

User

Peer

Shared information: cipher suite, secret  $s$ ,  $UserId$ , and  $PeerId$

`psa_pake_setup()`  
`psa_pake_set_user()`  
`psa_pake_set_peer()`

Generate  $x_1$  and  $x_2$   
Compute public keys  $g_1$  and  $g_2$   
Compute ZKP  $(V_1, r_1)$  for  $g_1$  and  $(V_2, r_2)$  for  $g_2$

Generate  $x_3$  and  $x_4$   
Compute public keys  $g_3$  and  $g_4$   
Compute ZKPs  $(V_3, r_3)$  for  $g_3$  and  $(V_4, r_4)$  for  $g_4$

`psa_pake_output()` for  $g_1, V_1, r_1, g_2, V_2$ , and  $r_2$

$(g_1, V_1, r_1, g_2, V_2, r_2)$

$(g_3, V_3, r_3, g_4, V_4, r_4)$

`psa_pake_input()` for  $g_3, V_3, r_3, g_4, V_4$ , and  $r_4$

Verify ZKPs and compute  $A$  and ZKP  $(V_5, r_5)$  for  $x_2 * s$

Verify ZKPs and compute  $B$  and ZKP  $(V_6, r_6)$  for  $x_4 * s$

`psa_pake_output()` for  $A, V_5$ , and  $r_5$

$(A, V_5, r_5)$

$(B, V_6, r_6)$

`psa_pake_input()` for  $B, V_6$ , and  $r_6$

Verify ZKP and compute  $K_a$

Verify ZKP and compute  $K_b$

If both sides used the same secret  $s$ , then  $K_a = K_b$

`psa_pake_get_shared_key()` to extract  $K_a$