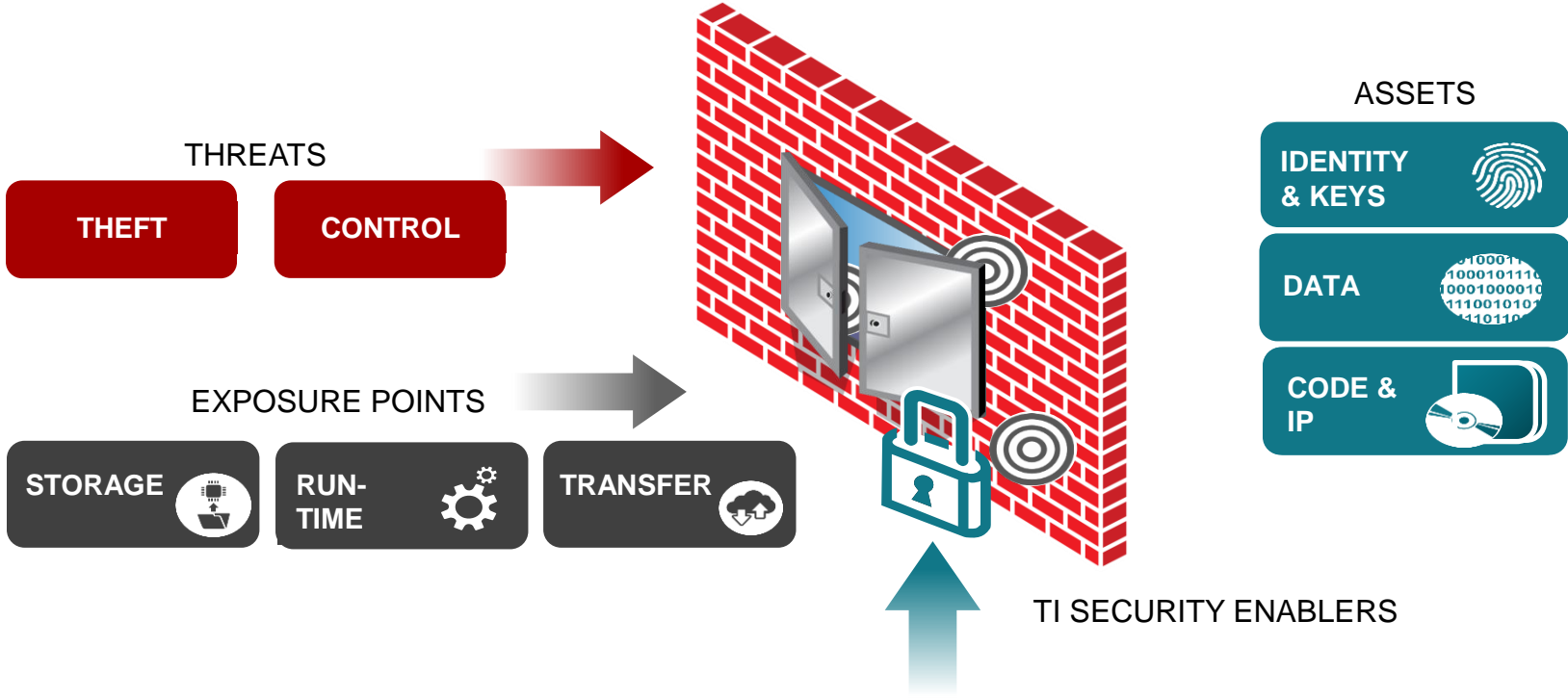


Application Processor Security

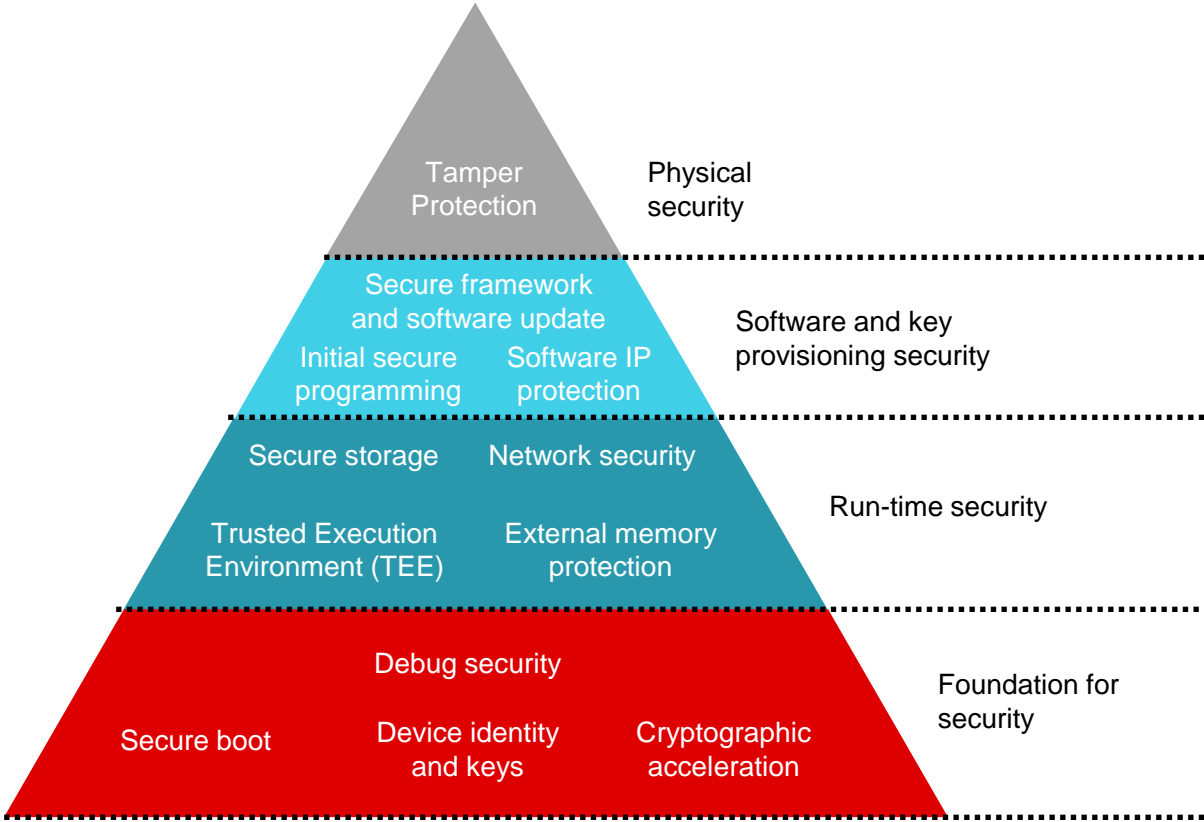
Processors Security Overview

Security basics and enablers



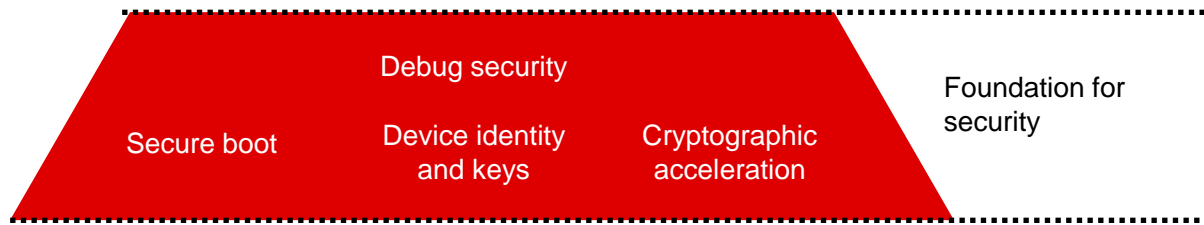
“TI’s security toolbox of **security enablers** helps address the emerging threats of an increasingly connected and complex world”

