Common Security Threats v1.0

EDITOR: Yigal Edery, Kameleon

CONTRIBUTORS:

Matt King, Oracle

Elaine Palmer, IBM Corporation

Eric Spada, Broadcom

Rob Wood, NCC Group

Alebrto Munoz, Intel

Revision History

Revision	Date	Guiding Contributor(s)	Description
1.0	2020-11-15	Yigal Edery, Kameleon	Initial Release

Table of Contents

Purpose & Audience

This document is intended to help create a common understanding of the scope of possible threats on hardware/firmware, and how they map to OCP Security workgroup standards & solutions.

The audience for this document includes, but is not limited to, system and system component designers, security information and event management (SIEM) system developers, and cloud service providers. In general, this document is for anyone who wants to understand the scope of threats that is being addressed by the OCP-security specs.

Note that this document is not a requirements document by itself. It can be seen as a 'mapping' between the relevant threats and the OCP security requirements document that addresses these types of threats.

Also note that the threats in the document are kept high level on purpose. Detailed corner cases and threats that are deep in the domain of a specific spec are covered in their respective specs.

Threats Landscape

This section maps the general possible threat vectors for OCP systems being secured by an OCP-Security spec related technologies, and their applicable mitigations, as well as explicitly call out threats that OCP Security project is not attempting to mitigate. For a standardized definition of the terms used in this document, please refer to the NIST SP800-154 special publication.

Threats in this document are divided into four groups:

- **In-scope:** Threats that current version of OCP specs (V1.0) address and have explicit measures defined on how to deal with them. This group is further divided into threats that could be launched via logical attack alone, and threats that in order to successfully launch them, physical access to the device is needed.
- **Explicit Assumptions:** threats that were addressed by making some explicit assumptions about the relevant environment, in order to mitigate them.
- **Out-of-scope:** Threats that the team either didn't yet address in detail in this iteration (which means they are great potential for future work), or considered them to be irrelevant to the groups charter for now.

The main "asset" being protected in V1.0 of the specs is the firmware in the system, and this was used as a primary prioritization criteria when deciding on scope and relevant threats.

In-Scope Threats

All threats listed in this section should be considered applicable both at a system level as well as an individual component level. This means systems, management subsystems, storage subsystems, adapters, are all considered within scope, as one security ecosystem.

Also, to help clarify the type/source of possible threat, this document lists separately threats that are logical (i.e. software only attacks) vs. threats that require physical access to the protected device.

In-Scope Threats - Attackers With Logical Access

The following table lists in-scope threats that only require at attacker to have "Logical" access to the protected device (i.e. being able to run software on the protected device):

Threat	OCP Mitigation(s)
Execution of unauthorized firmware. For example, loading unsigned or badly signed firmware, or firmware not matching white list hashes, or cloning software that was meant for one system onto another. Note that this threat includes both unauthorized firmware running on the main platform (e.g. an unauthorized BIOS), as well as on any of the devices/peripherals connected to it (e.g. a NIC device firmware).	**Secure Boot** - Authenticate signature over firmware/configuration or validate against the securely stored hash of a previously authenticated boot image **Secure Update** - signature checks during FW updates. **Attestation** - verify measurements of devices before admitting them into the platform.
Execution of authorized firmware containing known security defect(s). For example, loading and running an old, properly signed version, but with known exploitable bugs.	**Secure Boot & Secure Update** - Roll-back prevention. **Attestation** - detect measurements of defective firmware, then initiate **Secure Update**
Compromised private signing keys. For example, an attacker gains access to an OxM signing key and signs their own firmware.	**Secure Boot** & **Attestation** - support for key and/or ownership revocation.
Compromised device verification keys or security policies. For example, an attacker gains access to public key storage and	**Attestation** - Secure configuration of

