Technical Marketing

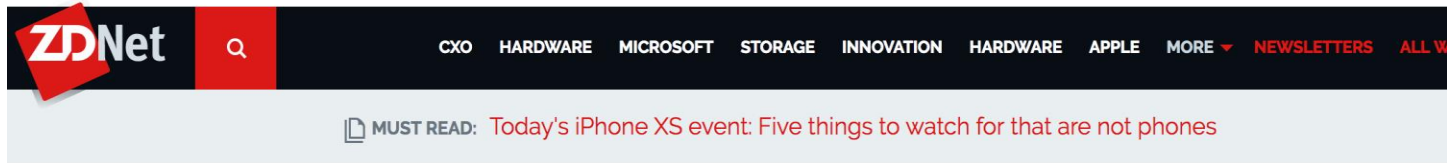
Arm China



Agenda

- PSA – A reminder of what it is and why it is useful
- Achievements over the last 12 months
- What's next?
- PSA Compliance – Introducing the new APIs
- Getting started with PSA

PSA launched at TechCon last year



ZDNet

CXO HARDWARE MICROSOFT STORAGE INNOVATION HARDWARE APPLE MORE NEWSLETTERS ALL W

MUST READ: Today's iPhone XS event: Five things to watch for that are not phones

Arm announces PSA security architecture for IoT devices

Arm hopes the adoption of its new PSA system will help protect trillions of connected devices in the future.



By Charlie Osborne for Zero Day | October 23, 2017 -- 14:00 GMT (15:00 BST) | Topic: Security

Not All Electronic Device Are Secure, But ARM's PSA May Change That

Jim McGregor Contributor
Tirias Research



| | | | | | | |
|-----------|------|-------------------|-------|--------|--------|--------|
| SEARCH IP | NEWS | INDUSTRY ARTICLES | BLOGS | VIDEOS | SLIDES | EVENTS |
|-----------|------|-------------------|-------|--------|--------|--------|

PSA: Next steps toward a common industry framework for secure IoT

tom's HARDWARE PRODUCT REVIEWS BUYING GUIDE!

Prevent glitches.
Really big ones.

Try game-changing automated patch management from SolarWinds® RMM.

Adverti

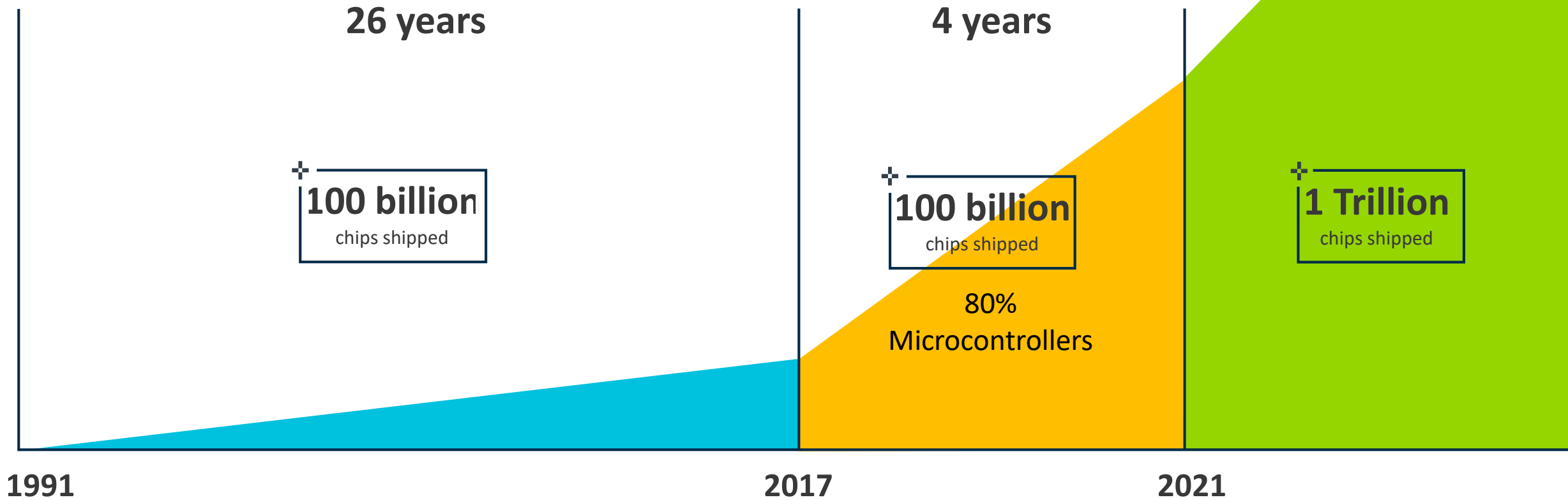
SECURITY > NEWS

Arm Reveals More Details About Its IoT Platform Security Architecture

by Lucian Armasu February 22, 2018 at 11:45 AM - Source: Arm News

Protecting the Next 1 Trillion

Billions on connected IoT devices will need to be secured & managed

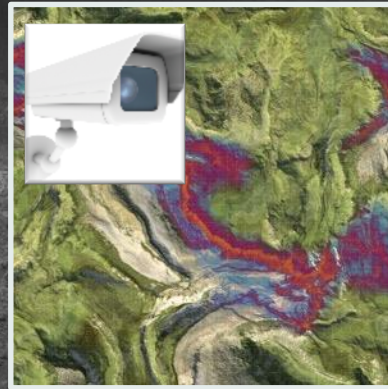


IoT – Still the Wild West?

