MACHINE

VARIABLES

tamper

valid_forge

INVARIANTS

inv4: $tamper = FALSE \land phase \in \{res, est, checking\} \Rightarrow p\_in = p\_out$

inv3: ⟨theorem⟩ $(tamper = FALSE \land phase = checking) \Rightarrow R\_est \leq R\_real$

inv5: $(tamper = TRUE \land phase = checking) \Rightarrow R\_real < R\_est$

inv8: $(tamper = TRUE \land valid\_forge = TRUE) \Rightarrow phase \neq detection$

inv9: $(valid\_forge = TRUE \land tamper = TRUE) \Rightarrow comp\_mac(p\_in(front) \mapsto p\_in(int) \mapsto p\_in(len)) = p\_in(mac)$

inv13: $(tamper = TRUE \land phase \in \{res, est, checking\}) \Rightarrow valid\_forge = TRUE$

EVENTS

Event Integrity ⟨ordinary⟩ ≙

refines Integrity

where

grd10: $tamper = FALSE \Rightarrow rec2 = p\_out$

grd9: $tamper = TRUE \Rightarrow rec2 = \{front \mapsto F\_real + 1, int \mapsto 2, len \mapsto F\_real - R\_real, mac \mapsto mac\_forge\}$

then

act3: $valid\_forge :| (tamper = FALSE \Rightarrow valid\_forge' = FALSE) \land (tamper = TRUE \Rightarrow valid\_forge' = bool(mac\_forge = comp\_mac(F\_real + 1 \mapsto 2 \mapsto F\_real - R\_real)))$

end

Event Tamper ⟨ordinary⟩ ≙

when

grd1: $phase = integrity$

grd2: $F\_real \neq l$

then

act1: $tamper := TRUE$

end

END

2