The importance of a secure supply chain...

35C3 - https://youtu.be/C7H3V7tkxeA
# THE SEVEN PROPERTIES

<table>
<thead>
<tr>
<th>Property</th>
<th>Key Questions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware-based Root of Trust</td>
<td>Does the device have a unique, unforgeable identity that is inseparable from the hardware?</td>
</tr>
<tr>
<td>Small Trusted Computing Base</td>
<td>Is most of the device's software outside the device's trusted computing base?</td>
</tr>
<tr>
<td>Defense in Depth</td>
<td>Is the device still protected if the security of one layer of device software is breached?</td>
</tr>
<tr>
<td>Compartmentalization</td>
<td>Does a failure in one component of the device require a reboot of the entire device to return to operation?</td>
</tr>
<tr>
<td>Certificate-based Authentication</td>
<td>Does the device use certificates instead of passwords for authentication?</td>
</tr>
<tr>
<td>Renewable Security</td>
<td>Is the device's software updated automatically?</td>
</tr>
<tr>
<td>Failure Reporting</td>
<td>Does the device report failures to its manufacturer?</td>
</tr>
</tbody>
</table>
Choice of Secure Hardware

- Secure silicon vendors
  MCUs, SoCs, etc.
- Secure silicon standards
  • DICE
  • TPMs

and Device Firmware

- Language and OS agnostic
- Secure silicon hardware
- Critical security services

Device Security

Hardware Root of Trust in secure silicon + High-integrity software operations
WHY HARDWARE SUPPORT?

• Serious problems with software-only solutions
• If a bug leads to disclosure of Device Identity secret, then how do we securely (and remotely) recover and re-provision a device?
• Cannot trust software to report its own health (Attestation)
• Roots of Trust (RoT), data encryption, entropy, etc.
  • How do we securely extend trust chain, store keys, etc.?
  • Need a hardware RoT
BEWARE SIMPLE HW SOLUTIONS

• Why not just store Device Identity key/secrets in fuses?
  • If malware can manage to read the fused key, then you are no better off than with a software-based key
• TPMs are great but, especially in IoT solutions, systems and components probably won’t have TPMs or similar silicon-based capabilities (cost, complexity, physical space on the MCU/SoC)
• We need something different
THE DICE MODEL

• In a DICE Architecture device startup (boot) is layered
• Beginning with a Unique Device Secret (UDS), secrets/keys are created unique to the device and each layer and configuration
• This derivation method means if different code or configuration is booted, secrets are different
• If a vulnerability exists and a secret is disclosed, patching the code automatically re-keys the device
WHEN SOMETHING CHANGES

- Updates provide a way to recover a device or component if bad code leaks a secret.
A DICE ARCHITECTURE (RIOT)

- Underlying architecture for HW-based Device Identity and Attestation (Azure)
- DeviceID – Stable and well protected long-term identifier for a device or component
- Alias Key – Derived from combination of unique device identity (HW) and identity of Device Firmware (SW)
- Integrates DICE-enabled HW with existing infrastructure
ENABLES KEY SCENARIOS

• Flexible security framework to enable many high-value scenarios
• Secure remote device recovery (Cyber Resilient Technology Initiative)
  • Recover unresponsive (i.e., p0wned, hung, etc.) devices
  • Greatly reduced cost: no need for physical device interaction
• Supply chain management (DICE-enabled components, e.g. Cerberus)
  • Several recent damaging cyber-attacks were the result of malware introduced in the supply chain
  • DICE attestation lets end-customers trust far less of the supply-chain, e.g., just the storage-subsystem or flash vendor
• Strong cryptographic identity, authenticity, licensing, and many more
ADOPTION

• Microsoft Azure IoT – Device Provisioning Service, imx-iotcore BSP
• Microchip – CEC1702 and CEC1302, SecureIoT
• NXP – i.MX and beyond, Layerscape LS1012
• WinBond – TrustME Secure Flash Memory
• Micron – Authenta Secure Flash Memory
• STMicroelectronics – SDTM32L0/L4 MCUs
• Others we can’t talk about yet (but they’re here at the show!)
• Open source – Project Cerberus, ARM Trusted Firmware (prototype)
• More announcements soon!
DICE TAKEAWAYS

• Flexible security framework, not one size fits all
• Minimal silicon requirements, low barrier to entry
• Foundation for strong cryptographic HW-based device identity and attestation, data at rest protection (sealing), and secure device update and recovery
• Public announcements from SoC, MCU, and flash memory vendors so far with more on the way
• Represents the ongoing work of the DICE Architectures Workgroup (DiceArch WG) in TCG. Come join us!