FOUNDATIONAL TRUST FOR IOT AND RESOURCE CONSTRAINED DEVICES

Vulnerabilities in modern computing, communications, and control systems allow cyber-attackers, who are increasingly sophisticated and relentless in their efforts, access to valuable, proprietary, personal, and sometimes, dangerous information. At the same time, new market segments, like the Internet of Things (IoT), are driving innovative architectures and creating solutions with challenging power, security, resource, and other constraints.

These constraints make an optimal security posture more difficult to create and maintain. While systems with a Trusted Platform Module (TPM) have many practical and flexible security benefits, the TPM is not feasible for most devices in the IoT space, like microcontroller units (MCUs) or systems on a chip (SoCs), or in components of more complex systems like peripheral devices or sensors in advanced driver-assistance systems (ADAS).

To address the need for increased security in IoT and other advanced products, systems and applications, the Trusted Computing Group (TCG) has established the DICE Architectures, or DiceArch (pronounced dīsˌärk) Work Group. Based on the Trusted Platform Architecture Hardware Requirements for a Device Identifier Composition Engine (DICE) draft specification, the work group is exploring new security and privacy technologies applicable to systems and components with or without a TPM.

The goal is to develop new approaches to enhancing security and privacy with minimal silicon requirements. Even simple computing capabilities combined with software techniques can establish a cryptographically strong device identity, attest software and security policy, and enable safe deployment and verification of software updates. These are all valuable security enhancing capabilities.

DICE works by breaking up boot into layers and creating secrets unique to each layer and configuration based on a Unique Device Secret (UDS). If different code or configuration is booted, at any point in the chain, the secrets will be different. Each software layer keeps the secret it receives completely confidential to itself. If a vulnerability exists and a secret is disclosed, patching the code automatically creates a new secret, effectively re-keying the device.

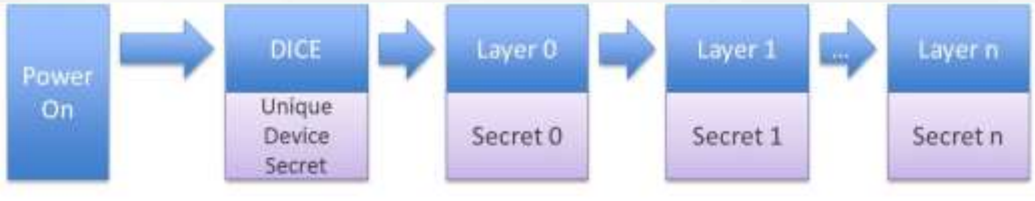

Figure 1. DICE starts unconditionally at power-on. DICE has exclusive access to the UDS. Each layer computes the secret for the next layer using a one-way function and a measurement of the next layer

The Implicit Identity Based Device Attestation draft reference document from the DiceArch WG details a use case built on the DICE draft specification. This architecture describes keys, crypto operations, and certificates for a cryptographic Device Attestation scheme. In addition to strong hardware-based Device Identity, and Device Attestation, one possible use for this architecture is as a foundation for a secure storage (sealing) implementation in resource-constrained devices. Compatible with IEEE 802.1AR - Secure Device Identity, the solution is intended for devices containing a Device Identifier Composition Engine.

Using a DICE Compound Device Identifier (CDI) as a basis for Device Identity, the solution involves basic assumptions for design constraints. For example, it assumes the Device Identity will be represented cryptographically as an asymmetric key pair so the public portion can be freely shared while the private portion remains secret. The private portion of this key is used to prove the device's identity. The benefit of limiting the number of assumptions is that it maximizes the set of Device Identity and Attestation scenarios that the reference supports.

While presuming DICE support in hardware, how the UDS is provisioned within a device is not specified - only that it has been provisioned. Also, since there is a clear advantage in relieving device manufacturers and vendors of the burden of maintaining secret databases of UDS values, this architecture presents a solution that does not require secret databases of UDS values.

