RASUES Security analysis

Sniffing

	Where	Threat	Consequences	CIA	Attacker	Property	Mitigation
Phase 1							
1	Attester ←all→ Relying party	Attacker may attempt to **listen** to the communication.	The **communication** authenticity is **broken among all entities**	Confidentiality	A_N(Netowrk)	F^S_4 (Integrity Assurance)	end-to-end encryption (TLS)
2	Attester ← Relying party	Attacker may attempt to **listen** to the communication.	Learning when the AWDT timer is **initialized**	Confidentiality	A_N(Netowrk)	F^S_4 (Integrity Assurance) F^S_2 (Recoverability)	end-to-end encryption (TLS) + cryptographic key
3	Target environment → Attestation environment	Attacker may attempt to **listen** to the communication.	Learning when the AWDT timer is **invoked**	Confidentiality	A_N(Netowrk)	F^S_4 (Integrity Assurance), F^S_2 (Recoverability)	end-to-end encryption (TLS) + TPM2 with timestamp in attestation environment
4	RVP ← Relying party	Attacker may attempt to **listen** to the communication.	Learning when the AWDT timer is **started**	Confidentiality	A_N(Netowrk)	F^S_4 (Integrity Assurance)	end-to-end encryption (TLS)
5	Verifier ← RVP	Attacker may attempt to **listen** to the communication.	Learning the **derived expected state in the whitelist**	Confidentiality	A_N(Netowrk)	F^S_4 (Integrity Assurance), F^S_1 (Updateability)	end-to-end encryption (TLS) + watchdog timer
6	Attester ← RVP	Attacker may attempt to **listen** to the communication.	Learning the **content of the updated transmission**	Confidentiality	A_N(Netowrk)	F^S_4 (Integrity Assurance), F^S_1 (Updateability)	end-to-end encryption (TLS) + watchdog timer
7	Target environment → Attestation environment **(2)**	Attacker may attempt to **listen** to the communication.	Learning the content of the update **during the update**	Confidentiality	A_N(Netowrk)	F^S_4 (Integrity Assurance) F^S_1 (Updateability)	end-to-end encryption (TLS) + TPM2
Phase 2							

8	Attester ← Verifier	Attacker may attempt to **listen** to the communication.	Learning the **content of the attestation request**	Confidentiality	A_N(Netowrk)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	end-to-end encryption (TLS) + nonce generation
9	Target environment → Attestation environment **(3)**	Attacker may attempt to **listen** to the communication.	Learning the **integrity measurement process**	Confidentiality	A_N(Netowrk)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	end-to-end encryption (TLS) + IMA, TPM2 + Nonce
10	Target environment ← Attestation environment	Attacker may attempt to **listen** to the communication.	Learning the **integrity measurement is completed**	Confidentiality	A_N(Netowrk)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	end-to-end encryption (TLS) + IMA + cryptographic key
11	Attester → Vérifier	Attacker may attempt to **listen** to the communication.	Learning the **evidence through the attestation response, thus results in reducing its trustworthiness**	Confidentiality	A_N(Netowrk)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	end-to-end encryption (TLS) + IMA + cryptographic key +nonce
Phase 3							
12	Attester ← Verifier **(2)**	Attacker may attempt to **listen** to the communication.	Learning the **attestation result**	Confidentiality	A_N(Netowrk)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	end-to-end encryption (TLS) + IMA + cryptographic key +nonce
13	Attester → Relying party	Attacker may attempt to **listen** to the communication.	Learning the **attestation result**	Confidentiality	A_N(Netowrk)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	end-to-end encryption (TLS) + nonce
14	Attester ← Relying party **(2)**	Attacker may attempt to **listen** to the communication.	Learning the **decision made by the relying party and knowing the deferral tickets**	Confidentiality	A_N(Netowrk)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	end-to-end encryption (TLS) + watchdog timer +nonce
15	Target environment → Attestation environment **(4)**	Attacker may attempt to **listen** to the communication.	Learning the **decision made by the relying party and knowing the deferral tickets**	Confidentiality	A_N(Netowrk)	F^S_4 (Integrity Assurance), F^S_2 (Recoverability)	end-to-end encryption (TLS) + watchdog timer
Blocking							