Security enablers in embedded systems

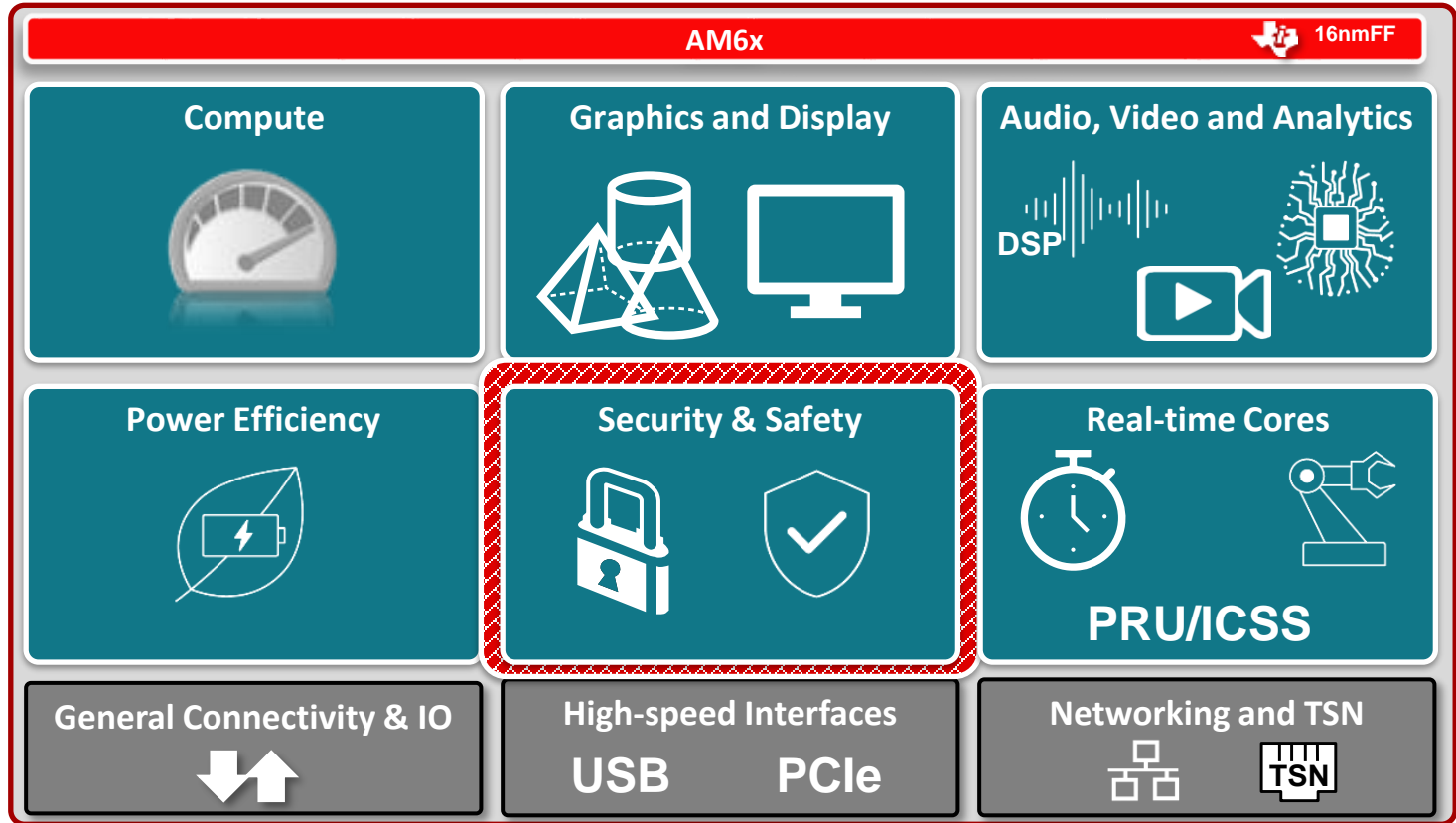


Security foundational enablers

| Processor Security Enabler | Description |
|----------------------------|---|
| Secure boot | Cornerstone of embedded security. Authenticates boot images and ensures their integrity. Establishes Root of Trust. |
| Device Identity and keys | Maintains device identity in a network |
| Crypto Acceleration | Dedicated HW acceleration for low latency and high throughput Crypto functions |
| Debug Security | Locks down JTAG in the Field. Board debug port can be opened only by authenticated SW. |



AM6x Cortex[®]-A based architecture



AM6x security

AM6x

 16nmFF

- Security is foundational to the overall architecture
- Isolation and separation of security functions
 - Security Controller (secure enclave)
 - ARM Trustzone