substitutes or adds their own verification keys, or manages to manipulate security policies on the device to a less secure state. E.g. Attacker compromises the 'take ownership' phase during device initial provisioning or later.	ownership, and signing the table of verification keys by the owner. Attestation includes measurement of security-relevant configuration data.
Misconfigured system. For example, security features not enabled, or system being in debug mode, leading to false trust. This can be either due to user-error, or done by a malicious actor like described in the previous threat.	**Attestation** - measurement of security-relevant configuration data.
FW update compromised. For example, a firmware update server is compromised, and provides malicious firmware update packages to the system, or a device itself is compromised in a way that prevents new updates from being accepted (making the current rogue firmware unpatchable).	**Secure Boot** - any install-time validation that is missed will be caught at boot-time. **Secure Update** - all firmware updates are verified and applied by the existing trusted firmware or by a dedicated RoT, rather than the downloaded firmware installing itself. **Secure Recovery** - fallback to a recovery image.
Unpatched known vulnerabilities. For example, a device has a known CVE (publically known vulnerability) which can be used by attackers to compromise the platform, the device, or a subcomponent in either.	**Post-Release patching process.** Vendors must follow a process to find, disclose and patch vulnerabilities, such as the post-release processes described in the **Secure Firmware Development best practices** document. Firmware patches can be deployed using the **Secure Update** feature.
FW/SW Bug(s). For example poor quality firmware with exploitable run-time vulnerabilities.	Minimize the risk by having all FW follow the recommendations in the "**Secure Firmware**

	Development Best Practices" Document
Insecure development environment. For example, "back doors" and/or unauthorized source code modifications get signed by the manufacturer.	Follow the recommendations in the "**Secure Firmware Development Best Practices**" Document
Compromised implementation of the security mechanism itself (the RoT). For example, someone finds a vulnerability in Secure Update that enables flashing / loading malicious firmware, or someone finds a way to predict keys being generated because of a poor implementation and/or weak source of entropy in the RoT's key generation subsystem.	RoT implementers must follow the recommendations in the "**Secure Firmware Development Best Practices**" Document **Secure Updates** - Ability to patch the security mechanism itself.
Use of unauthorized device resets to admit a malicious device into the platform after attestation. For example, a compromised device self-resets after successfully attesting, without going through re-attestation, and thus still has access even after firmware or configuration changes.	**Attestation** - devices must re-attest after every reset / FW update. **Note**: Platform design guidelines for reset-attacks resilience are currently out-of-scope, listed in the relevant section.

In-Scope Threats - Attackers with Physical Access

The following table lists in-scope threats that assumes the attacker has physical access to the protected device:

Threat	OCP Mitigation(s)
Attack using included interfaces/connectors/ports (without PCB modification). For example, a "drive-by" attack where someone inserts a specially crafted USB device to "infect" the system, or a specially crafted PCIe device into a system, which later uses DMA to steal data, or a jumper/switch that resets critical security parameters.	**Platform Requirements Spec** - System-design requirements, Power-flow controls, recommendations for disabling unnecessary interfaces.

Firmware modification via physical presence. For example, an attacker exploits physical presence, connects a SPI hook onto the Flash device, and replaces firmware image(s) with malicious one(s).	**SecureBoot & Attestation** - detect compromise during next boot.
Corrupting both the active and backup regions of the firmware. For example, an attacker uses a physical exploit (or a bug in the Secure Update mechanism, covered in the previous section) to completely overwrite the Flash.	**Secure Recovery** - fallback to an image that is not physically on the same Flash. (Either a backup flash, or an immutable recovery image)
Tampering with and/or secrets leakage from the RoT chip itself. For example, an attacker that tries to replace the RoT with a fake look alike, or modify it to behave maliciously, or read secret(s) of the RoT chip.	**Platform Requirement Spec** - RoT Tamper resistance requirements.
Spoofing a malicious system. For example, an attacker connects his own server that mimics legit hardware, or connects a rogue peripheral device to a legit system via a valid interface.	**Attestation** - Unique unclonable ID per PA RoT (platform) and per AC RoT (attestor device) to enable known inventory and rejecting unrecognized devices.

Explicit Assumptions

Some threats in the OCP security threat model are addressed by making some explicit assumptions about the environment or about the configuration of the systems being protected. The following table lists those assumptions.

