```
--[ SessKeyC( IB, Kcib ) ]->
[]
rule Register_pk:
    [ Fr(~ltkC) ]
    -->
    [ !Puk($C, ~ltkC), !Pubk($C, pk(~ltkC)), Out(pk(~ltkC)) ]
rule Reveal_ltk:
    [ !Puk(C, ltkC) ] --[ LtkReveal(C) ]-> [Out(ltkC) ]
rule WD_1:
    let m1 = sign{'1', $IDwd, ~nwd}skWD
    in
       [ Fr(~nwd), !Pubk($IDC, pkC), !Sk($IDwd, skWD) ]
    --[ OUT_WD_1 (m1) ]->
       [ Out ( m1 ), WD_1(<$IDwd, ~nwd, sign{'1', $IDwd, ~nwd}skWD>, pkC) ]
rule C_1:
    let m1 = sign{'1', $IDc, ~nc}skC
    in
       [ Fr(~nc), !Pubk($IDWD, pkWD), !Sk($IDc, skC) ]
    --[ OUT_C_1 (m1) ]->
       [ Out ( m1 ), C_1(<$IDc, ~nc, sign{'1', $IDc, ~nc}skC>, pkWD) ]
rule C_2:
   let  m1 = aenc{<nc, IDc, sign{'1', IDc, nc}skC>}pk(ltkIB)
        m2 = aenc{<nc, ~nwd, $IDwd, Amt_wd, POI_wd, sign{'2', nc, ~nwd, $IDwd,
Amt_wd, POI_wd, TID_wd}skib>}pkC
    in
       [!Pubk($IDwd, ltkWD), In ( m1), !M_data(Amt_wd, POI_wd, TID_wd), !Sk($IDwd,
skib),  !Pubk(IDc, pkC), Fr(~nwd)]
    --[IN_WD_1_nc (nc, m1 ), OUT_WD_1 ( m2 ) , Running(IDc, $IDwd, <'init' ,nc, ~nwd,
Amt_wd, POI_wd, TID_wd>)]->
       [ Out ( m2 ) , WD_1(<$IDwd, IDc, nc, ~nwd, Amt_wd, POI_wd, TID_wd, sign{'2',
nc, ~nwd, $IDwd, Amt_wd, POI_wd, TID_wd}skIB>, pkC) ]
rule C_3:
   let m2 = aenc{<nc, nwd, IDwd, Amt_wd, POI_wd, TID_wd>}pk(ltkC)
       m3 = aenc{<nwd, PI_cib, nc,  IDwd, Amt_wd, POI_wd, TID_wd, sign{'3', nwd,
PI_cib, nc, IDwd, Amt_wd, POI_wd, TID_wd}skC>}pkIB
   in
      [ C_2(IDc, nc), !Pubk( C, ltkC), In ( m2 ), !C_data(PI_cib), !Sk(IDc, skC), !Pubk(IB,
pkIB) ]
   --[IN_C_3_nwd( nwd, m2), OUT_C_3( m3 ), Commit(IDc, IDwd, <'init', nc, nwd>),
Running(IDwd, IDc, <'resp', nc, nwd>)]->
     [Out (m3), Secret (IDc, IDwd, nwd), Secret (IDc, IDwd, nc) ]
rule IB_1:
[ !Ltk($IB, ~ltkIB)
, In( request )
]
```