- Unregulated, no common standards
- Inconsistent approach to security
- Immature and fragmented end markets with diverse requirements
- Trusted data?



Jeep Hack



Mirai Botnet
DDos attack



Owlet Baby
Monitor



Abbott
Pacemaker

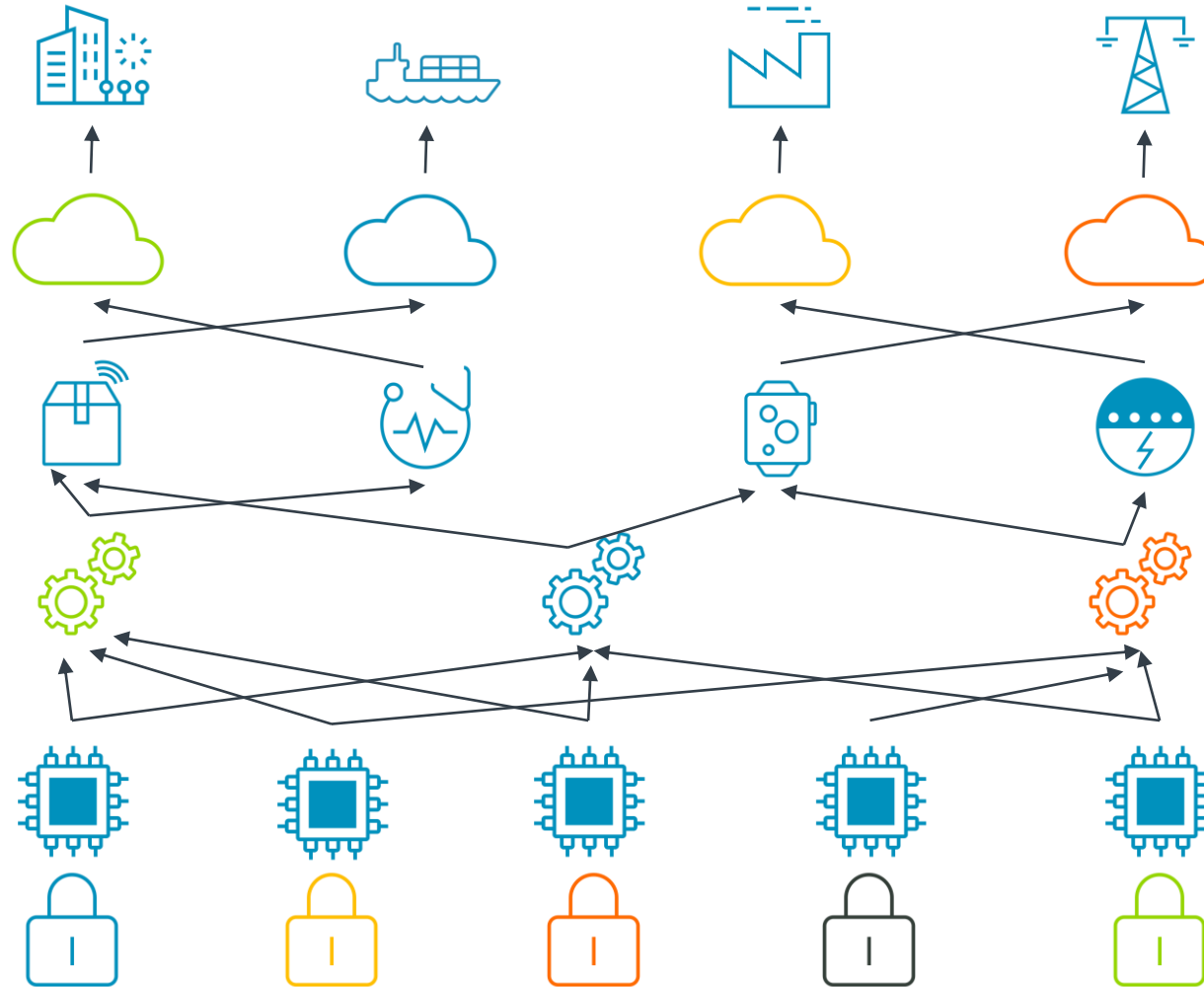
PSA Designed to Address the IoT Challenge