Threat	Explicit Assumptions
Physical hardware tampering of the platform. For example, an attacker that physically modifies the PCB, or replaces a component with a look alike (but fake and malicious) component. Note that physical tampering of the RoT itself is in-scope and covered in the previous section.	Physical security (limited physical access to the target platform) is assumed.
Provisioning time attacks against the RoT. For example, an RoT is provisioned with wrong firmware or wrong identity at	Secured manufacturing facility is assumed during

manufacturing time, which is then used to target the platform owner.	initial identity provisioning. **Note**: Full end-to-end supply chain & secure manufacturing is a broad topic, out of scope for current effort, and listed as such in the relevant section.
Attacks via approved unauthenticated devices. For example, a device that can't attest is being admitted into the system by policy.	Such permissive attestation policies should be avoided.
DoS of the attestation bus and/or the TPM interface bus. For example, an attacker manages to inject some malicious hardware which prevents measurements to be stored in the TPM, or prevents the PA RoT to communicate with and attest a peripheral.	Functional buses are assumed, thanks to physical security and some level of trust in the supply chain.

Out-of-Scope Threats

The following table lists some examples of threats that are considered out of scope for this round of the OCP-Security specs. It is not meant to be comprehensive. It lists some threats that were seen as adjacent / similar to the threats that are addressed by OCP-security, but were different enough to be left out of scope in version 1.0 of the specs.

Having threats out-of-scope does not mean they are not important! These threats could be brought in-scope in some subsequent version, after the group had time to define how to deal with these threats.

The right column calls out some existing practices and alternative mitigations that are commonly used to address such threats. The alternative mitigations are also not meant as a comprehensive list, but are rather used as examples to clarify possible approaches.

Threat	**Possible Relevant Mitigations**
End-to-end supply chain attacks and secure manufacturing. While the OCP-security specs help deal with some aspects of the supply chain risk by having a secure identity and firmware for every device, they do not cover the full breadth of the topic**.** For example,	Potential for future work and/or collaboration with other industry bodies such as TCG.

how to deal with insiders attacks in a secure facility, what exactly does it mean to trust the supply chain, where are the boundaries, what audit processes are needed, etc.	
Run time attacks (that do not involve persisting the attack into the firmware flash). For example, exploiting a run-time vulnerability in firmware and/or applications to gain control over the system.	Software for detecting security breach at run time (such as anti-malware, breach detection & response agents, etc.
Leakage of secrets off a system, via unlimited physical access. For example, an attacker accesses decommissioned/repurposed hardware to get data that was left on it.	Data encryption & secure erase of data, secure scrap procedures for on-chip keys, minimization of mutable state in the system design.
Compromised secrets leakage from a running system. For example, an attacker uses a platform reset attack (especially on platforms that enable support of "warm reset") in order to read residual data immediately after reset from DRAM. *(Note: not to be confused with secrets stored on the RoT itself, which are definitely in-scope to protect)*	Encrypt system memory, store secrets on a TPM on the protected device, follow recommended measures such as this one against warm reset, etc.
DoS a peripheral by bombarding it with attestation requests. For example, a compromised malicious platform sends frequent attestation requests to a critical NIC interface, preventing it from being used for out of band communications by the BMC.	Peripheral designers should take that into account, but OCP is not specing how to prevent or deal with such DoS attacks.
HW Bugs. For example, some design issue in the HW itself, that is exploitable at run-time and can be used to bypass the security of the system.	Potential for future work on secure HW design guidelines, to complement the OCP firmware security best practices document.
Resilience to reset attacks due to bad platform design. For example, platforms or peripherals that allow reset to happen without clearing measurements and re-attesting all peripherals.	Potential for future work and guidance for secure platform design.
Bus Attacks. An attacker may maliciously modify traffic between security critical components on the PCB[1] to interfere with the	Potential for future work. System design must not explicitly trust the traffic on

[1] Examples bus interposer attacks: https://github.com/nccgroup/TPMGenie
https://conference.hitb.org/hitbsecconf2019ams/materials/D1T1%20-%20Toctou%20Attacks%20Against%20Secure%20Boot%20-%20Trammell%20Hudson%20&%20Peter%20Bosch.pdf

correct functioning of the system, or to exploit code injection vulnerabilities.	easily-attacked data paths.