	Where	Threat	Consequences	CIA	Attacker	Property	Mitigation
Phase 1							
1	Attester ←all→ Relying party	Attacker may attempt to **block access** to the entity.	Preventing entities from communicating with each other	Availability	A_N(Netowrk)	F^S_4 (Integrity Assurance)	TLS can detect violations
2	Attester ← Relying party	Attacker may attempt to **block access** to the entity.	Blocking the **instruction of setting up the AWDT**	Availability	A_N(Netowrk) A_S(Software)	F^S_4 (Integrity Assurance) F^S_2 (Recoverability)	We assumed to be trusted
3	Target environment → Attestation environment	Attacker may attempt to **block access** to the entity.	Blocking **invoking AWDT**	Availability	A_N(Netowrk)	F^S_4 (Integrity Assurance), F^S_2 (Recoverability)	TPM2 in the attestation environment
4	RVP ← Relying party	Attacker may attempt to **block access** to the entity.	Blocking **starting AWDT**	Availability	A_N(Netowrk)	F^S_4 (Integrity Assurance)	end-to-end encryption (TLS)
5	Verifier ← RVP	Attacker may attempt to **block access** to the entity.	Blocking **deriving the expected state from the whitelist**	Availability	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance), F^S_1 (Updateability)	end-to-end encryption (TLS) + IMA
6	Attester ← RVP	Attacker may attempt to **block access** to the entity.	The update will not be delivered due to the provider RVP being blocked. Thus, the attester will not be updated and will remain in its current state, either vulnerable or benign	Availability	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance), F^S_1 (Updateability)	RVP is assumed to be trusted Since the timer has been initiated recently, an attack will be detected when the watchdog has timed out
7	Target environment → Attestation environment **(2)**	Attacker may attempt to **block access** to the entity.	Blocking triggering update at the measurement.	Availability	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance) F^S_1 (Updateability)	TPM2 in the attestation environment
Phase 2							
8	Attester ← Verifier	Attacker may attempt to **block access** to the entity.	Verifier not capable of initiating attestation request, thus cannot determine the trustworthiness of attester's expected state	Availability	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	end-to-end encryption (TLS), IMA, cryptographic key exchange, nonce

9	Target environment → Attestation environment **(3)**	Attacker may attempt to **block access** to the entity.	The integrity measurement will not be triggered in the attestation environment	Availability	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	IMA, cryptographic key exchange, TPM2, nonce
10	Target environment ← Attestation environment	Attacker may attempt to **block access** to the entity.	The attestation environment will not provide the evidence of the measurement	Availability	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	It will fail to start the measurement. It will wait until the watchdog timer has rolled out, IMA, and key exchange, nonce
11	Attester → Vérifier	Attacker may attempt to **block access** to the entity.	The attester is unavailable to comply with the required measurement and thus cannot continue with verification	Availability	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	Attestation will time out; thus, it will not be verified It will wait until the watchdog timer has rolled out. IMA and key exchange, nonce
Phase 3							
12	Attester ← Verifier **(2)**	Attacker may attempt to **block access** to the entity.	The effect is minimal, but the verifier cannot send the verified evidence to the attester	Availability	A_N(Netowrk),A_S (Software)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	IMA and key exchange, nonce
13	Attester → Relying party	Attacker may attempt to **block access** to the entity.	The Appraisal process will fail due to the relying party not receiving the attestation result. Thus, the attester might be malicious or vulnerable The device will not be able to receive any further updates and will become completely unavailable for updates or further assessments, or verification	Availability	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	Out of scope. The device will be rest, however, if the device is always being rest due to an attack, this leads to an endless loop, thus cannot be mitigated, since this is a type of DoS, nonce

14	Attester ← Relying party **(2)**	Attacker may attempt to **block access** to the entity.	Attester cannot retrieve the Deferral Ticket, indication of an attack. Denying access to the attester, thus the relying party cannot make a decision, leaving relying party unaware of the current status of the attester even though its expected state has been verified	Availability	A_N(Netowrk),A_S (Software)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	Watchdog timer, nonce
15	Target environment → Attestation environment **(4)**	Attacker may attempt to listen to the communication.		Availability	A_N(Netowrk),A_S (Software)	F^S_4 (Integrity Assurance), F^S_2 (Recoverability)	Watchdog timer

Modification

	Where	Threat	Consequences	CIA	Attacker	Property	Mitigation
Phase 1							
1	Attester ←all→ Relying party	Attacker may attempt to **modify the content** of all the entities.	Leads to the impersonation of another malicious device	Integrity	A_N(Netowrk), A_P(Priviledged)	F^S_4 (Integrity Assurance)	end-to-end encryption (TLS)
2	Attester ← Relying party	Attacker may attempt to **modify the content** of the entity.	Attacker modifying the AWDT timer. Preventing the attester from resetting if it is malicious	Integrity	A_N(Netowrk), A_S(Software), A_P(Priviledged)	F^S_4 (Integrity Assurance), F^S_2 (Recoverability)	The relying party is using the public cryptographic key of the attester to communicate with the attester in the attestation environment
3	Target environment → Attestation environment	Attacker may attempt to **modify the content** of the entity.	Modifying the invoke request, so the attestation environment might be invoked when it is not supposed to be	Integrity	A_N(Netowrk), A_S(Software), A_P(Priviledged)	F^S_4 (Integrity Assurance), F^S_2 (Recoverability)	Cryptographic key exchange, TPM2 end-to-end encryption (TLS)

