~M\_14 = PRF(PRF(preMasterSecret\_2,clientRandom\_4, serverRandom 4), server finished\_label, hash(((clientRandom\_4, serverRandom\_4,cert(subjectServer2\_1,pk(skServer2\_1), skCA\_1),penc(preMasterSecret\_2,pk(skServer2\_1))), PRF(PRF(preMasterSecret\_2,clientRandom\_4,serverRandom\_4), client\_finished\_label,hash((clientRandom\_4,serverRandom\_4, cert(subjectServer2\_1,pk(skServer2\_1),skCA\_1), penc(preMasterSecret\_2,pk(skServer2\_1))))))))  $\sim$ M\_15 = constructURL(subjectServer2\_1,path\_1)  $\sim$ M 16 = HMAC(constructURL(subjectServer2\_1,path\_1), A trace has been found. PRF(PRF(preMasterSecret\_2,clientRandom\_4,serverRandom\_4), serverRandom\_4,clientRandom\_4))  $\sim X_1 = (getURL(\sim M_8), \sim M_16) = (constructURL(subjectServer2_1,$ path\_1),HMAC(constructURL(subjectServer2\_1,path\_1), PRF(PRF(preMasterSecret\_2,clientRandom\_4,serverRandom\_4), serverRandom\_4,clientRandom\_4))) ~M\_17 = sign(makeMUDfile(constructURL(subjectServer2\_1, path\_1),aclList\_3),skServer2\_1) ~M\_18 = cert(subjectServer2\_1,pk(skServer2\_1),skCA\_1)  $\sim$  M 19 = HMAC((sign(makeMUDfile(constructURL(subjectServer2\_1, path\_1),aclList\_3),skServer2\_1),cert(subjectServer2\_1, pk(skServer2\_1),skCA\_1)),PRF(PRF(preMasterSecret\_2, clientRandom\_4,serverRandom\_4),serverRandom\_4, clientRandom 4))  $\sim$ M\_4 = cert(subjectServer2\_1,pk(skServer2\_1),skCA\_1) **Honest Process** Attacker {1}new skCA\_1  $\sim$ M = pk(skCA 1) {4}new skRogueCA\_1  $(\sim M_1, \sim M_2) = (pk(skRogueCA_1), skRogueCA_1)$ {7}new skServer\_3 {8}new subjectServer\_1  $\sim$ M\_3 = cert(subjectServer\_1,pk(skServer\_3),skCA\_1) {12}new skServer2\_1 {13}new subjectServer2\_1 ~M 4 = cert(subjectServer2\_1,pk(skServer2\_1),skCA\_1) [17] new skServerAttacker\_1  $(\sim M_5, \sim M_6) = (cert(subjectServer_1, pk(skServerAttacker_1), pk(skServer_1), pk(skServer_1), pk(skServer_1), pk(skServer_1), pk(skServer_1), pk(s$ skRogueCA 1),skServerAttacker\_1) {21} new path\_1 ~M\_7 = certDevice(subjectServer\_1,constructURL(subjectServer\_1,path\_1),skRogueCA\_1) Beginning of process Thing Beginning of process MUD\_File\_Server Beginning of process MUD\_Manager(pk(skCA\_1)) {113} event thingSentURL(constructURL(subjectServer2\_1, path\_1)) ~M\_8 = certDevice(subjectServer2\_1,constructURL( subjectServer2\_1,path\_1),skCA\_1) {115}event endThing ~M\_8 = certDevice(subjectServer2\_1,constructURL(subjectServer2\_1,path\_1),skCA\_1) {78} event managerReceivedURL(constructURL(subjectServer2\_1, path\_1)) {79}new clientRandom 4  $\sim$ M 9 = clientRandom 4 ~M 9 = clientRandom 4 {55} event serverAuthnStarted(cert(subjectServer2\_1, pk(skServer2\_1),skCA\_1)) {56} new serverRandom 4  $( \ M_10, \ M_11) = (serverRandom_4, cert(subjectServer2_1, pk(skServer2_1), skCA_1))$  $(\sim M_10, \sim M_4) = (serverRandom_4, cert(subjectServer2_1, pk(skServer2_1), skCA_1))$ [84] new preMasterSecret\_2 ~M 12 = penc(preMasterSecret 2,pk(skServer2 1)) ~M 13 ~M\_12 = penc(preMasterSecret\_2,pk(skServer2\_1)) ∼M 13 ~M 14 ~M 14 {93} event serverAuthnSuccessful(cert(subjectServer2\_1, pk(skServer2\_1),skCA\_1)) {95} event managerSentURL(constructURL(subjectServer2\_1, path\_1))  $(\sim M_15, \sim M_16)$ ~X\_1 {70} event serverReceivedURL(constructURL(subjectServer2\_1, path\_1)) {71}new aclList\_3 {73} event serverSentMUDfile(cert(subjectServer2\_1, pk(skServer2\_1),skCA\_1),makeMUDfile(constructURL(subjectServer2\_1,path\_1),aclList\_3))  $(\sim M_17, \sim M_18, \sim M_19)$ {75}event endFileServer  $(\sim M 17, \sim M 4, \sim M 19)$ {104} event managerReceivedMUDfile(cert(subjectServer2\_1, pk(skServer2\_1),skCA\_1),makeMUDfile(constructURL(subjectServer2\_1,path\_1),aclList\_3)) {105}event managerSentConfiguration(constructURL(subjectServer2\_1,path\_1),configurations(aclList\_3))  $\sim$ M 20 = configurations(aclList 3) {107}event endManager

Abbreviations

~M\_13 = PRF(PRF(preMasterSecret\_2,clientRandom\_4,

serverRandom 4), client\_finished\_label, hash((clientRandom\_4,

serverRandom\_4,cert(subjectServer2\_1,pk(skServer2\_1),

skCA\_1),penc(preMasterSecret\_2,pk(skServer2\_1)))))