Abbreviations
~M_5 = AES_CMAC(AES_CMAC(AES_CMAC(AES_CMAC(ZERO, concat(concat(concat(concat(PI,PCap),PS),p256(gen,exp_P_1)),gen)),p256(gen,exp_P_1)),prck),concat(rand_prov_2,auth_val_3))
~M_7 = AES_CCM(keys,AES_CMAC(AES_CMAC(AES_CMAC(ZERO,concat(concat(AES_CMAC(ZERO,concat(concat(concat(concat(PI,PCap),PS),p256(gen,exp_P_1)), gen)),rand_prov_2),rand_prov_2)),p256(gen,exp_P_1)), prsk),AES_CMAC(AES_CMAC(AES_CMAC(ZERO,concat(concat(AES_CMAC(ZERO,concat(concat(concat(PI,PCap), PS),p256(gen,exp_P_1)),gen)),rand_prov_2),rand_prov_2)), p256(gen,exp_P_1)),prsn))

A trace has been found.

Honest Process

Attacker

{1}new exp_P_1
{2}new exp_D_1

Beginning of process invite_prov

Beginning of process invite_dev

Beginning of process pubkey_exchange_noob_prov

Beginning of process pubkey_exchange_noob_dev

Beginning of process auth_outputoob_prov

Beginning of process auth_outputoob_dev

Beginning of process send_data_prov

Beginning of process recv_data_dev

Beginning of process outputoob_user

~M = PI

a

{7}insert pi_table_prov(addr_dev,PI)
{8}insert pcap_table_prov(addr_dev,a)

a_1

~M_1 = PCap

{11}insert pi_table_dev(addr_prov,a_1)
{12}insert pcap_table_dev(addr_prov,PCap)

{21}get pcap_table_prov(addr_dev,a)

~M_2 = PS

~M_3 = p256(gen,exp_P_1)

gen

{18}insert pubkey_table_prov(addr_prov,p256(gen, exp_P_1))
{19}insert pubkey_table_prov(addr_dev,gen)
{20}insert dhkey_table_prov(addr_prov,p256(gen, exp_P_1))

{30}get pi_table_dev(addr_prov,a_1)

a_2

a_3

~M_4 = p256(gen,exp_D_1)

{27}insert pubkey_table_dev(addr_prov,a_3)
{28}insert pubkey_table_dev(addr_dev,p256(gen, exp_D_1))
{29}insert dhkey_table_dev(addr_dev,p256(a_3,exp_D_1))

{57}get pubkey_table_prov(addr_prov,p256(gen,exp_P_1))
{56}get pubkey_table_prov(addr_dev,gen)
{55}get dhkey_table_prov(addr_prov,p256(gen,exp_P_1))
{31}new rand_prov_2

{87}get pubkey_table_dev(addr_prov,a_3)
{86}get pubkey_table_dev(addr_dev,p256(gen,exp_D_1))
{85}get dhkey_table_dev(addr_dev,p256(a_3,exp_D_1))
{58}new rand_dev_2
{59}new auth_val_3

auth_val_3

auth_val_3

~M_5

~M_5

{42}event send_prov(p256(gen,exp_P_1))

~M_6 = rand_prov_2

~M_6 = rand_prov_2

{46}event recv_prov(p256(gen,exp_P_1))
{51}insert key_table_prov(addr_dev,AES_CMAC(AES_CMAC(AES_CMAC(ZERO,concat(concat(AES_CMAC(ZERO,concat(concat(concat(concat(PI,PCap),PS),p256(gen,exp_P_1)), gen)),rand_prov_2),rand_prov_2)),p256(gen,exp_P_1)), prsk))
{54}insert nonce_table_prov(addr_dev,AES_CMAC(AES_CMAC(AES_CMAC(ZERO,concat(concat(AES_CMAC(ZERO,concat(concat(concat(concat(PI,PCap),PS), p256(gen,exp_P_1)),gen)),rand_prov_2),rand_prov_2)), p256(gen,exp_P_1)),prsn))

{92}get key_table_prov(addr_dev,AES_CMAC(AES_CMAC(AES_CMAC(ZERO,concat(concat(AES_CMAC(ZERO,concat(concat(concat(concat(PI,PCap),PS),p256(gen,exp_P_1)), gen)),rand_prov_2),rand_prov_2)),p256(gen,exp_P_1)), prsk))
{91}get nonce_table_prov(addr_dev,AES_CMAC(AES_CMAC(AES_CMAC(ZERO,concat(concat(AES_CMAC(ZERO,concat(concat(concat(concat(PI,PCap),PS),p256(gen,exp_P_1)), gen)),rand_prov_2),rand_prov_2)),p256(gen,exp_P_1)), prsn))

~M_7

The attacker tests whether
sdec(~M_7,AES_CMAC(AES_CMAC(AES_CMAC(ZERO,concat(concat(AES_CMAC(ZERO,concat(concat(concat(concat(~M,~M_1),~M_2),~M_3),gen)),~M_6),~M_6)),~M_3), prsk),AES_CMAC(AES_CMAC(AES_CMAC(ZERO,concat(concat(AES_CMAC(ZERO,concat(concat(concat(concat(~M,~M_1), ~M_2),~M_3),gen)),~M_6),~M_6)),~M_3),prsn)) = keys
is equal to
a_4.
This gives the attacker some information on the secret.