Poor Security

definition across
entire value chain

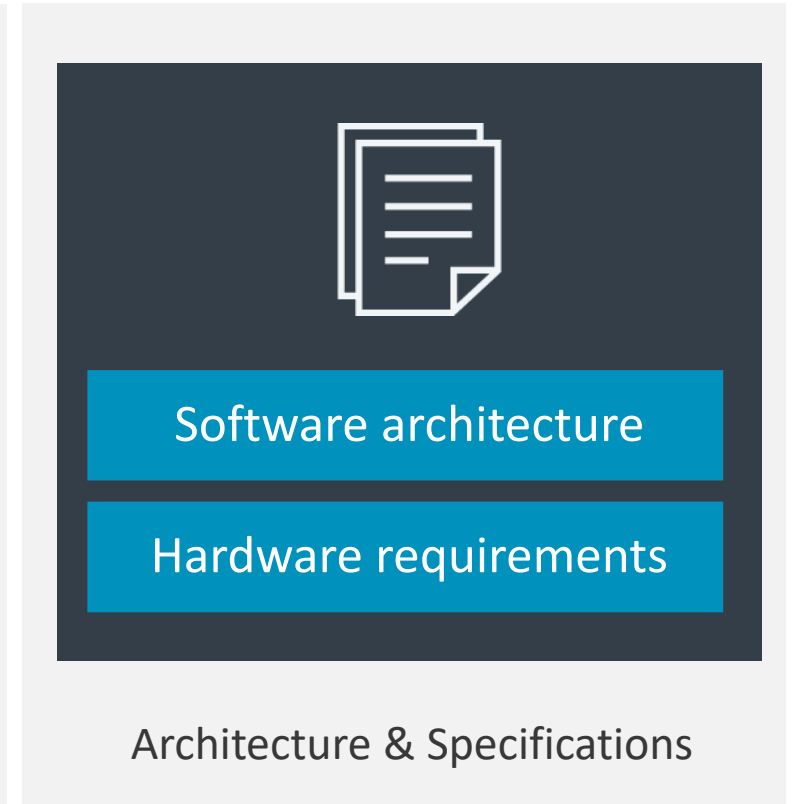
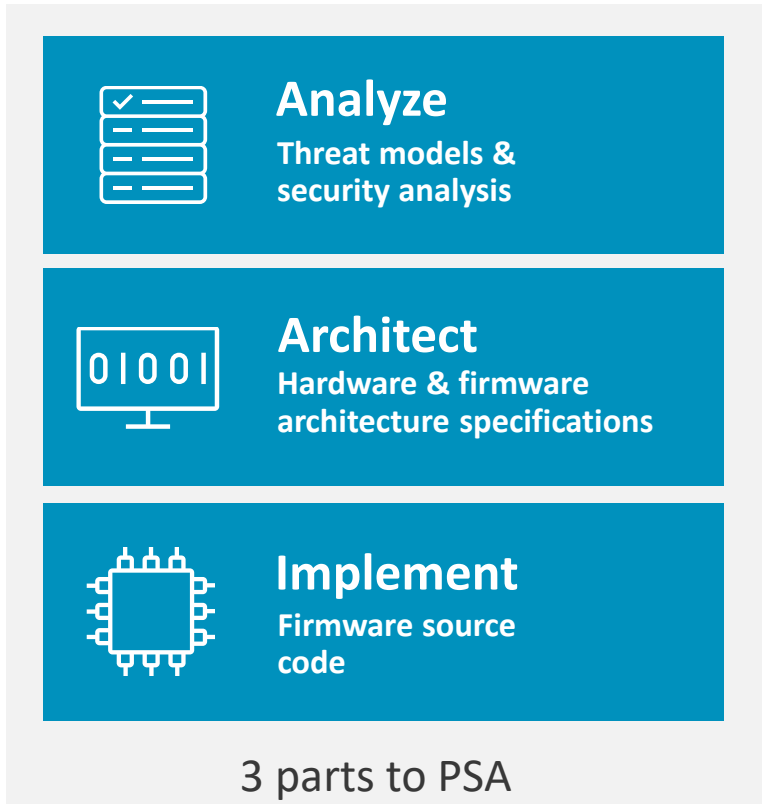
Tough to Scale to
the tiniest, low-cost
devices

Need to secure the
connection and
manage the devices



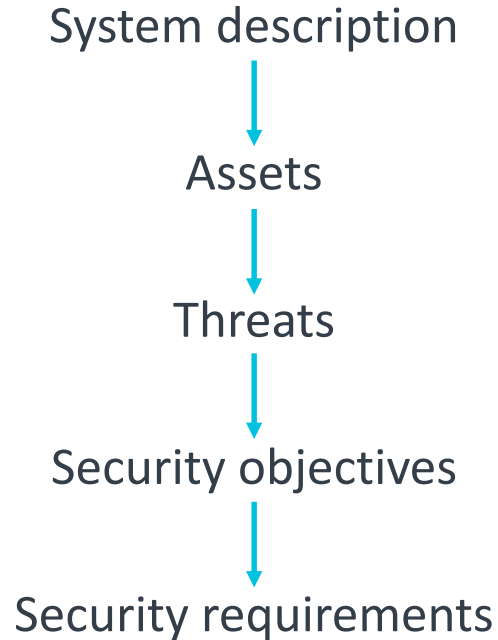
Platform Security Architecture (PSA)

A recipe for building a secure system & a reference implementation



Arm Provides Editable IoT Threat Models

Analysis leads to requirements



Understanding Security Requirements is an essential first step

Example

Asset: metering data to be protected in integrity & confidentiality

Threat: Remote SW attacks

Security objective: Strong Crypto

Security requirement: Hardware based key store



February'18 @ Embedded World

<https://pages.arm.com/psa-resources.html>

arm

Trusted Firmware-M Open Source Project

An open source project for rapid implementation

Trusted Firmware-M

Reference firmware for PSA architecture specification

Targeting M-profile SoCs (Initially Armv8-M)