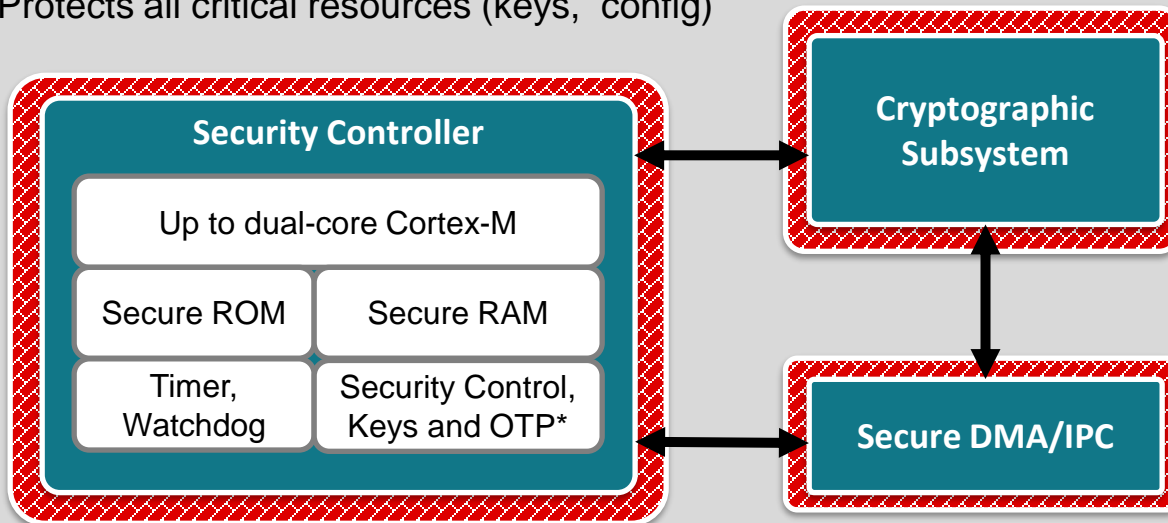


AM6x Security Controller

AM6x


16nmFF

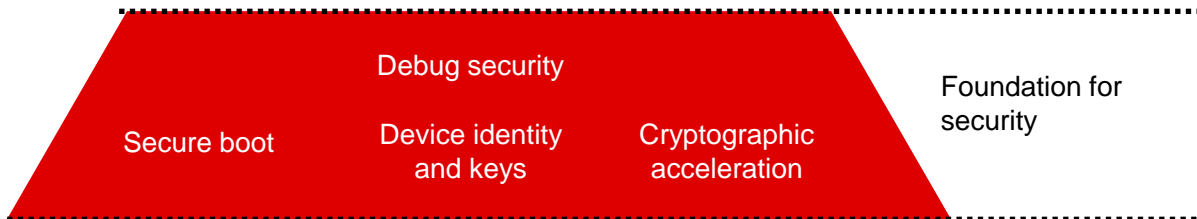
- Central control for security (secure boot, debug, etc.)
- Isolated from the rest of the system by firewalls
- Protects all critical resources (keys, config)



* OTP = One-Time Programmable Memory

Security foundational enablers

| Processor Security Enabler | Description |
|---|---|
| Secure boot  | Cornerstone of embedded security. Authenticates boot images and ensures their integrity. Establishes Root of Trust. |
| Device Identity and keys | Maintains device identity in a network |
| Crypto Acceleration | Dedicated HW acceleration for low latency and high throughput Crypto functions |
| Debug Security | Locks down JTAG in the Field. Board debug port can be opened only by authenticated SW. |

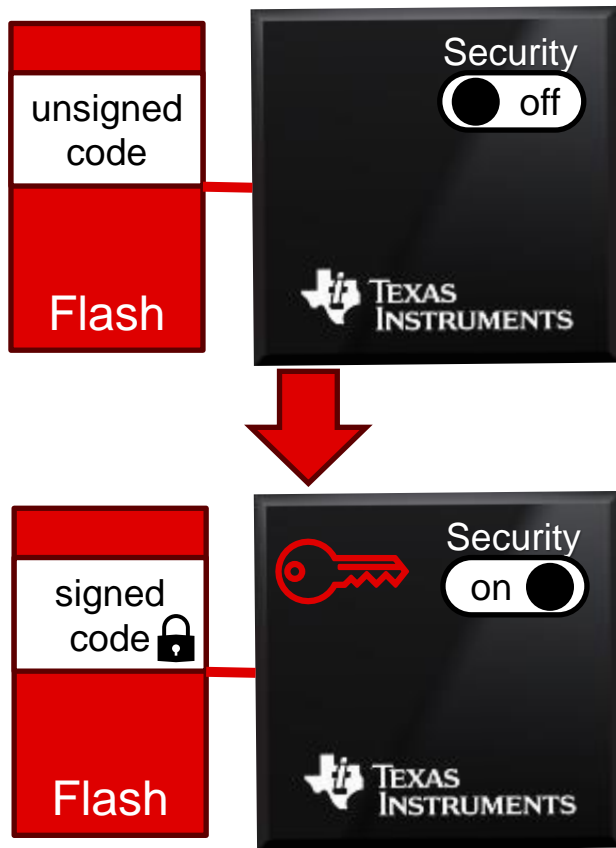


Secure boot

Secure boot is a hardware based “Root of Trust” to authenticate and protect boot code and data. Customers program their own keys using software/tools supplied by TI.

- When security is enabled, the device will only boot code specifically prepared for the device using asymmetric encryption and hashing
- Takeover Protection
 - My device only runs my software (authenticity and integrity)
 - Non-volatile one-time-programmable memory within device is configured so device will only boot “trusted” software. Ensure external flash content is not modified.
 - Overwriting flash or changing the boot source to load new code that is not signed will result in a boot failure
- Chain of Trust can be extended to following boot stages (i.e. OS or Application Image)

Non-secure Boot



Device Types – simplified development flow

General Purpose (GP)

- Device not used for secure operation
- No Security switch, security features disabled and cannot be enabled
- JTAG Enabled (unlocked)
- Crypto cores enabled



GP device cannot be changed to HS

High Security (HS)

