\sim M 3 = fl((senc((SUPI,pk(skUE_2),SNname),KEMkey(Encaps(pk(skHN),rUE_2))),KEMkey(Encaps(pk(skHN),rUE_2)))) \sim M 4 = idHN 3 \sim X 1 = (\sim M 1, \sim M 2, \sim M 3,a 1) = (KEMCipher(Encaps(pk(skHN), rUE_2)),senc((SUPI,pk(skUE 2),SNname),KEMkey(Encaps(pk(skHN),rUE 2))),fl((senc((SUPI,pk(skUE 2),SNname), KEMkey(Encaps(pk(skHN),rUE 2))),KEMkey(Encaps(pk(skHN),rUE 2)))),a 1) \sim X_2 = (ssID_3,KEMCipher(Encaps(pk(skHN),rUE_2)),senc((SUPI,pk(skUE 2),SNname),KEMkey(Encaps(pk(skHN), rUE 2))),fl((senc((SUPI,pk(skUE 2),SNname),KEMkey(Encaps(pk(skHN),rUE 2))),KEMkey(Encaps(pk(skHN), rUE 2)))),SNname,rSN 2) \sim X_3 = (ssID_3,KEMCipher(Encaps(pk(skUE_2),rHN_2)),SHA((keyseed((f3((k,KEMkey(Encaps(pk(skUE_2),rHN_2)))), f4((k,KEMkey(Encaps(pk(skUE_2),rHN_2)))),KEMkey(Encaps(pk(skUE 2),rHN 2)),f2((k,KEMkey(Encaps(pk(skUE_2),rHN_2)))),SNname)),rSN_2)),senc((keyseed((keyseed((f3((k,KEMkey(Encaps(pk(skUE 2),rHN 2)))), f4((k,KEMkey(Encaps(pk(skUE_2),rHN_2)))),KEMkey(A trace has been found. Encaps(pk(skUE 2),rHN 2)),xor(rSN 2,f5((k,KEMkey(Encaps(pk(skUE_2),rHN_2)))),SNname)),SNname)), SUPI),xor(keyseed((f3((k,KEMkey(Encaps(pk(skUE 2), rHN_2)))),f4((k,KEMkey(Encaps(pk(skUE_2),rHN_2)))), KEMkey(Encaps(pk(skUE_2),rHN_2)),f2((k,KEMkey(Encaps(pk(skUE 2),rHN 2))),SNname)),f5((k,KEMkey(Encaps(pk(skUE_2),rHN_2))))),xor(rSN_2,f5((k, KEMkey(Encaps(pk(skUE_2),rHN_2)))),fl((k,KEMkey(Encaps(pk(skUE_2),rHN_2)),rSN_2)),xor(f5((k,KEMkey(Encaps(pk(skUE 2),rHN 2))),rSN 2)) \sim M 5 = KEMCipher(Encaps(pk(skUE 2),rHN 2)) \sim M 6 = xor(rSN 2,f5((k,KEMkey(Encaps(pk(skUE 2),rHN 2)))) \sim M_7 = fl((k,KEMkey(Encaps(pk(skUE_2),rHN_2)),rSN_2)) \sim M 8 = keyseed((f3((k,DecapsKey(skUE 2,KEMCipher(Encaps(pk(skUE 2),rHN 2)))),f4((k,DecapsKey(skUE 2, KEMCipher(Encaps(pk(skUE 2),rHN 2)))),DecapsKey(skUE_2,KEMCipher(Encaps(pk(skUE_2),rHN_2))),f2((k,DecapsKey(skUE_2,KEMCipher(Encaps(pk(skUE_2), rHN 2))))),SNname)) \sim M 9 = senc(kseafUE,keyseed((keyseed((f3((k,DecapsKey(skUE 2,KEMCipher(Encaps(pk(skUE_2),rHN_2)))), f4((k,DecapsKey(skUE 2,KEMCipher(Encaps(pk(skUE_2), rHN_2))))),DecapsKey(skUE_2,KEMCipher(Encaps(pk(skUE 2),rHN 2))),xor(rSN_2,f5((k,KEMkey(Encaps(pk(skUE_2),rHN_2)))),SNname)),SNname))) Honest Process Attacker {1}new idHN_3 \sim M = pk(skHN) Beginning of process UE {9}new rUE 2 {12}new skUE 2 Beginning of process SN Beginning of process HN {16} event UESendReqSN(fl((senc((SUPI,pk(skUE_2), SNname),KEMkey(Encaps(pk(skHN),rUE_2))),KEMkey(Encaps(pk(skHN),rUE 2)))) $(\sim M_1, \sim M_2, \sim M_3, \sim M_4)$ $\sim X 1$ {37}new rSN 2 $\{38\}$ new ssID 3 {39} event SNSendReqHN(fl((senc((SUPI,pk(skUE_2), SNname),KEMkey(Encaps(pk(skHN),rUE_2))),KEMkey(Encaps(pk(skHN),rUE 2)))) $\sim X 2$ {63}event HNRecReqSN(fl((senc((SUPI,pk(skUE_2), SNname), KEMkey(Encaps(pk(skHN), rUE_2))), KEMkey(Encaps(pk(skHN),rUE 2)))) {64}new rHN 2 {79} event HNSendResSN(fl((k,KEMkey(Encaps(pk(skUE 2), rHN 2)),rSN 2))) ~X 3 {42} event SNRecResHN(fl((k,KEMkey(Encaps(pk(skUE 2), rHN 2)),rSN 2))) $(\sim M \ 5, \sim M \ 6, \sim M \ 7)$ $(\sim M \ 5, \sim M \ 6, \sim M \ 7)$ {22} event UERecResSN(fl((k,KEMkey(Encaps(pk(skUE 2), rHN 2)),rSN 2))) {30} event UESendConSN(keyseed((f3((k,DecapsKey(skUE_2,KEMCipher(Encaps(pk(skUE_2),rHN_2)))), f4((k,DecapsKey(skUE_2,KEMCipher(Encaps(pk(skUE_2), rHN_2))))),DecapsKey(skUE_2,KEMCipher(Encaps(pk(skUE_2),rHN_2))),f2((k,DecapsKey(skUE_2,KEMCipher(Encaps(pk(skUE 2),rHN 2)))),SNname))) ~M 8 \sim M 9 \sim M 10 = debugUE

Abbreviations

 \sim M 1 = KEMCipher(Encaps(pk(skHN),rUE 2))

 \sim M 2 = senc((SUPI,pk(skUE 2),SNname),KEMkey(Encaps(

pk(skHN),rUE 2)))

The attacker has the message $\sim M_10 = \text{debugUE}$ in

phase 1