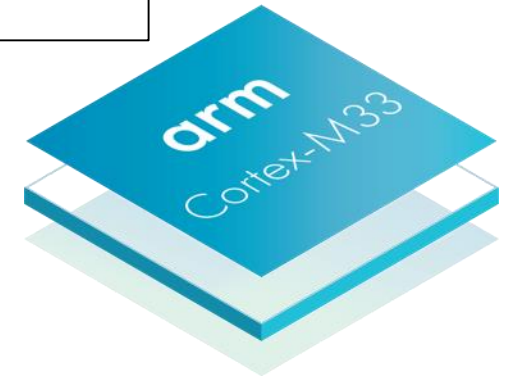
Available on www.trustedfirmware.org

Arm Mbed OS will include an implementation of PSA

Based on TF-M for secure services

Used by Mbed TLS, Pelion Device Mgmt & Mbed OS

Components being introduced now to
future Mbed releases



<https://connect.linaro.org/resources/hkg18/hkg18-212/>

April'18 @ Linaro Connect

PSA – Developer Crypto APIs

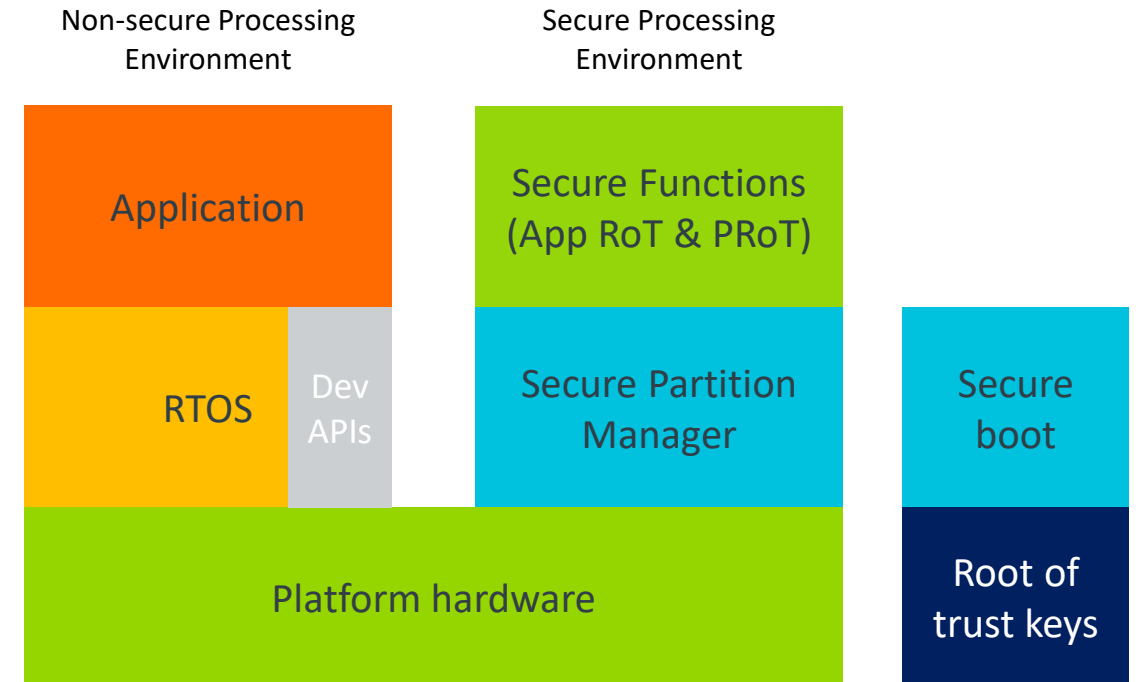
There are three sets of APIs in development. One of the most important is the top level, developer facing, APIs.

These APIs hide the hardware implementation details from the software engineer

Designed to be easy to use, with sensible defaults (more on this later)

```
1646 #define PSA_ASYMMETRIC_SIGN_OUTPUT_SIZE(key_type, key_bits, alg) \
1647     (PSA_KEY_TYPE_IS_RSA(key_type) ? ((void)alg, PSA_BITS_TO_BYTES(key_bits)) : \
1648     PSA_KEY_TYPE_IS_ECC(key_type) ? PSA_ECDSA_SIGNATURE_SIZE(key_bits) : \
1649     ((void)alg, 0))
1650
1651 /**
1652  * \brief Sign a hash or short message with a private key.
1653  *
1654  * \param key      Key slot containing an asymmetric key pair.
1655  * \param alg      A signature algorithm that is compatible with
1656  *                 the type of \c key.
```

https://git.trustedfirmware.org/trusted-firmware-m.git/tree/interface/include/psa_crypto.h



August - APIs for review

PSA Security Model – 10 Goals

Device should support

1. Unique instance ID
2. Attestation e.g. Entity Attestation Token
3. Secure Storage
4. Secure Boot
5. Isolation of ROT Services
6. Secure update process
7. Validation of updates
8. Anti-rollback feature
9. Security lifecycle supported with attestation
10. TRNG and Nonce services