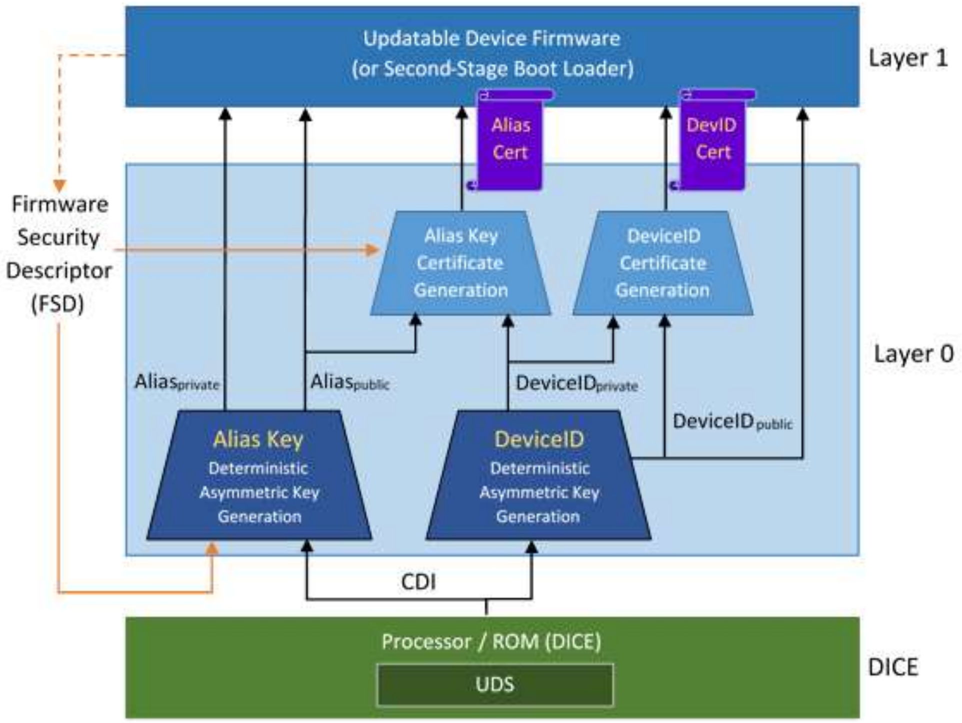
Updatable Device Firmware
(or Second-Stage Boot Loader)
Layer 1
Alias Cert
DevID Cert
Firmware Security Descriptor (FSD)
Alias Key Certificate Generation
DeviceID Certificate Generation
Layer 0
$\text{Alias}_{\text{private}}$
$\text{Alias}_{\text{public}}$
$\text{DeviceID}_{\text{private}}$
$\text{DeviceID}_{\text{public}}$
Alias Key Deterministic Asymmetric Key Generation
DeviceID Deterministic Asymmetric Key Generation
CDI
Processor / ROM (DICE)
UDS
DICE

Figure 2. Implicit Identity Based Device Identity Architecture

In this architecture, DICE is the root of trust for measurement. Since its misbehavior cannot be detected, it must be inherently trusted. The architecture relies on DICE unconditionally generating the correct CDI for Layer 0 with Layer 0 being the next link in the chain of trust. DICE establishes that the device booted the First Mutable Code provided by the manufacturer. This enables detection of persistent modification of Layer 0 and above.

The diagram provides detail for First Mutable Code (Layer 0) because this layer is responsible for constructing the foundational identity and attestation elements upon which Device Firmware (Layer 1) relies. Comparatively, the use case places few requirements on Device Firmware and the operation of the DICE layer.

Rolling out the DICE

TCG's DiceArch approach holds promise to enhance security and privacy for systems with a TPM and provide viable security and privacy foundations for systems without a TPM. The work group is focused on requirements, use cases, security/privacy benefits, and end-to-end solutions for software architectures and APIs, based on DICE Architectures.

Today, Micron, Microsoft, STMicroelectronics and others actively involved in the work group are already supporting DiceArch so interested users can immediately start evaluating and even implementing it in their products and systems.

Resources
https://trustedcomputinggroup.org/work-groups/dice-architectures/
Micron: http://investors.micron.com/releasedetail.cfm?releaseid=1022470
Microsoft: https://azure.microsoft.com/en-us/blog/azure-iot-supports-new-security-hardware-to-strengthen-iot-security/