Background Check Model

Attestation Result

Verifier → TAM (Relying Party)

Evidence

TAM (Relying Party) → Teep Broker ← Teep Agent ← Attester

Evidence   Evidence

I think this draft is based on **Background Check Model**

A Usable Formal Methods Sample Problem from TEEP
https://datatracker.ietf.org/doc/draft-mt-ufmrg-teep-sample/

Passport Model

Verifier   TAM (Relying Party)

1 Evidence

3 Attestation Result

2 Attestation Result

Teep Broker ← Teep Agent ← Attester

Evidence   Evidence

**Author
(TC Signer)**

| Verifier | Internet | TAM
(Relying Party) | Internet | Teep
Broker | Teep Agent | authenticated
& confidential
channel | Attester |

event(TCSigner_query_process_start)

event(TAM_query_process_start)

event(TEEP_Agent_query_process_start)

event(TCSigner_query_process_end)

event(Verifier_process_start)

Query event(TCSigner_query_process_end) ==> event(TAM_query_process_start) is false.
Query inj-event(TCSigner_query_process_end) ==> inj-event(TAM_query_process_start) is false.
Query event(TAM_query_process_end) ==> event(TCSigner_query_process_start) is false.
Query inj-event(TAM_query_process_end) ==> inj-event(TCSigner_query_process_start) is false.

3.1 NonceRequest

3.1 NonceResponse

3.2 QueryRequest {token, challenge, ~~supported-teep-cipher-suites, supported-suit-cose-profiles~~, data-item-requested(trusted-components, attestation)} SK_TAM

event(Attester_process_start)

3.3 EvidenceRequest {challenge}

event(Attester_process_message(**evidence**))

event(TEEPAgent_query_process_evidence(**evidence**))

3.3 EvidenceResponse {challenge, claims} SK_ATTESTER

3.3 QueryResponse {token, ~~selected-teep-cipher-suite~~, attestation-payload-format(EAT), attestation-payload({challenge, claims}SK_ATTESTER), ~~tc-list~~} SK_AGENT

Query event(Attester_process_end) ==> event(TEEP_Agent_query_process_start) || event(TEEP_Agent_update_process_start) is true.
Query inj-event(Attester_process_end) ==> inj-event(TEEP_Agent_query_process_start) || inj-event(TEEP_Agent_update_process_start) is false.
Query event(TEEP_Agent_query_process_end) ==> event(Attester_process_start) is true.
Query inj-event(TEEP_Agent_query_process_end) ==> inj-event(Attester_process_start) is false.

Query event(TEEP_Agent_query_process_end) ==> event(TAM_query_process_start) is true.
Query inj-event(TEEP_Agent_query_process_end) ==> inj-event(TAM_query_process_start) is false.
Query event(TAM_query_process_end) ==> event(TEEP_Agent_query_process_start) is true.
Query inj-event(TAM_query_process_end) ==> inj-event(TEEP_Agent_query_process_start) is true.

event(TAM_query_process_evidence(**evidence**))

Evidence

event(Verifier_process_message(**evidence**))

Attestation Result

event(Attester_process_end)

Query event(Verifier_process_end) ==> event(TAM_query_process_start) || event(TAM_update_process_start) is true.
Query inj-event(Verifier_process_end) ==> inj-event(TAM_query_process_start) || inj-event(TAM_update_process_start) is true.
Query event(TAM_query_process_end) ==> event(Verifier_process_start) is true.
Query inj-event(TAM_query_process_end) ==> inj-event(Verifier_process_start) is false.

event(Verifier_process_end)

event(TAM_query_process_end)

event(TEEP_Agent_query_process_end)

Query event(Verifier_process_end) ==> event(Attester_process_start) is true.
Query inj-event(Verifier_process_end) ==> inj-event(Attester_process_start) is true.
Query event(Verifier_process_message(evidence_message_8)) ==> event(Attester_process_message(evidence_message_8)) is true.
Query inj-event(Verifier_process_message(evidence_message_8)) ==> inj-event(Attester_process_message(evidence_message_8)) is true.

Query event(Verifier_process_message(evidence_message_8)) ==> event(TAM_query_process_evidence(evidence_message_8)) || event(TAM_update_process_evidence(evidence_message_8)) is false.
Query inj-event(Verifier_process_message(evidence_message_8)) ==> inj-event(TAM_query_process_evidence(evidence_message_8)) || inj-event(TAM_update_process_evidence(evidence_message_8)) is false.
Query event(TAM_query_process_evidence(evidence_message_8)) ==> event(TEEPAgent_query_process_evidence(evidence_message_8)) is true.
Query inj-event(TAM_query_process_evidence(evidence_message_8)) ==> inj-event(TEEPAgent_query_process_evidence(evidence_message_8)) is true.
Query event(TEEPAgent_query_process_evidence(evidence_message_8)) ==> event(Attester_process_message(evidence_message_8)) is true.
Query inj-event(TEEPAgent_query_process_evidence(evidence_message_8)) ==> inj-event(Attester_process_message(evidence_message_8)) is false.

Background Check Model

Author (TC Signer) | Verifier | Internet | TAM (Relying Party) | Internet | Teep Broker | Teep Agent | authenticated & confidential channel | Attester

event(TCSigner_update_process_start)
event(TAM_update_process_start)
event(TEEP_Agent_update_process_start)

event(TCSigner_install_start(**TC**))

3.5 SoftwareUpdate

{TC_manifest, TC_seq_nr, TC} SK_AUTHOR

event(TAM_install_process(**TC**))

event(Verifier_process_start)

Query event(TCSigner_update_process_end) ==> event(TAM_update_process_start) is false.
Query inj-event(TCSigner_update_process_end) ==> inj-event(TAM_update_process_start) is false.
Query event(TAM_update_process_end) ==> event(TCSigner_update_process_start) is true.
Query inj-event(TAM_update_process_end) ==> inj-event(TCSigner_update_process_start) is false.