4	RVP ← Relying party	Attacker may attempt to **modify the content** of the entity.	Modifying the content of the AWDT so that the attacker misconfigures the initialization time	Integrity	A_N(Netowrk), A_S(Software), A_P(Priviledged)	F^S_4 (Integrity Assurance)	Cryptographic key exchange, AWDT, end-to-end encryption (TLS)
5	Verifier ← RVP	Attacker may attempt to **modify the content** of the entity.	Modifying the whitelist	Integrity	A_N(Netowrk), A_S(Software), A_P(Priviledged)	F^S_4 (Integrity Assurance), F^S_1 (Updateability)	end-to-end encryption (TLS), AWDT, IMA
6	Attester ← RVP	Attacker may attempt to **modify the content** of the entity.	Modifying the supplied update before installation	Integrity	A_N(Netowrk), A_S(Software), A_P(Priviledged)	F^S_4 (Integrity Assurance), F^S_1 (Updateability)	end-to-end encryption (TLS), TPM2
7	Target environment → Attestation environment **(2)**	Attacker may attempt to **modify the content** of the entity.	Modifying how the attestation environment or triggering measurement by an unintended entity	Integrity	A_N(Netowrk), A_S(Software), A_P(Priviledged)	F^S_4 (Integrity Assurance), F^S_1 (Updateability)	There is a TPM in the attestation environment and TPM
Phase 2							
8	Attester ← Verifier	Attacker may attempt to **modify the content** of the entity.	Invoking untrusted or modified attestation requests. Unnecessary attestation request challenge	Integrity	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	end-to-end encryption (TLS), IMA, TPM2, nonce
9	Target environment → Attestation environment **(3)**	Attacker may attempt to **modify the content** of the entity.	Modifying when triggering the attestation measurement	Integrity	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	Cryptographic key exchange, TPM2, IMA
10	Target environment ← Attestation environment	Attacker may attempt to **modify the content** of the entity.	Modifying the integrity measurement of attestation	Integrity	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	Cryptographic key exchange, TPM2, IMA
11	Attester → Verifier	Attacker may attempt to **modify the content** of the entity.	Modifying the evidence. Verifying wrong evidence or wrong Device Resulting in an attestation to be legitimate	Integrity	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	Cryptographic key exchange, TPM2, IMA

			Note: what is included in the evidence, such as the device ID. Device ID, Nonce, public/private keys, and PCR.				
Phase 3							
12	Attester ← Verifier **(2)**	Attacker may attempt to **modify the content** of the entity.	Modifying the attestation results provided by the verifier. This breaks the integrity of RATS passport-model.	Integrity	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	Cryptographic key exchange, TPM2
13	Attester → Relying party	Attacker may attempt to **modify the content** of the entity.	Attacker modifying the evidence and the results to be sent to the relying party. The replying party might be fooled to accept the given evidence, thus making a wrong decision	Integrity	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	The evidence is signed with the verifier's cryptographic key during attestation. TPM2
14	Attester ← Relying party **(2)**	Attacker may attempt to **modify the content** of the entity.	Modifying the decision made by the relying party and changing the deferral tickets of AWDT. Preventing it from resetting on time	Integrity	A_N(Netowrk), A_S(Software), A_P(Priviledged)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	end-to-end encryption (TLS)+ watchdog timer
15	Target environment → Attestation environment **(4)**	Attacker may attempt to **modify the content** of the entity.	Modifying the decision and AWDT by the attester	Integrity	A_N(Netowrk), A_S(Software), A_P(Priviledged)	F^S_4 (Integrity Assurance), F^S_2 (Recoverability)	end-to-end encryption (TLS) + watchdog timer

