Example of Redacted RoC

This is to demonstrate what the CCSSA-PR would be reviewing for the peer review. Note the comments in the *CCSSA Findings* and *Evidence Gathered* sections for a requirement section within the CCSS RoC.

LEVEL I REQUIREMENTS					
Requirement	1.01.1.1 The cryptographic keys and seeds are created by the actor who will be using it.				
Audit Finding Summary	**In-Place**	**In-Place with Comparable Control**	**Qualified for In-Place**	**Not In-Place**	**Not Applicable**
	☐	☐	☐	☐	☒
CCSSA Findings	ACME key generation mechanisms are executed during account setup by code executed within the MPE secure enclave as part of the customer's account setup. The MPC key consists of three MPC key shares. Two of the key shares are stored within the ACME environment and are used by automated signing agents. The third key share is installed on the customers device (usually a mobile device) within the Trusted Execution Environment of that device (TEE) and used by the ACME automated signing agent installed on the customers same device.				
Evidence Gathered	*For each of the testing types below, provide information on the evidence gathered. If evidence of the type has not been gathered, indicate that next to the evidence type.*				

	Interviews	Interviews with [INTERVIEW_1] who confirmed that the key management functions such as generation of MPC key shares are completely automated.
	Documents	Key Management --- DOCUMENT_15 DOCUMENT_17 DOCUMENT_19 DOCUMENT_20 DOCUMENT_40 DOCUMENT_47 DOCUMENT_51 DOCUMENT_52
Requirement	1.01.1.2 In cases where an automated agent will make use of a cryptographic key/seed, it is required that the administrator of that system generate the key/seed on a suitable offline system with sufficient entropy, have this key/seed transferred securely onto the target device, and then securely deleted using CCSS-compliant data sanitization techniques to protect the confidentiality of the key/seed.	

Audit Finding Summary	**In-Place**	**In-Place with Comparable Control**	**Qualified for In-Place**	**Not In-Place**	**Not Applicable**

	☒	☐	☐	☐	☐
CCSSA Findings	ACME key generation mechanisms are executed during account setup by Trusted code executed within the MPE secure enclave as part of the customer's account setup. The MPE secure enclave provides a secure execution environment within the device's random-access memory. MPE is a set of CPU instruction codes that are hard coded into selected CPU chips and provide security functions. The MPE functions can only be called by "Trusted" code which is code signed by an entity's signing key. The signed Trusted code is executed within the designated secure enclave. The MPE functions work independently of the devices core functions (non-MPE CPU code, devices operating system, BIOS and application software). Each MPC key share is created individually on 3 different devices. The two MPC key shares created and managed by ACME are created and reside within the MPE secure enclave on the same device in which the MPC key share was created. The CCSSA reviewed the code audit reports that included the review of the MPE code functions and confirmed that MPC key shares are created, used, and stored on the same MPE secure enclave. The entropy used for the creation of the MPC key share is sourced from the TOPO interface provided on the same devices as the MPE secure enclave. For further information regarding the entropy refer to *CCSSA Findings* in requirement 1.01.3.1. The intent of this requirement is to ensure the secure transmission of a key from the place of creation to the place of execution and storage. ACME' use of the MPE chipset instructions provides a secure environment that is independent of the device's environment on which the MPE mechanism resides. There is no transmission of the MPC key outside of the MPE environment on the device during the creation of the ACME owned and managed MPC key shares.				
Evidence Gathered	*For each of the testing types below, provide information on the evidence gathered. If evidence of the type has not been gathered, indicate that next to the evidence type.*				

	Interviews	Interviews with [INTERVIEW_1] who confirmed that the architecture of the key generation mechanisms are only executed within the MPE environment for the ACME devices.
	Documents	Key Management --- DOCUMENT_15 DOCUMENT_17 DOCUMENT_19 DOCUMENT_20 DOCUMENT_40 DOCUMENT_47 DOCUMENT_51 DOCUMENT_52 Code Audits --- DOCUMENT_11 DOCUMENT_12 DOCUMENT_18 DOCUMENT_44

		MPE Official Documentation --- DOCUMENT_55

LEVEL II REQUIREMENTS					
Requirement	1.01.1.3 A digital signature for the key creation software is generated, published, and validated prior to each execution.				
Audit Finding Summary	**In-Place**	**In-Place with Comparable Control**	**Qualified for In-Place**	**Not In-Place**	**Not Applicable**
	☐	☒	☐	☐	☐
CCSSA Findings	The key creation software, before deployment, is signed with the signing key of either the ACME Head of DevOps or the ACME CISO. If the key creation software is not signed with the signing key of one of the authorized ACME employees, then the key creation code cannot execute which is enforced by the MPE mechanism. The SDLC and deployment processes execute several checks to ensure the code has not been tampered with during the development process. The deployment of code can only be completed with approval from the ACME CISO. Any change to code which contains cryptographic functions such as key generation functions is first audited by a third-party auditor skilled and qualified to conduct code audits and cryptographic functions before the code is deployed to production.				

Evidence Gathered	*For each of the testing types below, provide information on the evidence gathered. If evidence of the type has not been gathered, indicate that next to the evidence type.*	
	CCSS Committee Decisions	[CCSS_DECISION_4] Agreement from member of CCSS Committee that ACME SOC2 Type2 report findings can be submitted as evidence for this CCSS audit.
	Interviews	Interviews with [INTERVIEW_1] who confirmed that the key generation code (and all code that performs cryptographic functions) must be signed with the signing key of either the ACME Head of DevOps or the ACME CISO. Interviews with [INTERVIEW_7] who confirmed: 1. The only ACME employees who are authorized to sign the code are either the ACME Head of DevOps or the ACME CISO. 2. A third-party audit of any code containing cryptographic functions is required before deployment. Interviews with [INTERVIEW_5] who confirmed that an audit is performed annually on development and deployment processes to ensure a third-party audit of the code is performed when a change is made to any code containing cryptographic functions.

	Documents	Key Management --- DOCUMENT_15 DOCUMENT_17 DOCUMENT_19 DOCUMENT_20 DOCUMENT_40 DOCUMENT_47 DOCUMENT_51 DOCUMENT_52 Code Audits --- DOCUMENT_11 DOCUMENT_12 DOCUMENT_18 DOCUMENT_44 Penetration Testing Reports --- DOCUMENT_21

		DOCUMENT_22 DOCUMENT_23 Change Management --- DOCUMENT_3 SOC2 TYPE 2 Audit Findings Report SOC2 TYPE 2 Bridge Letter --- DOCUMENT_6 DOCUMENT_13 Access Management --- DOCUMENT_45 GRC Audit Frequency Schedule --- DOCUMENT_50

		SDLC Processes ---------- DOCUMENT_2 DOCUMENT_43 DOCUMENT_54
LEVEL III REQUIREMENTS		
No Level III Requirements.		