HS - FS

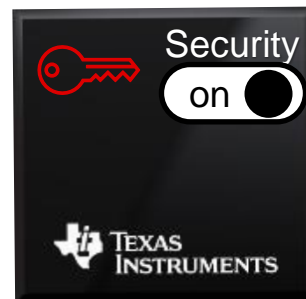
- Field Securable (FS)
- Development and testing
- No secure boot
- JTAG closed only for Security Controller
- Security largely disabled



HS-FS device type is shipped from TI for HS and typically included on EVMs

HS - SE

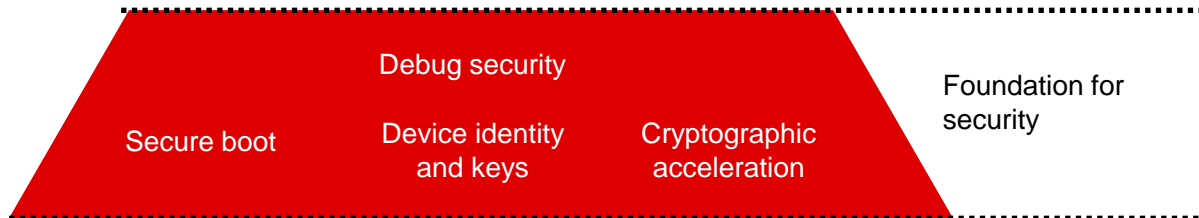
- Security Enforced (SE)
- Production
- Enforce secure boot
- JTAG closed completely
- All available security features active



non-reversible
OTP writing

Security foundational enablers

| Processor Security Enabler | Description |
|----------------------------|---|
| ✓ Secure boot | Cornerstone of embedded security. Authenticates boot images and ensures their integrity. Establishes Root of Trust. |
| Device Identity and keys | Maintains device identity in a network |
| Crypto Acceleration | Dedicated HW acceleration for low latency and high throughput Crypto functions |
| Debug Security | Locks down JTAG in the Field. Board debug port can be opened only by authenticated SW. |



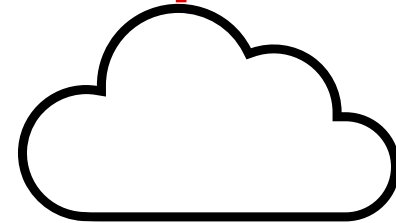
Device identity

- Hardware Unique Identification (ID)
- Useful for network authentication
 - Restrict access to pre-approved IDs
 - Contributes to auto-provisioning
 - Helps detect unauthorized access attempts
- Very helpful for fleet management
 - Often needed for software management and updates
 - Know what devices are running what software at all times
 - Confidently monitor device status and life cycle



“I am...”

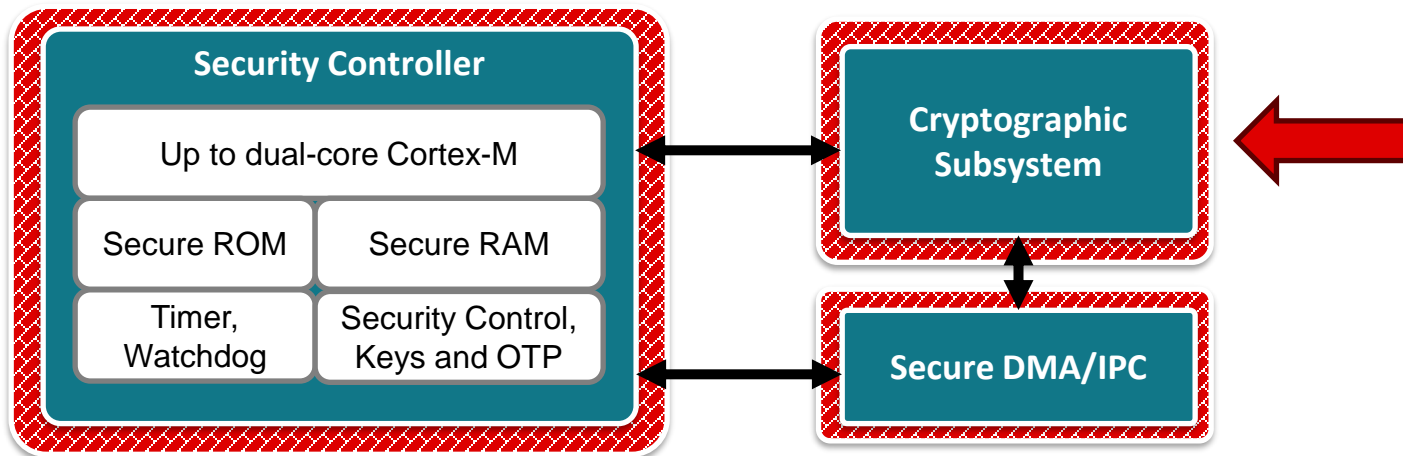
“Who are you?”