Platform Security Architecture – One Year On

A complete security offering – now public & freely available

Analyze



Threat models
& security analyses



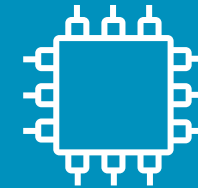
Architect



Hardware & firmware
architect specifications



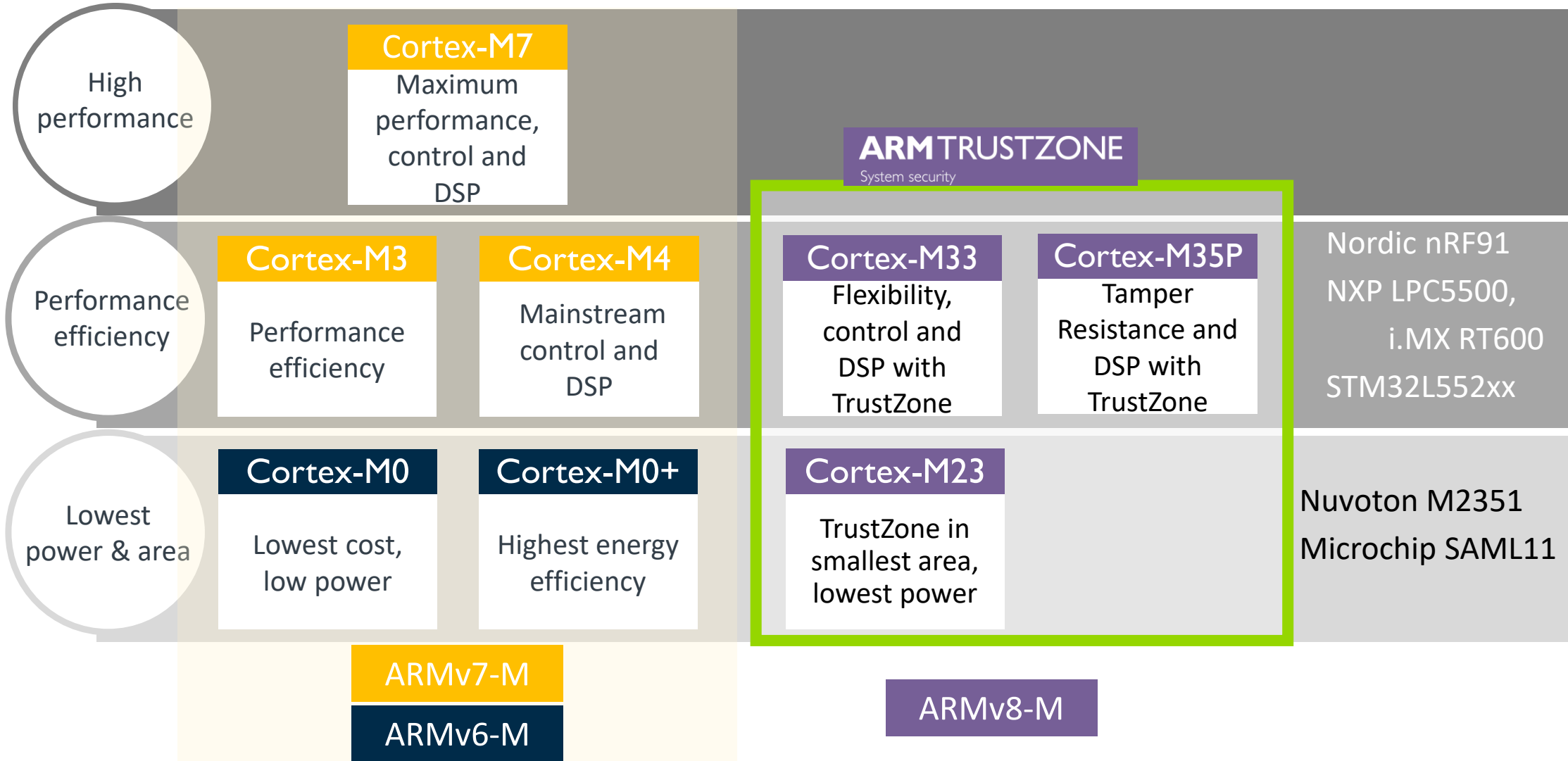
Implement



Firmware
source code



Bringing TrustZone to the Cortex-M family



PSA COMPLIANCE

Announcing the New APIs & Test Suites

A Common Developer Experience

Built on APIs and test suites

One of the core goals of PSA is making security easier for developers
Most software developers need simple to use high level APIs with sensible defaults

