~M_2 = choice[aenc(((vid_9,n_9),sign((vid_9,n_9),vsk_7),cert(vid_9,pk(vsk_7),cask_4)),pk(cask_4)), aenc(((vid_8,n_8),sign((vid_8,n_8),vsk_6),cert(vid_8,pk(vsk_6),cask_5)),pk(cask_5))] $\begin{array}{l} \sim X_1 = (\sim M_3, \sim M_4, \sim M_5) = \frac{-1}{choice} [(attvid_5, attsk_5, cert(attvid_5, pk(attsk_5), cask_4)), (attvid_4, attsk_4, cert(attvid_4, pk(attsk_4), cask_5))] \end{array}$ ~M 7 = choice[aenc(((vid_11,n_11),sign((vid_11,n_11),vsk_9),cert(vid_11,pk(vsk_9),cask_4)),pk(cask_4)),aenc(((vid_10,n_10),sign((vid_10,n_10),vsk_8),cert(vid_10,pk(vsk_8),cask_5)),pk(cask_5))] A trace has been found. ~M_8 = choice[aenc(((groupkey_response,vid_9,n_9, gsk(vid_9,gmsk_4),gpk(gmsk_4)),sign((groupkey_response, vid_9,n_9,gsk(vid_9,gmsk_4),gpk(gmsk_4)),cask_4)), pk(vsk_7)),aenc(((groupkey_response,vid_8,n_8, gsk(vid_8,gmsk_5)),sign((groupkey_response, vid_8,n_8,gsk(vid_8,gmsk_5)),gpk(gmsk_5)),cask_5)), pk(vsk_6))] **Honest Process** Attacker {1}new gmsk_4 {2}new cask_4 {3}new gmsk_5 {4}new cask_5 \sim M = pk(choice[cask_4,cask_5]) {10} new vid_10 {10} new vid_8 {11}new vsk_8 {11} new vsk_6 {12} new n_10 {12} new n_8 {13}new vid_11 {13} new vid_9 {14} new vsk_9 {14} new vsk_7 {15}new n_11 {15} new n 9 $\sim M_1 = \frac{\text{choice}[\text{cert}(\text{vid}_9,\text{pk}(\text{vsk}_7),\text{cask}_4),\text{cert}(\text{vid}_8,\text{pk}(\text{vsk}_6),\text{cask}_5)]}{\text{vid}_8,\text{pk}(\text{vsk}_6),\text{cask}_5)]$ {239}new attvid_4 {240}new attsk_4 {21} event ValidGroupKeyRequestSent(choice[vid_9, vid_8]) {241}new attvid_5 {242}new attsk_5 {245}event choice[AttackerGetsEnrollmentCertificate(attvid_5,pk(attsk_5)),AttackerGetsEnrollmentCertificate(attvid_4,pk(attsk_4))] ~M 2 ~X 1 \sim M_6 = choice[cert(vid_11,pk(vsk_9),cask_4),cert(vid_10,pk(vsk_8),cask_5)] {21} event ValidGroupKeyRequestSent(choice[vid_11, vid_10]) ~M 7 **~**M 2 {287} get v 119: table suchthat (if choice[true, false] then (success?(1-proj-revokedcerts(v_119)) && (1-proj-2-tuple(1-proj-3-tuple(choice[((vid_9,n_9),sign((vid_9,n_9),vsk_7),cert(vid_9,pk(vsk_7),cask_4)),caught-fail])) =nf 1-proj-revokedcerts(v_119))) else (success?(1-proj-revokedcerts(v_119)) && (1-proj-2-tuple(1-proj-3-tuple(choice[caught-fail,((vid_8,n_8),sign((vid_8,n_8),vsk_6), cert(vid_8,pk(vsk_6),cask_5))])) = nf 1-proj-revokedcerts(v_119)))): else branch taken {270} event choice [ValidGroupKeyRequestReceived(cask_4,vid_9),ValidGroupKeyRequestReceived(cask_5, vid_8)] {275}event choice[ValidGroupPrivateKeySent(vid_9, gsk(vid_9,gmsk_4),gpk(gmsk_4)),ValidGroupPrivateKeySent(vid_8,gsk(vid_8,gmsk_5),gpk(gmsk_5))] **~**M 8 ~M 8 {30}new vpseudosk_10 {31}new vpseudosk_11 {32} new m_16 [52] if choice[true,false]
This process performs a test that may succeed on one side and not on the other.

Abbreviations