Cryptographic acceleration

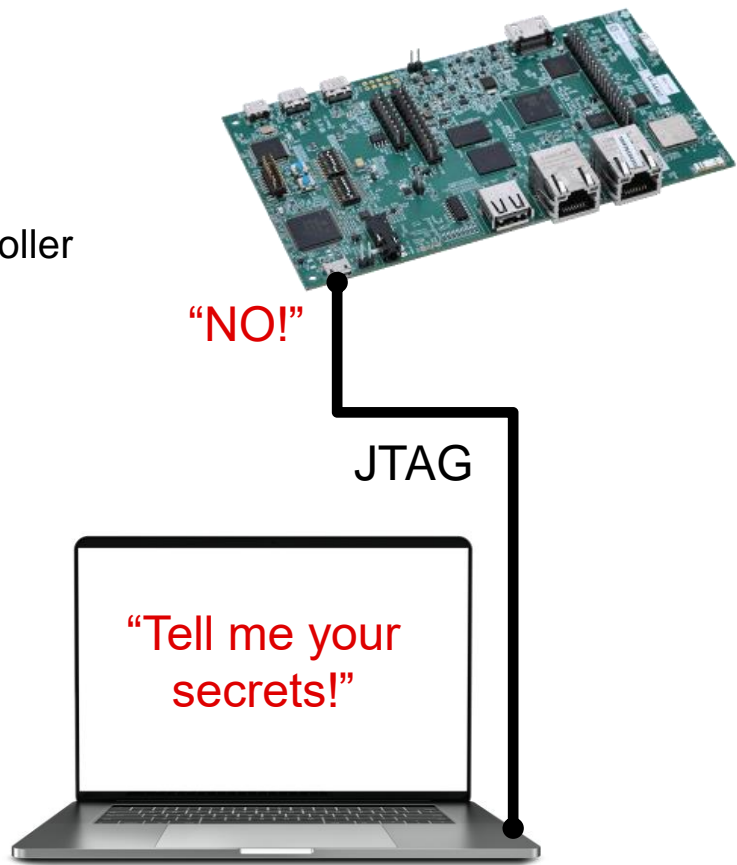
- Accelerates secure boot operations
- Smart context for non-secure world use
- Supports popular industry standards
 - Asymmetric, Symmetric, Hashing and Random Number Generation

| | |
|--------------------------------|---|
| AES : 128,192, 256 | ✓ |
| SHA2: 224, 256, 384, 512 | ✓ |
| Chinese Crypto (SM2, SM3, SM4) | ✓ |
| True RNG | ✓ |
| DRBG | ✓ |
| PKA: RSA 2K acceleration | ✓ |
| PKA: RSA 4K acceleration | ✓ |
| PKA: ECC acceleration | ✓ |



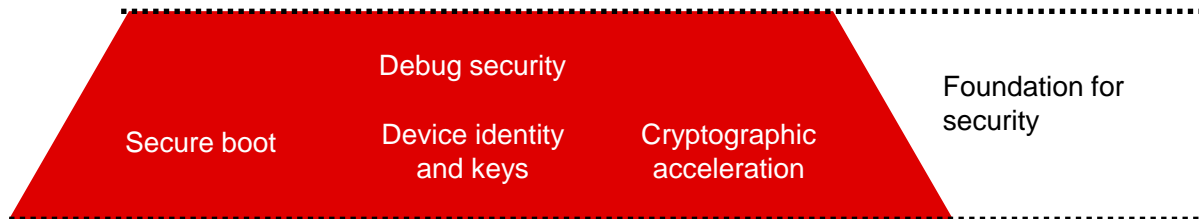
JTAG debug security

- Debug (JTAG) port
 - Closed by default on High-Security (HS) devices
 - HS-FS protects JTAG access to the Security Controller
 - HS-SE locks JTAG port
 - Can be closed permanently via eFuse setting
 - Software unlock JTAG with proper verification
 - Device UID or Wildcard match
- Device UID (Unique ID) for unlock
 - Available from tools and software
 - Also output during peripheral boot modes
 - USB, UART & Ethernet



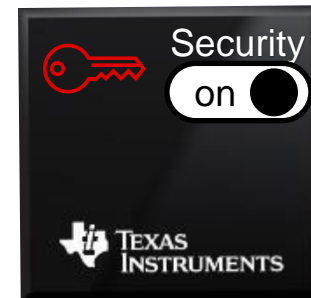
Security foundational enablers

| Processor Security Enabler | Description |
|----------------------------|---|
| ✓ Secure boot | Cornerstone of embedded security. Authenticates boot images and ensures their integrity. Establishes Root of Trust. |
| ✓ Device Identity and keys | Maintains device identity in a network |
| ✓ Crypto Acceleration | Dedicated HW acceleration for low latency and high throughput Crypto functions |
| ✓ Debug Security | Locks down JTAG in the Field. Board debug port can be opened only by authenticated SW. |