3.1 NonceRequest

3.6 Update   {token2, {TC_manifest, TC_seq_nr, software}SK_AUTHOR} SK_TAM

event(Attester_process_start)

3.1 NonceResponse

event(TC_successfully_installed(**TC**))   3.3 EvidenceRequest   {challenge}

event(Attester_process_message(**evidence**))

event(TEEPAgent_update_process_evidence(**evidence**))   3.3 EvidenceResponse   {challenge, claims} SK_ATTESTER

event(TAM_update_process_evidence(**evidence**))   3.7 Success   {token2} SK_AGENT

Evidence

Query event(Attester_process_end) ==> event(TEEP_Agent_query_process_start) || event(TEEP_Agent_update_process_start) is true.

Query event(TEEP_Agent_update_process_end) ==> event(TAM_update_process_start) is true.
Query inj-event(TEEP_Agent_update_process_end) ==> inj-event(TAM_update_process_start) **cannot be proved**.
Query event(TAM_update_process_end) ==> event(TEEP_Agent_update_process_start) is true.
Query inj-event(TAM_update_process_end) ==> inj-event(TEEP_Agent_update_process_start) is true.

event(Verifier_process_message(**evidence**))

Query inj-event(Attester_process_end) ==> inj-event(TEEP_Agent_query_process_start) || inj-event(TEEP_Agent_update_process_start) is false.
Query event(TEEP_Agent_update_process_end) ==> event(Attester_process_start) is true.
Query inj-event(TEEP_Agent_update_process_end) ==> inj-event(Attester_process_start) is false.

Attestation Result

Query event(Verifier_process_end) ==> event(TAM_query_process_start) || event(TAM_update_process_start) is true.
Query inj-event(Verifier_process_end) ==> inj-event(TAM_query_process_start) || inj-event(TAM_update_process_start) is true.
Query event(TAM_update_process_end) ==> event(Verifier_process_start) is true.
Query inj-event(TAM_update_process_end) ==> inj-event(Verifier_process_start) is false.

event(Verifier_process_end)   event(TCSigner_update_process_end)   event(TAM_update_process_end)   event(TEEP_Agent_update_process_end)   event(Attester_process_end)

Query event(TC_successfully_installed(TCSigner_message_3)) ==> event(TCSigner_install_start(TCSigner_message_3)) is true.
Query inj-event(TC_successfully_installed(TCSigner_message_3)) ==> inj-event(TCSigner_install_start(TCSigner_message_3)) is false.

Query event(Verifier_process_end) ==> event(Attester_process_start) is true.
Query inj-event(Verifier_process_end) ==> inj-event(Attester_process_start) is true.
Query event(Verifier_process_message(evidence_message_8)) ==> event(Attester_process_message(evidence_message_8)) is true.
Query inj-event(Verifier_process_message(evidence_message_8)) ==> inj-event(Attester_process_message(evidence_message_8)) is true.

Query event(Verifier_process_message(evidence_message_8)) ==> event(TAM_query_process_evidence(evidence_message_8)) || event(TAM_update_process_evidence(evidence_message_8)) is false.
Query inj-event(Verifier_process_message(evidence_message_8)) ==> inj-event(TAM_query_process_evidence(evidence_message_8)) || inj-event(TAM_update_process_evidence(evidence_message_8)) is false.
Query event(TAM_update_process_evidence(evidence_message_8)) ==> event(TEEPAgent_update_process_evidence(evidence_message_8)) is true.
Query inj-event(TAM_update_process_evidence(evidence_message_8)) ==> inj-event(TEEPAgent_update_process_evidence(evidence_message_8)) is true.
Query event(TEEPAgent_update_process_evidence(evidence_message_8)) ==> event(Attester_process_message(evidence_message_8)) is true.
Query inj-event(TEEPAgent_update_process_evidence(evidence_message_8)) ==> inj-event(Attester_process_message(evidence_message_8)) is false.

Summary of Security Properties

Target security properties	Target entities	Verification results	Consideration	Discussion
Authentication (between entities with explicit communication)	TC Signer ==> TAM TAM ==> TC Signer	False	can be true with SSH secure channel / out of scope in TEEP	(TC means Trusted Components.)
	Verifier ==> TAM	**True(injective)**	**replay protection with nonce**	If Verifier want to check the TAM who provides the evidence, Verifier needs extra authentication & authorization.
	TAM ==> Verifier	**True**(non-injective)		
	TAM ==> TEEP Agent	**True(injective)**	**replay protection with token**	
	TEEP Agent ==> TAM	**True**(non-injective)		
	TEEP Agent ==> Attester	**True** (non-injective)		TEEP Agent does not provide own freshness, even with challenge by TAM(= nonce by Verifier)
	Attester ==> TEEP Agent	**True**(non-injective)		
Integrity (message integrity / with implicit message exchange)	TEEP Agent(TC) ==> TC Signer(TC)	**True**(non-injective)		(TC means Trusted Components.)
	Verifier(evidence) ==> Attester(evidence)	**True(injective)**	**replay protection with nonce / challenge**	Although Verifier does know the TAM and the evidence itself, does not know which TAM provides the evidence.
Replay protection (described above, altogether)	-	-	-	-
Confidentiality (not mandatory)	Attestation Evidence	-	-	-
	Trusted Components	-	-	-