= PSA Developer APIs

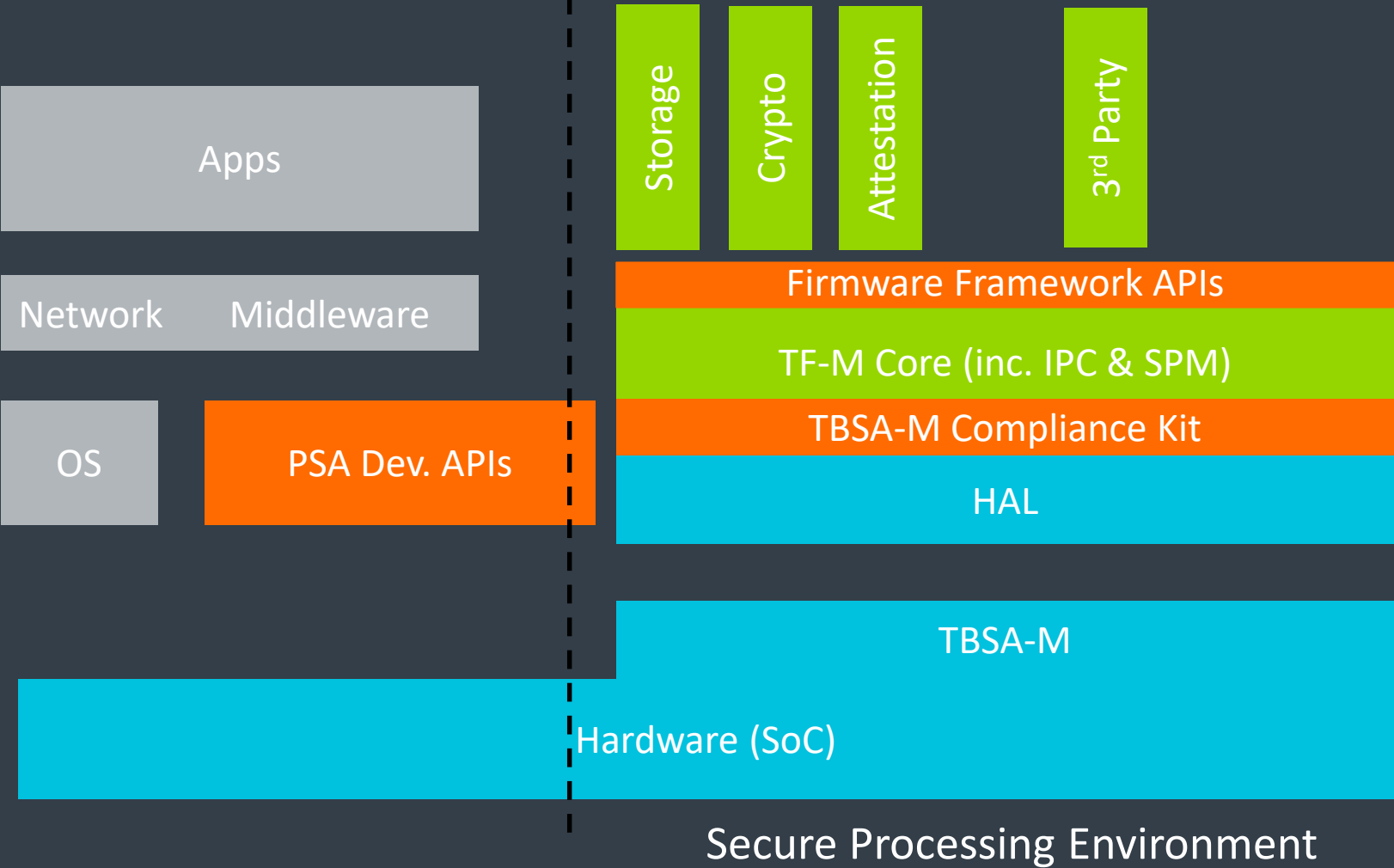
Security experts wanting to write their own secure functions (Application RoT services) should be enabled to do so in a standard way

= Firmware Framework APIs

Chip makers will want to have a standard way to interface their security hardware with firmware

= TBSA-M Architecture Compliance Kit /HAL

APIs & Compliance Kits



Three easy to use APIs

- Developer facing functional APIs for software dev and RTOS
- Firmware Framework & IPC APIs for 3rd party development of secure functions (Application RoT Services)
- TBSA APIs for silicon partners

https://git.trustedfirmware.org/trusted-firmware-m.git/tree/interface/include/psa_crypto.h

PSA Developer Crypto API

Foundations for device security

Developer-facing API

- Expose crypto services through a user-friendly interface
- Never reveal keys
- Store all keys into secure storage

Hardware Support

- Crypto acceleration
- Random Generation
- Secure Storage
- Secure Elements
- Secure functions running on a dedicated core

Applications

- Open a secure socket
 - Mutual authentication
 - Integrity
 - Confidentiality
- Validate signatures
- Authenticate device
- Sign attestations
- Authenticate incoming commands

PSA Developer Secure Storage API

Provide support for data-at-rest protection

Developer-facing API

- Key/Value oriented storage
- Protection through:
 - Device-bound encryption
 - Integrity
 - Rollback protection
 - Access control

Lower-level Services

- PSA Root of Trust uses internal flash, fuses, or secure elements to store device keys.
- Application Root of Trust can be based on external, untrusted storage: Secure Storage API provides the adequate protection level

Entity Attestation Token - Basics

A standard way to make signed claims about a device

EAT provides a trust signal

An Entity Attestation Token enables devices to make standard or bespoke claims
e.g. Device ID, Secure boot state

Uses CBOR Web Token (CWT) which scales to low-end devices

Cryptographically signed

Being standardized in IETF & GlobalPlatform

Entity Attestation Token

Device ID
Secure boot state
Firmware version
EAN-13 ...

