Tamarin

```
lemma SecrecyClientToServer:
 all-traces
 "∀ k m #i. (ClientSentMessage( k, m ) @ #i) ⇒ (¬(∃ #x. K( m ) @ #x))"
/*
guarded formula characterizing all counter-examples:
"∃ k m #i. (ClientSentMessage( k, m ) @ #i) ∧ ∃ #x. (K( m ) @ #x)"
*/
simplify
solve( !ClientKey( k ) ▶₁ #i )
 case 3_Client
 solve( !KU( ~m ) @ #vk )
   case ClientSendMessage
   solve( splitEqs(1) )
     case split_case_1
     solve( !KU( g_s^~c ) @ #vk.2 )
       case 1_Client
       SOLVED // trace found
     qed
   qed
 qed
qed
```

Proverif

```
new m: Plaintext creating m_4 at {18} in copy a

new i: Z creating i_1 at {2} in copy a_1

out(io, ~M) with ~M = exp(g,i_1) at {4} in copy a_1

in(io, g) at {5} in copy a_1

event ClientAcceptedKey(exp(g,i_1)) at {7} in copy a_1

insert ClientKey(exp(g,i_1)) at {8} in copy a_1

get ClientKey(exp(g,i_1)) at {21} in copy a

event ClientSentMessage(m_4,exp(g,i_1)) at {19} in copy a (goal)

out(io, ~M_1) with ~M_1 = enc(m_4,exp(g,i_1)) at {20} in copy a

The event ClientSentMessage(m_4,exp(g,i_1)) is executed at {19} in copy
a.
The attacker has the message dec(~M_1,~M) = m_4.
A trace has been found.
RESULT not (event(ClientSentMessage(m_4,k_6)) && attacker(m_4)) is
false.
```

Verifpal

```
Result • confidentiality? m1 — When:

g_c → G^nil ← mutated by Attacker (originally G^c)
k_cs → G^nil^s
c1 → ENC(G^nil^s, CONCAT(stoc, m1))
k_sc → G^s^c
msg1_c → DEC(G^s^c, ENC(G^nil^s, CONCAT(stoc, m1)))
stoc_c → SPLIT(DEC(G^s^c, ENC(G^nil^s, CONCAT(stoc, m1))))?
m1_c → SPLIT(DEC(G^s^c, ENC(G^nil^s, CONCAT(stoc, m1))))?
unnamed_0 → ASSERT(SPLIT(DEC(G^s^c, ENC(G^nil^s, CONCAT(stoc, m1))))?, stoc)?
c2 → ENC(G^c^s, CONCAT(stoc, m1)) ← mutated by Attacker (originally ENC(k_sc, msg2_c))
msg2_s → DEC(G^nil^s, ENC(G^c^s, CONCAT(stoc, m1)))
ctos_s → SPLIT(DEC(G^nil^s, ENC(G^c^s, CONCAT(stoc, m1))))?
          m1 (m1) is obtained by Attacker.
```