Replay/Spoof

	Where	Threat	Consequences	CIA	Attacker	Property	Mitigation
Phase 1							

1	Attester ←all→ Relying party	Attacker may attempt to **replay or spoof** the communication	Untrusted entities will participate in the communication, thus replay or spoof other entities	Integrity	A_N(Netowrk), A_P(Priviledged)		end-to-end encryption (TLS) communication is established with only trusted entities
2	Attester ← Relying party	Attacker may attempt to **replay or spoof** the initialization of AWDT using new or existing credentials	The attester will comply with malicious AWDT initialization, thus it will become under the attacker's control	Integrity	A_N(Netowrk), A_S(Software), A_P(Priviledged)		TLS timestamp will not allow using same time twice
3	Target environment → Attestation environment	Attetester **replay or spoof AWDT invoking process**	The attester will start with incorrect AWDT initialization, thus manipulated	Integrity	A_N(Netowrk), A_S(Software), A_P(Priviledged)		TLS will detect violation from untrusted parties
4	RVP ← Relying party	The attacker incorrectly informs the relying party about the AWDT initialization	The relying party will miscalculate the timer and become under the attacker's control	Integrity	A_N(Netowrk), A_S(Software), A_P(Priviledged)		TLS will detect violation from untrusted parties
5	Verifier ← RVP	Spoofing the expected state from the RVP	Attester can always mark itself as legitimate after remote attestation	Integrity	A_N(Netowrk), A_S(Software), A_P(Priviledged)		The relying party configure public/private key
6	Attester ← RVP	Installing a malicious update and impersonating the RVP	This led the attester to be controlled by the attacker	Integrity	A_N(Netowrk), A_S(Software), A_P(Priviledged)		We require for all update a signature from RVP
7	Target environment → Attestation environment **(2)**	Spoofing attester to trigger measurement	The attester will perform the measurement	Integrity	A_N(Netowrk), A_S(Software), A_P(Priviledged)		
Phase 2							
8	Attester ← Verifier	Attacker impersonating verifier and initializing attestation request	The attester will repeatedly comply with an untrusted verifier; thus, the verifier has unauthorized access to	Integrity	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	Nonce exchanges is done constantly to ensure freshness **(Trusted)**

			the attester’s data				
9	Target environment → Attestation environment **(3)**	Attacker impersonates the target environment to invoke the attestation environment	This causes the attestation environment to reply to untrusted	Integrity	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	end-to-end encryption (TLS), IMA, TPM, nonce
10	Target environment ← Attestation environment	Attacker may attempt to spoof or replay the attester’s contents	The integrity measurement has been compromised	Integrity	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	end-to-end encryption (TLS), IMA, TPM, nonce
11	Attester → Vérifier	Attacker may attempt to spoof or replay the attester’s contents	Using existing communication credentials in the attester or spoofing it and reply to the verifier	Integrity	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	Nonce exchanges are done constantly to ensure freshness
Phase 3							
12	Attester ← Verifier **(2)**	Attacker may attempt to spoof or replay the verifier’s contents	The passport model pattern is broken; thus, it is no longer trusted	Integrity	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	Cryptographic key exchange, TPM, nonce
13	Attester → Relying party	Attacker may attempt to spoof or replay the attester’s contents	The attester can reproduce evidence/results and send it to the relying party to be appraised	Integrity	A_N(Netowrk), A_S(Software)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	Using AWDT with activation of the Deferral Ticket, nonce
14	Attester ← Relying party **(2)**	Attacker may attempt to spoof or replay the relying party's contents	The attacker can send a wrong decision/deferral ticket to the attester. Resulting in always trusting the atester	Integrity	A_N(Netowrk), A_S(Software), A_P(Priviledged)	F^S_4 (Integrity Assurance), F^S_3 (Freshness)	Relying party assumed to be trusted. The watchdog timer has a challenge-response protocol using HMAC when setting the timer. nonce
15	Target environment → Attestation environment **(4)**	Attacker may attempt to spoof or replay the attestester’s contents		Integrity	A_N(Netowrk), A_S(Software), A_P(Priviledged)	F^S_4 (Integrity Assurance), F^S_2 (Recoverability)	end-to-end encryption (TLS) + watchdog timer

Message sequence in the RASUES protocol

	Where	Threat	Consequences	CIA		Property	Mitigation
Phase 1							
1	Attester ←all→ Relying party						
2	Attester ← Relying party						
3	Target environment → Attestation environment						
4	RVP ← Relying party						
5	Verifier ← RVP						
6	Attester ← RVP						
7	Target environment → Attestation environment **(2)**						
8	Attester ← Verifier						
9	Target environment → Attestation environment **(3)**						
10	Target environment ← Attestation environment						
11	Attester → Vérifier						

12	Attester ← Verifier **(2)**						
13	Attester → Relying party						
14	Attester ← Relying party **(2)**						
15	Target environment → Attestation environment **(4)**						