Crypto Signing

PSA Developer Attestation API

Attest your device identity and capabilities using EAT*

Self-health reports

Challenge/response where devices report on:

- Serial Number
- Vendor
- Firmware version(s)
- Hardware capabilities

Enrollment

On-board key generation inside a secure enclave.


Dynamic identity enrolment

Change of ownership

Generated reports

Signed reports on measurements

*Entity Attestation Token

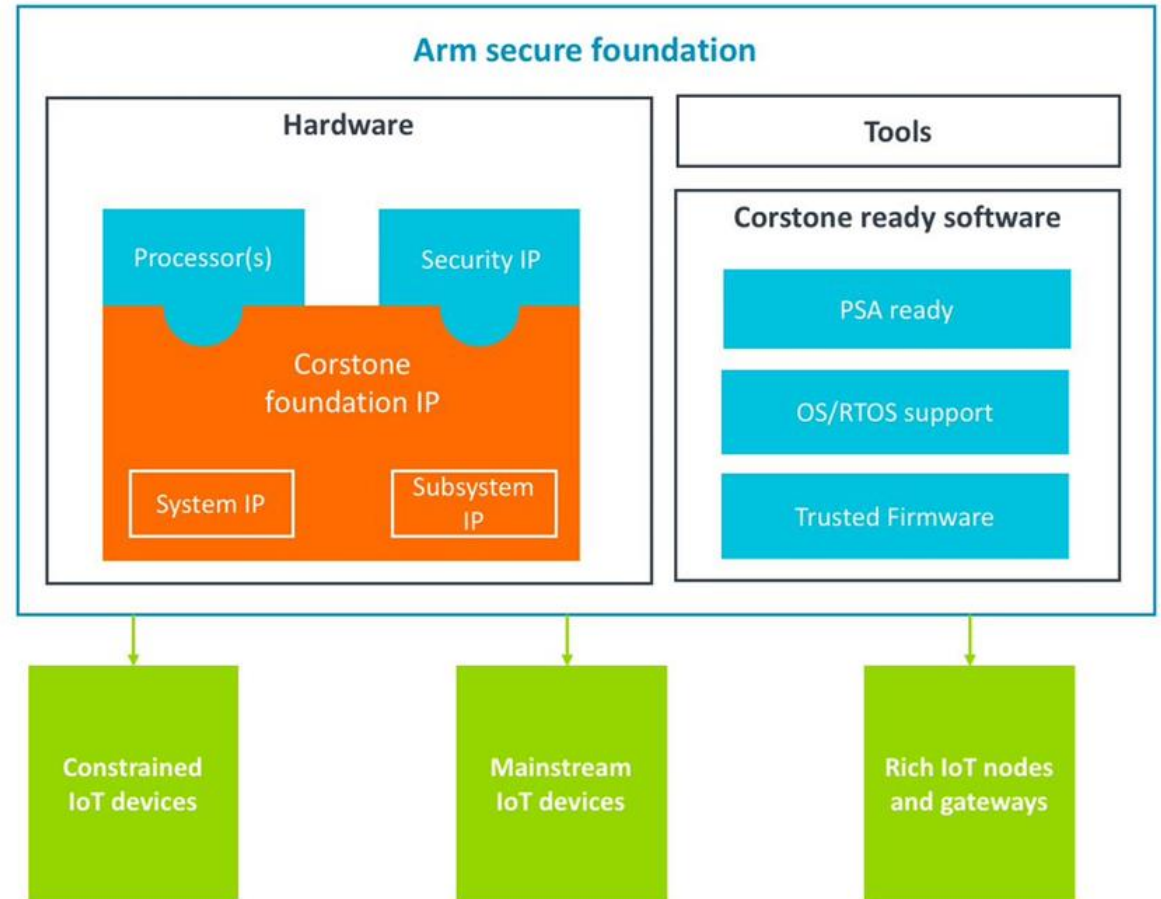


Getting Started with a Secure System

Arm secure foundation solutions

Complete system approach

- Corstone foundation IP (former SDKs):
 - Pre-verified, configurable system and subsystem IP
 - Modifiable subsystem IP
 - Pre-integrated with processor and security IP
- Development tools (including FPGA/test chip boards)
- Corstone ready software (e.g. Mbed OS)



