Trusted Platform Module in Windows 11 – Discussion Questions

Prepared By: Brandon Ma (30290589), Huma Naiman (30303242), Arunima Negi (30281210), Karanveer Singh (30282286)

Questions:

1. Considering the 18-year span from TPM 1.2's release around 2003 to TPM 2.0's finalization in 2019, what factors likely influenced the relatively slow pace of TPM standard evolution, and how did TCG balance the competing demands of maintaining backward compatibility, addressing emerging cryptographic vulnerabilities, and responding to evolving enterprise security requirements throughout this period?

2. Suppose you are tasked with selecting a TPM implementation to protect the cryptographic operations of a water treatment plant for a large city with a population of 1 million people. Which TPM implementation would you select? Explain some of the considerations for why your choice.

3. What is the difference between Secure Boot and Measured Boot, and how does a TPM support Measured Boot through its extend operation in PCRs

Answers:

1. The relatively slow pace of the TPM standard's evolution from TPM 1.2 to TPM 2.0 reflects a balanced approach addressing several competing demands.

   One critical factor was backward compatibility. While TPM 1.2 maintained continuity within its ecosystem, TPM 2.0 is not backwards compatible due to fundamental architectural changes. TPM 2.0 introduced multiple cryptographic algorithms (ECC, SHA-256, AES), more flexible authorization models, and support for multiple hierarchies, requiring a re-architected specification incompatible with TPM 1.2 software and firmware interfaces. TCG purposefully decided against backward compatibility to enable significant security and functionality improvements impossible within TPM 1.2 limitations. Maintaining compatibility would have constrained cryptographic agility and cutoff innovations needed to meet modern security challenges, so TCG opted for a clean break, accepting the migration burden in exchange for a modernized standard.

   Other factors included the rigorous standardization process involving collaboration among hardware manufacturers, software providers, and governments, which ensured broad interoperability but lengthened development time. In parallel, the growing body of cryptographic research exposed vulnerabilities in older algorithms requiring TPM 2.0's

crypto agility to support newer, stronger algorithms securely. Additionally, evolving enterprise security requirements demanded features like remote attestation, policy-based authorization, and firmware TPM implementations beyond TPM 1.2's scope.

TCG balanced these demands by introducing new capabilities while supporting migration paths, offering flexible implementation options (discrete chips, firmware TPM, etc.), and primarily targeting futureproofing over legacy compatibility. This approach ensured TPM remained a robust hardware root of trust foundation amid rapidly evolving security landscapes.

2. The first aspect to consider when deciding on the implementation of TPM to use is to think about the application of the computer system that will use it. In this case, we are tasked with ensuring that the computing platform of a water treatment plant for a large city is safe and secure. Water treatment is considered a piece of critical infrastructure, but it is especially important when it needs to serve over a million people. Therefore, we should select a TPM with the highest possible security level, which is a discrete TPM. Other important considerations are that critical infrastructure is not only important to those it serves, but it is a high value target for attackers. In this context, security is paramount and budgets are large enough to absorb the higher cost of a discrete TPM. Finally, the use case itself is what discrete TPM's are designed for, protecting critical systems.

3. Secure Boot prevents untrusted code from running by verifying digital signatures before execution, while Measured Boot records what actually ran during startup. Secure Boot enforces a "verify then execute" model, whereas Measured Boot uses "measure then report" model. The TPM supports Measured Boot by storing cryptographic hashes of each boot component in its PCRs.

   The TPM's extend operation updates a PCR as:

   New_PCR = HASH(Current_PCR || New_Measurement)

   This chaining makes PCR values reflect the exact boot sequence, enabling later attestation to prove whether the system booted in a trusted state.