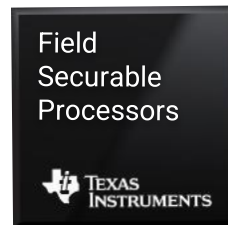
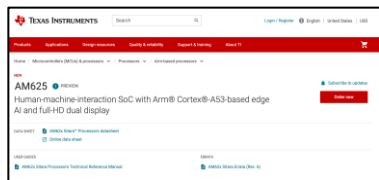
Processor security feature highlights

- **Security Controller**
 - Up to Dual-core security coprocessor for centralized security control
 - Reduces attack surface for critical assets (e.g. Keys)
 - Dedicated, Crypto channel, DMA and IPC for HSM
- **Field securable (HS-FS) device type**
 - Device behaves as non-secure device until the keys are programmed
 - Helps customers test their hardware prior to productization with secure keys
- **Secure boot**
 - Customer can program hardware (efuse) keys themselves
 - Supports encrypted and authenticated boot
 - **Uses industry standard x509-based booting certificates**
- **Dual root key support**
 - Support for two sets of hardware (efuse) root keys
 - Enables customers to switch to a new root key in the field
- **Enhanced firewall architecture**
 - Dynamic access control to all SoC resources (memories, peripherals, cores, etc.)
 - Provides the ability to promote or demote access to resources
- **Smart cryptographic subsystem**
 - Chinese Crypto support
 - ECDSA and DRBG standards hardware support, in addition to AES, 3DES, SHA1/2, MD5
 - Ability to proxy security master (e.g. SMS) to promote or demote incoming data streams
 - Improved performance with ability to push data streams to secure world with minimal context switching
- **Enhanced debug control**
 - Security aware debugging (e.g. ability to lock secure world while debugging public world)
 - SMS controlled challenge-response protocol for opening debug



Processor security experience

One flow for non-secure and secure designs. Simply add security when/if needed with minimal impact to the design. Consistent across TI AM6x Processors.



- One ti.com presence
- One set of collateral (TRM, DS, etc.)
- One set of benchmarks
- One support path (e2e)

- One EVM
- One SDK
- One set of tools
- One software flow

- One device to sample
- One board design
- One device throughout design
- **One security experience that can be reused across projects**

One Decision: easily add security when needed to any TI HS Processor

Security getting started

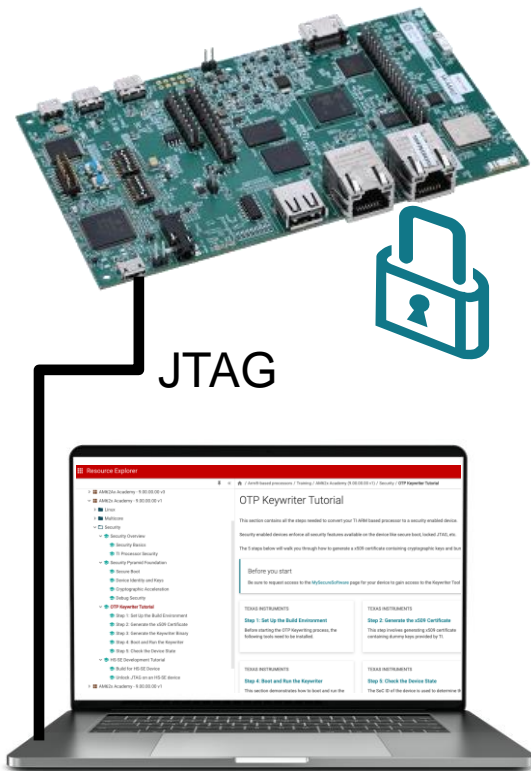
Start using security today with a current Starter Kit (SK), Software Development Kits (SDKs), and tools with **Security Academy!**

Learn hands-on how to use secure boot and JTAG:

1. Sign software with TI "shared" private keys
2. Program "known" public keys to a device
3. Verify secure boot
4. Unlock JTAG for debug

| Device | Academy |
|--------|----------------------|
| AM62x | Link |
| AM62Ax | Link |

step by step
instructions



Note: This process is very similar for all AM6x family members...

Thank You!

Security Features Comparison 1/2

| Enabler | Feature | AM335x | AM437x | AM438x | AM570x/ AM574x | AM64x/ AM243x | AM62x/A/P/ AM67x | AM68x/ AM69x | TDA4/ DRA8 |
|---------------------------|--|--------|--------|--------|-------------------|------------------|---------------------|-----------------|---------------|
| Cryptography Acceleration | AES : 128,192, 256 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| | 3DES : DES , 3DES | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| | SHA2-224, 256 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| | SHA2: 384, 512 | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| | Chinese Crypto (SM2, SM3, SM4) | | | | | | ✓ | | |
| | True RNG | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| | DRBG | | | | | ✓ | ✓ | ✓ | ✓ |
| | PKA: RSA 2K acceleration | | | ✓ | | ✓ | ✓ | ✓ | ✓ |
| | PKA: RSA 4K acceleration | | | ✓ | | ✓ | ✓ | ✓ | ✓ |
| | PKA: ECC acceleration | | | | | ✓ | ✓ | ✓ | ✓ |
| DMA support for crypto | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | |
| Device ID and Keys | Device Public ID via ROM API | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| | Bootimg keys via OTP (MPK/SMPK, KEK, MEK/SMEK) | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| | General OTP bits and other OTP (MSV, SWRV) | ✓ | ✓ | ✓ | ✓ | ✓ (384b) | ✓ (1024) | ✓ (1024b) | ✓ (1024b) |
| Secure Boot | Customer blows own keys (Standard secure) | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| | RSA 2048-based | ✓ | ✓ | ✓ | ✓ | ✓* | ✓* | ✓* | ✓* |
| | RSA 4096-based | | ✓ | ✓ | | ✓ | ✓ | ✓ | ✓ |
| | ECDSA (NIST Curve) | | | | | ✓* | ✓* | ✓* | ✓* |
| | Authenticated boot | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| | Encrypted boot | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| | Model ID check | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Anti-roll back check | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | |
| Debug Security | SW-controlled JTAG access | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| | Permanent disable JTAG via efuse | | | | | ✓ | ✓ | ✓ | ✓ |
| | Security-aware debugging | | | ✓ | | ✓ | ✓ | ✓ | ✓ |

Not all features may be enabled by software.

Security Features Comparison 2/2

| Enabler | Feature | AM335x | AM437x | AM438x | AM570x/ AM574x | AM64x/ AM243x | AM62x/A/P/ AM67x | AM68x/ AM69x | TDA4/ DRA8 |
|-------------------------------------|---|--------|--------|--------|-------------------|------------------|---------------------|-----------------|---------------|
| External Memory Protection | DDR obfuscation | ✓ | ✓ | ✓ | ✓ | | | | |
| | DDR encryption | | | ✓ | | | | | |
| | OSPI/HyperFlash Encryption | | | | | ✓* | ✓* | ✓* | ✓* |
| Trusted Execution Environment (TEE) | ARM TrustZone: CPU, L1/L2 cache, GIC | | ✓ | ✓ | ✓ | ✓** | ✓** | ✓** | ✓** |
| | Firewall: RAM, DDR, peripherals | | ✓ | ✓ | ✓ | ✓** | ✓** | ✓** | ✓** |
| | Secured crypto context | | ✓ | ✓ | ✓ | ✓** | ✓** | ✓** | ✓** |
| | Secure DMA/data-path | | ✓ | ✓ | ✓ | ✓** | ✓** | ✓** | ✓** |
| | Secure WDG/Timer | | ✓ | ✓ | ✓ | ✓** | ✓** | ✓** | ✓** |
| | Clock, reset, power management security | | ✓ | ✓ | | ✓** | ✓** | ✓** | ✓** |
| | Secure IPC | | ✓ | ✓ | ✓ | ✓** | ✓** | ✓** | ✓** |
| Networking Security | IPSEC data path acceleration (no inline) | | | | | ✓* | ✓* | ✓* | ✓* |
| | SRTP/TLS data path acceleration (no inline) | | | | | ✓* | ✓* | ✓* | ✓* |
| | Auto key material fetch | | | | | ✓* | ✓* | ✓* | ✓* |
| Secure Storage | Secure storage using Linaro OPTEE solution | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Software IP Protection | Encrypted boot | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| | ROM/SW API for software protection | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Initial Secure Programming | Secure key programming (MEK/SMEK) | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| | Random key programming (KEK) | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Secure Firmware & Update | Secure boot loaders (PPA, DMSC etc) | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| | Aux keys for updates | ✓ | ✓ | ✓ | ✓ | ✓* | ✓* | ✓* | ✓* |
| Physical Security | Environmental monitoring (V, F temp, clock) | | | ✓ | | | | | |
| | Glitch protection (voltage, frequency) | | | ✓ | | | | | |
| | Enclosure protection (wire mesh, switch) | | | ✓ | | | | | |
| | 1-cycle clear register in case of event | | | ✓ | | | | | |
| | Internal secure timestamp | | | ✓ | | | | | |
| | Laser/IR deterrent device physical layout | | | ✓ | | | | | |

** TI supports OPTEE as runtime SW solution in Linux/Android SDK. Must consult 3P for any other OS.

Not all features may be enabled by software.