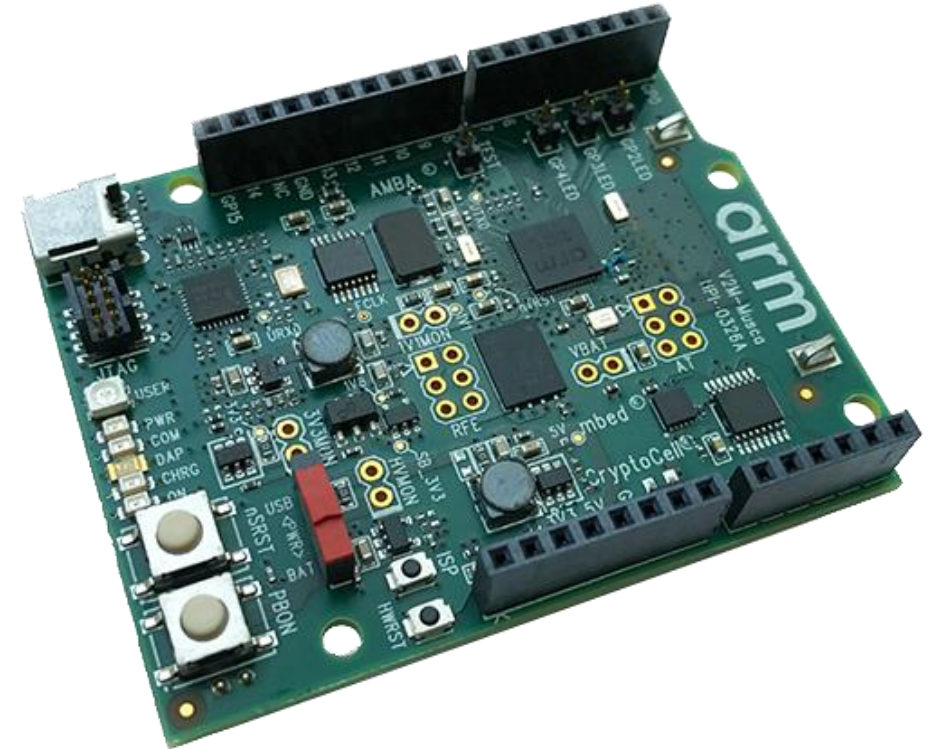
Development Platforms for PSA

Fixed Virtual Platform of SSE-200
(Arm website) or

Musca-A1: development board for PSA

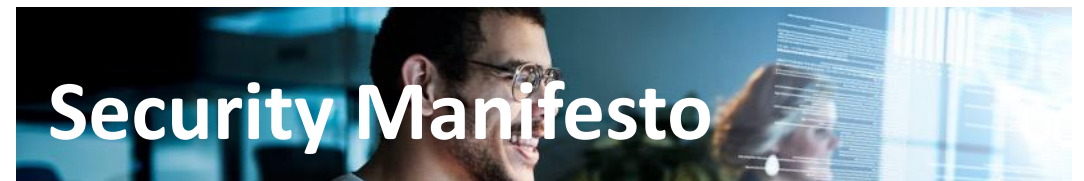
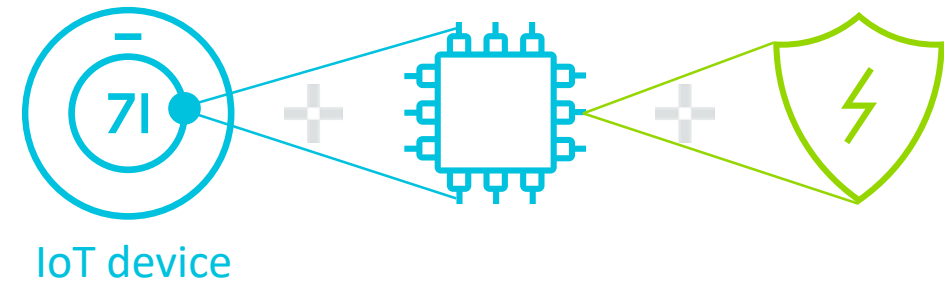
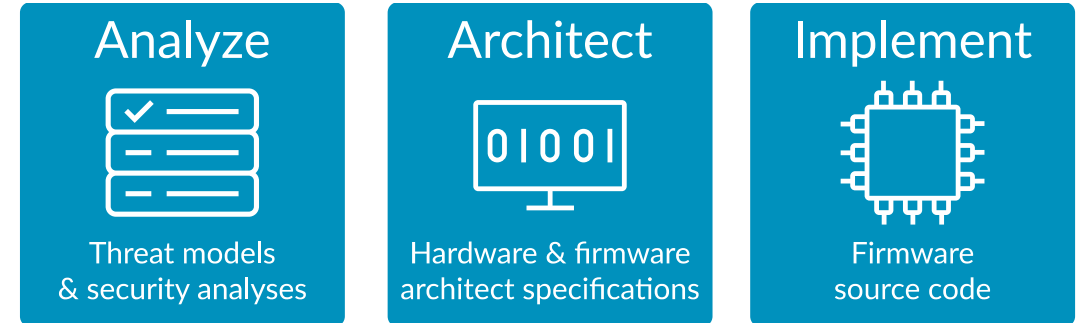
Request your free board (arm.com/musca)

- Arm Cortex-M33 based dev board
- Used for internal software development
- Test chip built on PSA recommendations
- Prototype your system



Summary – Making security easier

- PSA makes security easier to implement through a common architecture
- In the last 12 months we have delivered on threat models, open source & architecture documents
- Today we announced the PSA APIs and Architecture Compliance Kits to help build an ecosystem
- PSA provides a complete set of free security deliverables reducing TTM and cost



Trademark and copyright statement

The trademarks featured in this presentation are registered and/or unregistered trademarks of <COMPANY NAME> (or its subsidiaries) in the EU and/or elsewhere. All rights reserved. All other marks featured may be trademarks of their respective owners.

Copyright © 2018

Thank You!