Referenced OCP specs

This document maps threats to the various OCP specs. Links to those specs are included here for completeness:

1. **Secure Boot** - https://www.opencompute.org/documents/secure-boot-pdf
2. **Attestation** - https://www.opencompute.org/documents/attestation-v1-0-20201104-pdf
3. **Secure Update & Recovery** (Work in progress)
    a. Update - https://docs.google.com/document/d/1Tea1Nfg9T5R7O-pVtorGhQ0UHQzCdMBMckT2hJfBKB8
    b. Recovery - https://docs.google.com/document/d/1lCUGMOTLtIUrvgkCdwQgz570-ZEKSXimsagFPjBYCYc/edit#heading=h.slco0rjeyu1e
4. **Platform Requirements** (Work in progress)
    a. Top level Platform Notes - https://docs.google.com/document/d/1PPNjE3Sp05Zv9N0Gy4vaEyyCbXiFumCRfv-FUwPOsFQ
    b. Platform requirements topic - https://docs.google.com/document/d/1WYjhj79NS-TiAeRuHG1-Vn_u0itZcwxiFHkBiSHuLVU
    c. Tamper resistence topic - https://docs.google.com/document/d/1qa4B3A3fy6Jm-APwvuSZU8jnNnELO0Dv8QbwTjMXg9g/edit
5. **Secure Firmware Development Best Practices** - https://github.com/opencomputeproject/Security/blob/master/SecureFirmwareDevelopmentBestPractices.md

Appendix A: OCP Badge Threats Checklist

As part of being approved for OCP compliance and security badges, vendors are required to provide collateral that explains how their product meets the OCP security requirements and addresses the various threats. This section should be used as a template for that collateral.

Please fill the following information and submit as part of requesting an OCP badge. Please keep it brief (a paragraph or two per each question) and self-contained. This document should be readable even without reading any external documents. You may refer to external support documentation for additional support, but not instead of filling the relevant sections.

We welcome feedback on this form and submission process, and strive to improve from year to year. Please feel free to also pass your feedback to the OCP-Security team!

About the product

1. Describe the product you're requesting an OCP badge for, and which OCP categories of products does it map to:

2. At high level, which of the OCP specs does the product comply with?

   Please check all that applies, and if the product complies only partially, list the notable exceptions.

OCP Spec	**Partial or Full?** (if partial, please list here the requirements that are **not** met)
❑ Secure Boot	
❑ Attestation - as a peripheral/attester	
❑ Attestation - as a platform/verifier	
❑ Secure Firmware Updates & Recovery	
❑ Platform Requirements	

Security Threat Model and Assumptions

For each threat described in this document, briefly explain what features & mitigations the product has incorporated in order to support protection against them, and any notable exceptions if the threat is not fully covered.

Logical-Access Attacks

3. Describe how the product mitigates the risk of execution of unauthorized firmware:

4. Describe how the product mitigates the risk of execution of authorized firmware containing known security defect(s):

5. Describe how the product mitigates the risk of Compromised private signing keys:

6. Describe how the product mitigates the risk of Compromised device verification keys or hash tables:

7. Describe how the product mitigates the risk of Misconfigured system(s):

8. Describe how the product mitigates the risk of a FW update compromise:

9. Describe the vendor's process for finding and mitigating known vulnerabilities (CVE's):

10. Describe how the product mitigates the risk of having FW/SW Bug(s)

11. Describe how the product mitigates the risk of Insecure development environment:

12. Describe how the product mitigates the risk of a Compromised implementation of the RoT:

13. Describe how the product mitigates the risk of Use of unauthorized device resets:

Physical Access Attacks

14. Describe how the product mitigates the risk of attack using included interfaces:

15. Describe how the product mitigates the risk of firmware modification via physical presence:

16. Describe how the product mitigates the risk of corrupting both the active and backup regions of the firmware:

17. Describe how the product mitigates the risk of tampering and/or secrets leakage from the RoT:

18. Describe how the product mitigates the risk of spoofing a malicious system:

Explicit Assumptions

OCP specs are making some explicit assumptions about the following threats. Please provide any information on how your device adheres to these assumptions, or if it does not, what's your alternative mitigations.

19. How does the product deal with the risk of physical hardware tampering of the platform?

20. Describe how your provisioning process mitigates provisioning time attacks:

21. Describe how the product addresses the risk of attacks via approved unauthenticated devices:

22. Does the product have any special treatment for preventing DoS of the attestation bus?

Additional In-Scope Threats & Mitigations

23. **Optional:** Please list any additional security mitigations not covered by the OCP specs, that your device implements, and the threat(s) they are designed to address:

Additional Out-of-Scope Threats

24. **Optional:** Please list any additional threats, beyond what's covered in OCP's out-of-scope threats list, which are out of scope for your product:
